

Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»



**Информационные технологии и системы 2019
(ИТС 2019)**

Материалы международной научной конференции
(БГУИР, Минск, Беларусь, 30 октября 2019)

**Information Technologies and Systems 2019
(ITS 2019)**

Proceeding of The International Conference
(BSUIR, Minsk, Belarus, 30th October 2019)

Минск БГУИР 2019

УДК 004
ББК 32.81
И74

Редакционная коллегия:

Л. Ю. Шилин (главный редактор), Д.П. Кукин, А. А. Иванюк, А. В. Марков, В. В. Голенков, А. А. Навроцкий, В. С. Муха, А. Ф. Трофимович, Я. Г. Гриневич, С.В. Колосов, А. Б. Гуринович (ответственный секретарь)

Информационные технологии и системы 2019 (ИТС 2019) :
И74 материалы международной научной конференции, БГУИР, Минск, Беларусь, 30 октября 2019 г. – Information Technologies and Systems 2019 (ITS 2019) : Proceeding of The International Conference, BSUIR, Minsk, 30th October 2019/ редкол. : Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2019. – 305 с.

ISBN 978-985-543-542-7.

Сборник включает прошедшие рецензирование доклады международной научной конференции «Информационные технологии и системы 2019» (ИТС 2019).

Сборник предназначен для преподавателей высших учебных заведений, научных сотрудников, студентов, аспирантов, магистрантов, а также для специалистов предприятий в сфере IT-технологий.

Материалы сборника одобрены организационным комитетом и печатаются в авторской редакции.

**УДК 004
ББК 32.81**

ISBN 978-985-543-542-7

© УО «Белорусский государственный университет информатики и радиоэлектроники», 2019

ПРЕДИСЛОВИЕ

30 октября 2019 года в городе Минске прошла международная научная конференция «Информационные технологии и системы» (ИТС 2019), организованная факультетом информационных технологий и управления учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», базовой организации государств-участников СНГ по высшему образованию в области информатики и радиоэлектроники.

В течение дня на пленарном и пяти секционных заседаниях было представлено 147 докладов по следующим направлениям:

Секция «Системы управления»:

- моделирование систем управления;
- автоматизированные исследования систем управления;
- проектирование систем управления;
- микропроцессорные системы управления;
- информационные технологии в системах управления;
- системы передачи и защиты дискретной информации;
- системы телемеханики;
- элементы и устройства систем управления.

Секция «Интеллектуальные информационные технологии»:

- проектирование баз знаний;
- интеллектуальный поиск информации;
- интеллектуальные решатели задач;
- пользовательские интерфейсы интеллектуальных систем.

Секция «Проектирование встраиваемых систем»:

- проектирование и быстрое прототипирование встраиваемых систем на ПЛИС;
- методологии проектирования IP-компонент встраиваемых систем;
- программная и аппаратная защита IP-компонент;
- контролепригодное проектирование и тестирование встраиваемых систем;
- реконфигурируемые встраиваемые системы.

Секция «Системы обработки информации»:

- математические модели и методы обработки информации;
- статистические, численные, логические, графовые, нейросетевые и другие;
- анализ многомерных данных;
- методологические проблемы и технологии подготовки специалистов в области систем обработки информации;
- технологии и стандарты программирования и проектирования систем обработки информации;
- системы и средства совместной работы пользователей;
- защита информации в автоматизированных системах обработки информации.

Секция «Математическое моделирование и компьютерная графика»:

- методы решения уравнений математической физики;
- методы нелинейной оптимизации;
- модели и методы радиофизики и приборов СВЧ;
- модели и методы тепло и массопереноса;
- математические модели и методы компьютерной графики;
- аппаратное обеспечение компьютерной графики;
- алгоритмы и методы интерактивных программных продуктов.

В работе конференции приняли участие учёные и специалисты Республики Беларусь, Российской Федерации, Украины.

Докладчики представляли как высшие учебные заведения, в их числе: БГУ, БГУИР, БНТУ, БГТУ, БГЭУ, БрГТУ, ПГУ, МИУ, Военная академия РБ, Гомельский государственный технический университета им. П. О. Сухого, Гродненский государственный университет им. Янки Купалы, ГГУ им. Ф. Скорины, Волжский государственный университет водного транспорта, Каспийский институт морского и речного транспорта, ГУВПО «Белорусско-Российский Университет», Могилевский государственный университет продовольствия, Luxembourg Institute of Health, International University of Beirut, Евразийский национальный университет имени Гумилёва – так и институты и коммерческие организации: НАН РБ, Объединенный институт проблем информатики НАН РБ, ИДСТУ СО РАН, ОАО «АГАТ-системы управления», филиал АО «СО ЕЭС» «Региональное диспетчерское управление энергосистемы Иркутской области», EPAM Systems.

Оргкомитет благодарит всех, кто счел возможным поделиться результатами своей деятельности и принял участие в работе конференции.

ORGANISING COMMITTEE

Prof.Dr.	Vadim Bogush	Rector of The Belarusian State University of Informatics and Radioelectronics
PhD.	Sergey Dick	First Vice-Rector of BSUIR
Ph.D.	Anatoly Osipov	Vice-Rector for Research of BSUIR
Ph.D.	Valeriy Prutkov	Vice-Rector for Education of BSUIR
Ph.D.	Maxim Davydov	Vice-Rector for Education of BSUIR
Prof.Dr.	Leonid Shilin	Dean of The Faculty of Information Technologies and Control of BSUIR
	Dmitry Kuznetsov	Vice-rector for Personal Development of BSUIR

PROGRAMME COMMITTEE

Prof.Dr.	Vladimir Mukha	Professor of Information Technologies in Automated Systems Department of BSUIR
Prof.Dr.	Vladimir Golenkov	Professor of Intellectual Information Technologies Department of BSUIR
Prof.Dr.	Vladimir Golovko	Head of Intellectual Information Technologies Department of BrSTU
Prof.Dr.	Alexander Ivaniuk	Professor of Computer Science Department of BSUIR
Ph.D.	Dmitry Kukin	Head of Computational Methods and Programming Department of BSUIR
Ph.D.	Alexander Markov	Head of Control Systems Department of BSUIR
Ph.D.	Anatoly Navrotsky	Head of Information Technologies in Automated Systems Department of BSUIR
Ph.D.	Vadim Zhuravlev	Head of Theoretical Bases of Electrical Engineering Department of BSUIR
Prof.Dr.	Stanislav Kolosov	Professor of Computational Methods and Programming Department of BSUIR
Ph.D.	Daniil Shunkevich	Head of Intellectual Information Technologies Department of BSUIR

Пленарные доклады

Plenary session

ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫЕ ФУНКЦИИ

Заливако С. С., Иванюк А. А.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {zalivako, ivaniuk}@bsuir.by

В статье рассмотрены базовые компоненты физической криптографии цифровых устройств – физически неклонлируемые функции (ФНФ). Интерес к данной тематике обусловлен возможностью извлечения уникальных характеристик интегральных схем (ИС) для построения неклонлируемых идентификаторов и генерирования случайных числовых последовательностей. Как правило, схемотехнические реализации ФНФ гораздо более эффективны с точки зрения аппаратных затрат в сравнении с классическими алгоритмами шифрования и хеширования. Приведен обзор основных областей применения ФНФ, а также описаны методы экспериментального исследования архитектур ФНФ на FPGA (Field-Programmable Gate Array).

ВВЕДЕНИЕ

Количество устройств Интернета вещей (Internet of Things, IoT), подключенных к сети, в 2018 составило порядка 7 млрд [1]. По прогнозу компании Ericsson [2] их число к 2020 году увеличится до 30 млрд в первую очередь за счет развития мобильной связи пятого поколения. В связи с повсеместным распространением Интернета вещей одной из важнейших проблем в настоящее время является безопасность и конфиденциальность доступа к хранимым устройствами данным [3]. Решением этой проблемы является применение методов аппаратной и программной криптографии: шифрования, обфускации, внедрения цифровых водяных знаков и отпечатков пальцев, идентификации, аутентификации и др. [4]. Поскольку рассматриваемый класс устройств является требовательным к энергопотреблению и площади кристалла интегральной схемы (ИС), методы физической криптографии оказались более предпочтительными [5]. Для аппаратной реализации методов физической криптографии, как правило, используются физически неклонлируемые функции (ФНФ), которые предназначены для генерирования секретных ключей и идентификаторов цифровых устройств для их последующей аутентификации [6].

В данной статье приведены результаты научных исследований по тематике физически неклонлируемых функций под руководством профессора Иванюка А. А. в период с 2012 по 2019 годы. Впервые на постсоветском пространстве проблематика ФНФ была опубликована профессором Ярмоликом В. Н. в 2011 году в журнале “Информатика” [7]. Профессор Ярмолик является ведущим специалистом в области проектирования надежных цифровых устройств и систем.

1. ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫЕ ФУНКЦИИ

Формально ФНФ может быть описана значениями пар входных и соответствующих им выходных параметров, которые для ФНФ, реализованных в полупроводниковых устройствах,

являются соответственно значениями входных сигналов запроса CH (Challenge) и значениями выходных сигналов ответа R (Response) [8]. Любая ФНФ может быть представлена с помощью множества всевозможных пар запрос-ответ (Challenge-Response Pairs, CRP), а также является функцией преобразования множества запросов CH_i во множество ответов R_i :

$$R_i = \text{PUF}(CH_i) \quad (1)$$

В силу того, что современные ИС обладают множеством физических характеристик, точные значения которых являются уникальными и непредсказуемыми для каждой произведенной копии, эти параметры могут быть использованы в качестве основы для реализации ФНФ. Исторически первыми были оптические ФНФ, основанные на направленной под определенным углом световой волне (запрос) для получения уникальной интерференционной картины (ответ) [9]. В работе [10] впервые было предложено использовать различия в задержке распространения сигнала по симметричным путям для реализации ФНФ типа арбитр (Arbiter PUF). Эта идея в дальнейшем была использована, например, в ФНФ, основанной на таблицах поиска (Lookup Table PUF) [11]. Разность частот кольцевых генераторов (Ring Oscillator PUF) [12], а также уникальность значений частот (Bistable Ring PUF) [13] были использованы в качестве основы для генерирования пар запрос-ответ. Множество реализаций ФНФ основано на использовании состояния элементов памяти после инициализации: ФНФ на основе статического оперативного запоминающего устройства (СОЗУ) (SRAM PUF) [14], динамической памяти с произвольным доступом (DRAM PUF) [15], ФНФ типа бабочки (Butterfly PUF) [16], ФНФ на основе асинхронного RS-триггера (SR latch PUF) [17], а также магнитно-резистивной оперативной памяти, основанной на спиновых эффектах (Spin-Transfer Torque Magnetoresistive Random-Access Memory) [18]. Также влияние качественного и количественного состава покрытия ИС на значения

электрической емкости ее элементов (Coating PUF) [19] было успешно использовано для реализации идентификации и аутентификации ИС с помощью ФНФ.

С учетом многообразия возможных реализаций ФНФ на кристалле ИС следует выделить несколько областей применения ФНФ: цифровые водяные знаки и отпечатки пальцев [20], генерирование случайных числовых последовательностей [21], идентификация и аутентификация [22], реализация аппаратных хэш-функций [23], обнаружение аппаратных закладок [24], генерирование ключей шифрования [25], радиочастотные идентификаторы [26] и др. Более того, ведущие мировые производители полупроводниковых устройств применяют ФНФ в устройствах интернета вещей (Samsung [27]), серийно выпускаемых программируемых логических интегральных схемах (Intel [28], Xilinx [29]), защите цифровых устройств от нелегального копирования (Qualcomm [30]), устройствах радиочастотной идентификации (Intrinsic ID [31]), цифровых узлах транспортных средств (Accenture [32]) и других разработках.

В данной статье рассматривается неклонированная идентификация и аутентификация, схемотехнические решения, направленные на снижение уязвимости к криптографическим атакам, генерирование случайных числовых последовательностей, а также экспериментальные исследования ФНФ с помощью аппаратно-программных комплексов.

II. НЕКЛонируемая Идентификация ЦИФРовых УСТРОЙств

Классическая реализация ФНФ типа арбитр (АФНФ) основана на сравнении задержек двух идентичных сигналов, распространяемых по топологически идентичным путям [10]. Схемная реализация АФНФ приведена на рис. 1.

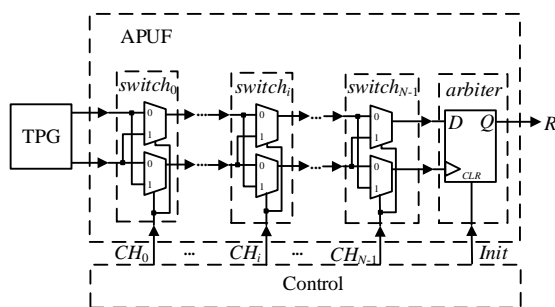


Рис. 1 – Классическая реализация ФНФ типа арбитр

Модуль TPG (Test Pulse Generator) предназначен для генерирования переднего и/или заднего фронта сигнала в зависимости от реализации схемы арбитра. Каждый из переключателей

$switch_i$ имеет два режима работы: прямой — когда сигнал распространяется по тому же пути, и перекрестный — в этом случае сигнал меняет линию связи. В качестве элемента, определяющего, какой из сигналов пришел быстрее, используется синхронный D-триггер, который при небольших значениях (точные значения зависят от технологического процесса изготовления ИС) разницы задержек сигналов переходит в метастабильное состояние и тем самым значение на его выходе становится непредсказуемым. В результате данного эффекта показатель стабильности АФНФ, реализованных на ПЛИС, не превышает 0,6 при максимальном значении 1,0.

Для обнаружения метастабильного состояния арбитра было предложено два схемотехнические решения [33]. Одно из решений, основанное на применении четырех D-триггеров, приведено на рис. 2.

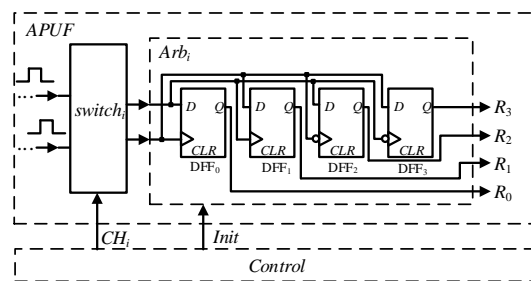


Рис. 2 – Реализация арбитра на основе четырех D-триггеров

Предлагаемая реализация арбитра строится не только передним (триггеры DFF₀ и DFF₁), но и задним фронтом (триггеры DFF₂ и DFF₃) тестового импульса. Следовательно, значения на выходе арбитра показывают взаимоотношение целых импульсов, а не только их части (передних фронтов), как в классической реализации АФНФ. Таким образом, ответ модифицированного арбитра может быть представлен четырехразрядным числом $\{R_0, R_1, R_2, R_3\}$. Предложенная реализация арбитра на базе четырех D-триггеров позволяет обнаруживать метастабильные состояния, которые, в свою очередь, могут быть обозначены как третий символ выходного алфавита (X) в дополнение к имеющимся 0 и 1.

Другим решением для обнаружения метастабильного состояния является реализация схемы арбитра в виде асинхронного RS-триггера, как показано на рис. 3. Предложенный арбитр состоит из двух логических элементов ИЛИ-НЕ с обратной связью, а также двух D-триггеров. В отличие от классической реализации АФНФ предлагаемая схема арбитра управляется задним фронтом тестового сигнала. Как показано в работе [34], метастабильное состояние на выходе RS-триггера наблюдается в случае, когда сигнал

на входах установки и сброса одновременно переходит из состояния логической единицы в состояние логического нуля. Признаком метастабильного состояния является затухающее колебание, производимое RS-триггером. Если явление метастабильности не было обнаружено, то RS-триггер мгновенно перейдет в состояние стабильного логического нуля или единицы.

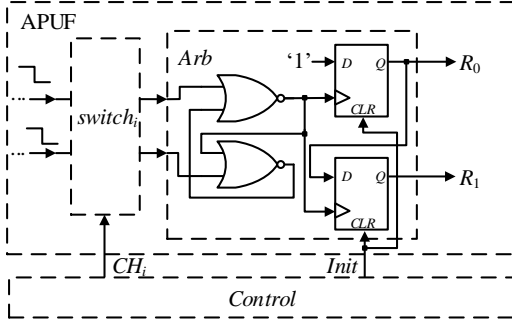


Рис. 3 – Реализация арбитра на основе асинхронного RS-триггера

Таким образом, на выходе арбитра возможно появление трех стабильных состояний: логического нуля, логической единицы и высокочастотного затухающего колебания (High Frequency Oscillation, HFO). Обнаружение перечисленных состояний может быть реализовано с помощью сдвигового регистра, состоящего из двух D-триггеров: стабильный ноль кодируется как пара $(R_0 = 0, R_1 = 0)$, стабильная единица – как $(R_0 = 1, R_1 = 0)$, а состояние HFO – как $(R_0 = 1, R_1 = 1)$. Следовательно, значение бита ответа R_1 показывает, что арбитр находится в

стабильном (когда $R_1 = 0$) или метастабильном ($R_1 = 1$) состоянии.

По результатам тестирования предложенная реализация АФНФ с модифицированным арбитром как на базе четырех D-триггеров, так и RS-триггера, продемонстрировала высокие значения характеристик уникальности ($\approx 0,49$) и стабильности ($\approx 0,99$). Таким образом, в результате характеристика стабильности была значительно улучшена с 0,57 до 0,99 с учетом дополнительных аппаратных затрат, не превышающих 2 % от исходной реализации схемы арбитра.

Использование ФНФ в таких приложениях, как медицинская электроника [35], RFID-карты доступа к объектам повышенной секретности [36], системы управления стратегическими объектами (например, атомными электростанциями) [37] и т. п. требует предельно высокого уровня стабильности ее ответов ($\approx 1,0$).

Модифицированные схемы арбитров для обнаружения метастабильных состояний позволяют значительно улучшить характеристики стабильности, но, к сожалению, не обеспечивают требуемого уровня стабильности. В связи с этим появляется необходимость в разработке более точной модели АФНФ не только для обнаружения метастабильных состояний, но и их предсказания с вероятностью не менее 0,95 по виду подаваемого запроса. Уточненная математическая модель задержки времени распространения сигнала позволила разработать критерии для тестирования запросов с целью их классификации на менее стабильные (слабые) и более стабильные (сильные).

На рис. 4 показана структурная схема АФНФ.

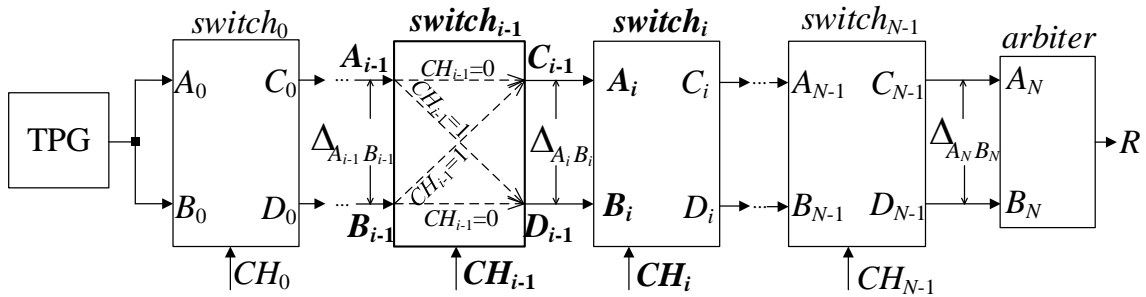


Рис. 4 – Структурная схема ФНФ типа арбитр

Согласно определению ФНФ, множества значений $\{\Delta_{C_{i-1}A_{i-1}}, \Delta_{D_{i-1}A_{i-1}}, \Delta_{C_{i-1}B_{i-1}}, \Delta_{D_{i-1}B_{i-1}}\}$ и $\{\Delta_{A_i C_{i-1}}, \Delta_{B_i D_{i-1}}\} \forall i = 1, \dots, N$ являются уникальными и неповторяющимися не только для схемной реализации всех звеньев одной схемы на одном полупроводниковом кристалле, но и на множестве кристаллов.

В зависимости от подаваемого значения CH_{i-1} формируются два уникальных маршрута прохождения двух тестовых импульсов от входных портов A_{i-1}, B_{i-1} до портов следующей компоненты A_i, B_i . Как было показано выше, на выходе арбитра R может появиться метастабильное состояние, что ухудшает характеристику стабильности ФНФ. Такая ситуация происходит в

результате того, что разность во времени задержки сигналов, распространяющихся по симметричным путям, попадает в интервал $[t_{low}, t_{high}]$. В свою очередь, значения t_{low} и t_{high} определяются временными характеристиками схемы арбитра. Для классической реализации с помощью единственного D-триггера значение $t_{low} = -t_{hold}$, а значение $t_{high} = t_{setup}$ (t_{setup} , t_{hold} – время предустановки и удержания триггера, соответственно). Если же арбитр был реализован на основе RS-триггера, то параметры t_{low} и t_{high} определяются в зависимости от характеристик напряжения и симметричности реализации логических элементов ИЛИ-НЕ, которые лежат в основе реализации RS-триггера [34].

Учет перечисленных параметров для схемы арбитра гарантирует появление стабильного значения ответа на выходе R . В противном случае схема арбитра может оказаться в метастабильном состоянии, при котором значение ответа на выходе R будет непредсказуемым. В итоге значение ответа на выходе R зависит от результирующей разницы между фронтами сигналов $\Delta_{A_N B_N}$:

$$R = \begin{cases} 0, & \text{если } \Delta_{A_N B_N} \leq t_{low}, \\ 1, & \text{если } \Delta_{A_N B_N} \geq t_{high}, \\ X, & \text{если } t_{low} < \Delta_{A_N B_N} < t_{high}. \end{cases} \quad (2)$$

Значение результирующей разницы $\Delta_{A_i B_i}$ для блока $switch_i$ формально можно выразить следующей функцией γ от двух аргументов:

$$\Delta_{A_i B_i} = \gamma(\delta_{i-1}^{CH_{i-1}}, \Delta_{A_{i-1} B_{i-1}}), \quad (3)$$

где $\delta_{i-1}^{CH_{i-1}}$ – уникальная характеристика звена $switch_{i-1}$, значение которой зависит от бита запроса CH_{i-1} ; $\Delta_{A_{i-1} B_{i-1}}$ – временная разница фронтов сигналов на входе звена $switch_{i-1}$.

В свою очередь, значение $\Delta_{A_N B_N}$ можно выразить как:

$$\Delta_{A_N B_N}(CH_{N-1}, CH_{N-2}, \dots, CH_0) = \sum_{j=0}^{N-1} (\delta_j \prod_{i=0}^j Sign_i). \quad (4)$$

Функция арифметического знака задержки $Sign_i$ может быть представлена как:

$$Sign_{i-1} = 1 - 2 \cdot CH_{i-1}. \quad (5)$$

На основе описанной в соотношении (4) математической модели АФНФ был предложен алгоритм определения стабильности запроса CH на основании изменения младшего и старшего разрядов. Данный алгоритм позволяет оценить вероятность попадания ответа R на запрос CH в регион метастабильности. На рис. 5 показаны результаты параметрического моделирования 16-разрядной АФНФ на ПЛИС Xilinx Artix-7 [38].

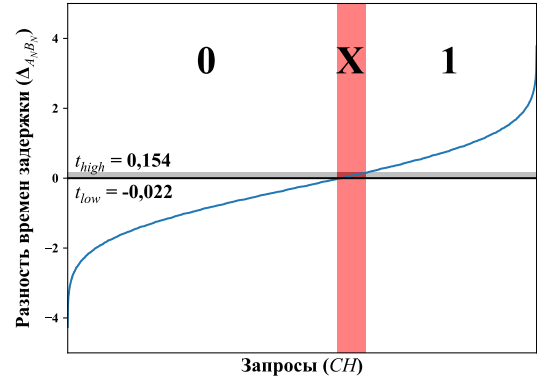


Рис. 5 – Пары запрос-ответ, отсортированные по значению $\Delta_{A_N B_N}$

В результате моделирования показано, что из $2^{16} = 65536$ пар запрос-ответ 67 % являются стабильными, а 33 % – нестабильными. Данный анализ подтверждает недостаток АФНФ с небольшой разрядностью, поскольку значительная часть пар запрос-ответ были классифицированы как нестабильные. На практике при реализации АФНФ разрядностью как минимум 128, даже несмотря на значительный процент отбракованных запросов, мощность множества стабильных пар запрос-ответ достаточна для того, чтобы произвести идентификацию без повторного использования пар запрос-ответ.

Применение алгоритмов, предложенных в работах [39–41] показало, что стабильность АФНФ была улучшена до 1,0. Алгоритм был также верифицирован в условиях изменения температуры окружающей среды от -40 до $+90$ °С. Результаты тестирования показателя P_{stable} (вероятности стабильности всего множества пар запрос-ответ) приведены на рис. 6.

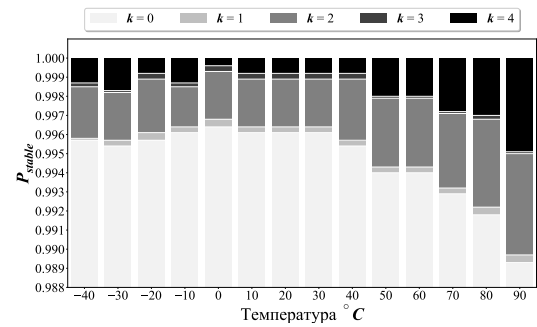


Рис. 6 – Результаты тестов стабильности при изменении температуры

Показатель k обозначает количество бит запроса, измененных для тестирования запроса. Например, если $k = 2$, то изменению подвергаются два младших бита запроса. Следовательно, для обеспечения вероятности 1,0 достаточно изменить в запросе 4 младших бита для определения его стабильности.

Также была экспериментально исследовано 20 идентичных ПЛИС Xilinx Artix-7 и одна

ПЛИС Xilinx ZC706 [42]. В результате было установлено, что расположение региона метастабильности, а, соответственно, и показатель уникальности АФНФ обусловлен как особенностями конкретного кристалла ПЛИС, так и расположением ФНФ на кристалле. Результат эксперимента показан на рис. 7.

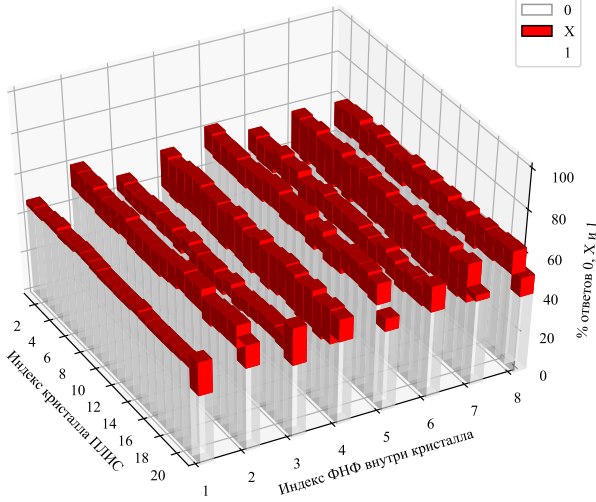


Рис. 7 – Распределение регионов 0, X и 1 для различных кристаллов и компонент АФНФ

Было показано, что генерирование уникального неклонированного идентификатора цифрового устройства может быть эффективно реализовано с помощью ФНФ типа арбитр на платформе ПЛИС. При этом отсутствие идеальной симметрии путей может быть скомпенсировано за счет модификации схема арбитра, а также применения алгоритмов тестирования запросов с целью определения вероятности их стабильности. Таким образом, стабильность и уникальность генерируемого идентификатора может быть обеспечена в том числе при реализации ФНФ на ПЛИС.

III. СНИЖЕНИЕ УЯЗВИМОСТИ К КРИПТОГРАФИЧЕСКИМ АТАКАМ С ПОМОЩЬЮ МАШИННОГО ОБУЧЕНИЯ В ПРОТОКОЛАХ АУТЕНТИФИКАЦИИ

Разность времени задержки распространения сигнала по симметричным путям в АФНФ является линейной функцией, как показано ранее. Аналогично соотношению (2) значение задержки Δ может быть представлено как функция от запроса CH . Не ограничивая общности, представим зависимость ответа R ФНФ для идеально симметричного арбитра на основе единственного D-триггера (в этом случае $t_{low} = t_{high} = 0$):

$$R = \begin{cases} 0, & \text{если } \Delta(CH) < 0, \\ 1, & \text{если } \Delta(CH) > 0. \end{cases} \quad (6)$$

На рис. 8 показана зависимость значения задержки $\Delta(CH)$ в зависимости от отсортиро-

ванных бинарных значений запросов CH для 16-разрядной АФНФ. Данные для графика получены в результате проведения параметрического моделирования ПЛИС Xilinx ZC706 в среде Vivado. Из рис. 8 видно, что осуществить разделение двух классов (значений логического нуля и единицы) по значению $\Delta(CH)$ возможно с помощью линейной функции (прямой линии).

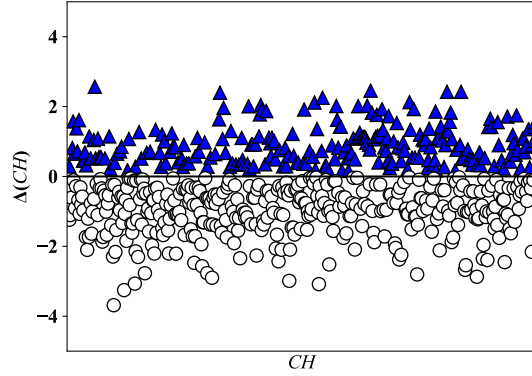


Рис. 8 – Зависимость разности задержек распространения сигнала Δ в зависимости от значения запроса CH

В связи с тем, что разделимость множества пар запрос-ответ очень высокая, многие исследователи показали, что модель, основанная на идеальном арбитре, позволяет правильно предсказать порядка 98–99 % для 64-разрядной АФНФ, имея от 300 до 400 пар запрос-ответ [43], с использованием метода опорных векторов или логистической регрессии.

Авторами предложен [40–41] метод снижения уязвимости к криптографическим атакам с помощью машинного обучения. Данный метод основан на предварительной обработке запросов АФНФ с помощью многоканального сигнатурного анализатора (Multiple Input Signature Register, MISR). MISR позволяет произвести обфускацию запросов АФНФ, тем самым делая зависимость ответов от запросов нелинейной. Более того, основным предназначением блока MISR является вычисление сигнатур для тестирования входных данных, когда ИС спроектирована по принципу тестопригодности (Design For Testability, DFT). Таким образом, если на FPGA реализованы процедуры для самотестирования, то MISR может использоваться не только для тестирования, но и для обработки запросов АФНФ без дополнительных аппаратных затрат.

Схемная реализация предлагаемой модификации АФНФ с помощью блока MISR приведена на рис. 9. Как правило, MISR является частью структуры самотестирования BILBO (Built-In Logic Block Observer), которая может быть сконфигурирована в четырех режимах: чтения входных данных, памяти, сдвигового регистра с обратной связью и многоканального сигнатурно-

го анализатора. В предлагаемой реализации используется только два режима: режим памяти ($\alpha_0 = 0$), когда инициализирующее значение загружается в MISR, и режим MISR ($\alpha_0 = 1$) для вычисления запросов, подаваемых на вход АФНФ.

Как показали последние исследования [44], несмотря на отсутствие у злоумышленника пря-

мого доступа к множеству пар запрос-ответ, криптографическая атака может быть проведена с помощью оптимизации методом черного ящика [45]. Наиболее применимым в настоящее время является метод эволюционной стратегии адаптации матриц ковариации (Covariance Matrix Adaptation Evolution Strategy, CMA-ES) [46].

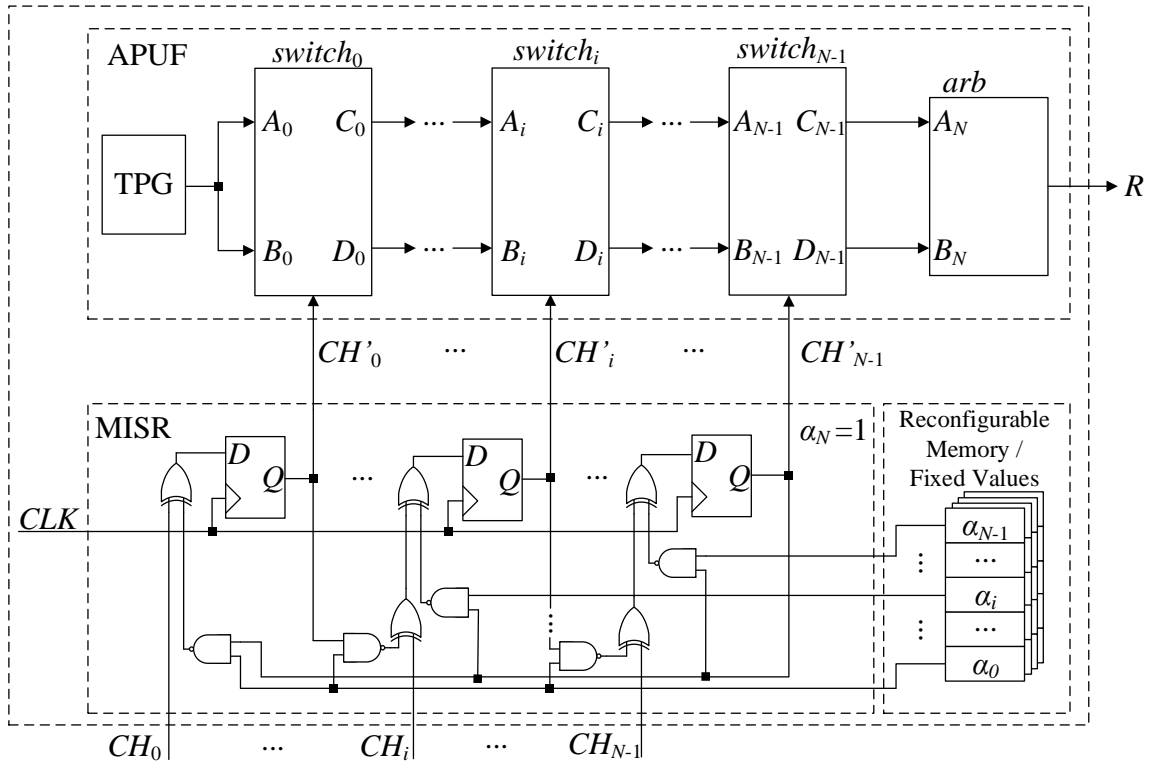


Рис. 9 – Предложенная модификация АФНФ с помощью MISR

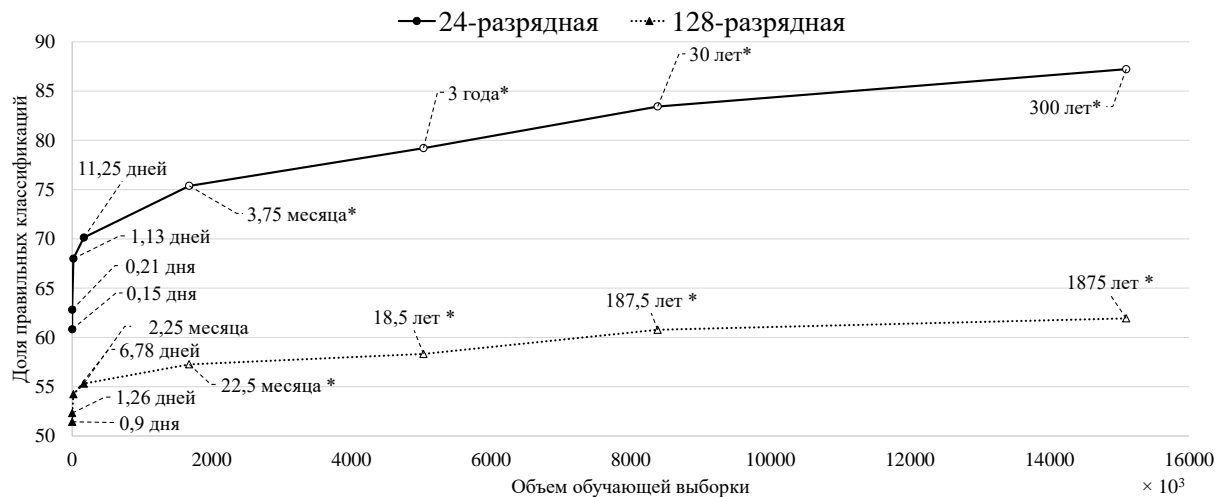


Рис. 10 – Доля успешных классификаций в зависимости от объема обучающей выборки

Данная модификация АФНФ была подвержена криптографическим атакам с помощью методов опорных векторов и эволюционной стра-

тегии. Максимальное значение доли предсказанных значений составило 55 % с применением эволюционной стратегии при этом объеме обучаю-

щей выборки составил порядка 10^9 пар запрос-ответ и время обучения – около 3 месяцев. Результат криптографической атаки показан на рис. 10 (точки черного цвета на графике обозначают реальные экспериментальные данные, а точки белого цвета – экстраполированные значения).

Таким образом, применение MISR для обфускации запросов АФНФ позволило снизить

практическую уязвимость к криптографическим атакам с помощью машинного обучения, сравнимую с применением алгоритмов хеширования SHA-256 [40–41].

На основе данной модификации АФНФ авторами был предложен протокол аутентификации, общая схема которого приведена на рис. 11.

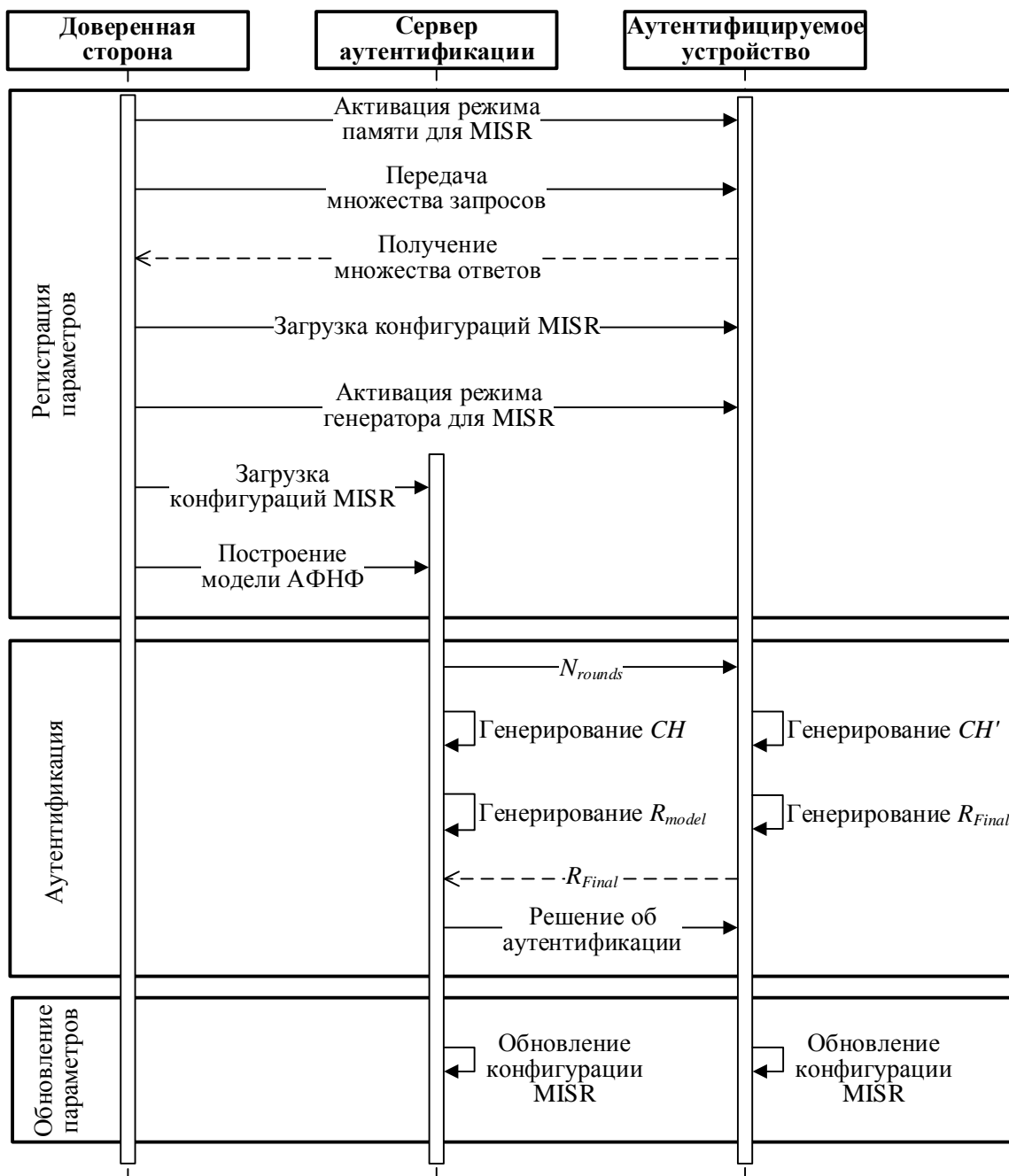


Рис. 11 – Протокол аутентификации на основе модифицированной АФНФ

Предложенный протокол включает в себя три этапа:

1. **Регистрация параметров.** На данном этапе осуществляется инициализация сервера аутентификации и аутентифицируемого устройства путем настройки параметров. На устройстве активируется режим памяти, чтобы запросы не обрабатывались MISR. После чего происходит генерирование множества пар запрос-ответ (например, для 24-разрядной АФНФ требуется несколько миллионов пар, а для 128-разрядной – несколько миллиардов), чтобы построить модель АФНФ с помощью искусственной нейронной сети. Доверенная сторона сохраняет модель АФНФ, а также генерирует параметры MISR (коэффициенты полинома (i) и начальное значение ($seed$)), которые затем загружаются в реконфигурируемую память аутентифицируемого устройства. Далее происходит активация режима MISR на устройстве, а также передача модели АФНФ и параметров MISR на сервер аутентификации. Таким образом, устройство и сервер готовы к следующему этапу.

2. **Собственно аутентификация.** Процесс аутентификации инициируется устройством. Далее сервер отправляет K случайных значений N_{rounds} , в результате чего на устройстве формируется значение ответа R_{Final} , которое отправляется обратно на сервер. Программная модель АФНФ на сервере используется для вычисления ответа R_{model} , который в силу точ-

ности построенной модели должен совпадать со значением R_{Final} . Решение об аутентификации принимается блоком анализатора запросов на сервере на основании равенства или неравенства значений R_{Final} и R_{model} .

3. **Обновление параметров.** С целью обеспечения большей надежности протокола параметры MISR предполагается обновлять не реже раза в месяц. Поскольку количество возможных полиномов в соответствии с функцией Эйлера для 128-разрядного MISR составляет $\approx 1,3 \times 10^{36}$, случайная выборка нескольких из них для периодического обновления позволит избежать криптографической атаки методом исчерпывающего перебора вариантов.

Предложенный протокол позволяет произвести надежную аутентификацию устройства, содержащего АФНФ, без необходимости хранить экспоненциально большое число пар запрос-ответ на сервере, поскольку модель АФНФ потребляет в сотни раз меньше ресурсов памяти для ее хранения.

В отличие от существующих методов построения программных моделей АФНФ, которые позволяют добиться точности от 95 до 98 %, разбиение ответов на четверки и их классификация позволяет добиться стопроцентной точности. Авторами было предложено использовать классификатор, состоящий из трех этапов, которые показаны на рис. 12.

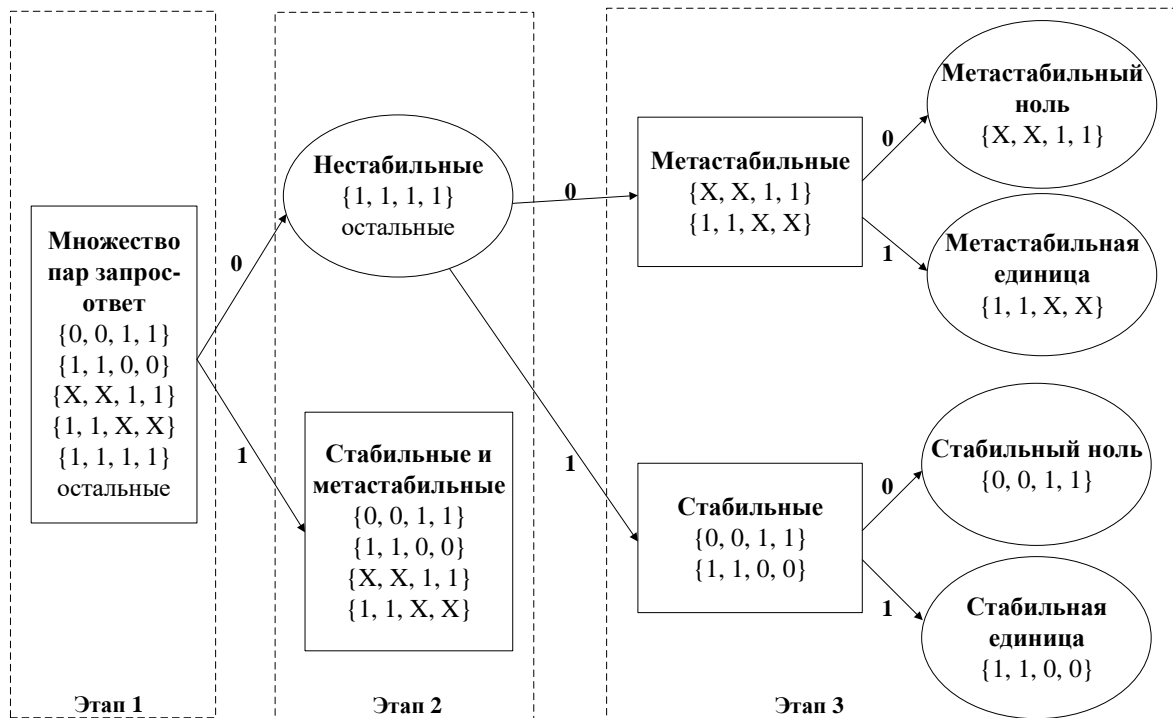


Рис. 12 – Структурная схема алгоритма классификации

Первый этап классификации является наиболее сложным, поскольку алгоритму необходимо отделить стабильные четверки-ответы от нестабильных. Для решения данной задачи была использована глубокая нейронная сеть (Deep Neural Network, DNN), архитектура которой показана на рис. 13.

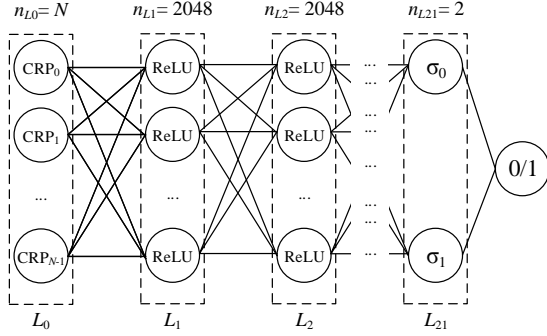


Рис. 13 – Структурная схема алгоритма классификации

Входной слой сети (L_0) состоит из N узлов, каждый из которых является значением знака задержки в запросе ($Sign_i = \{-1, 1\}$, $i = 0..N - 1$). Выходы первого слоя далее последовательно подаются на 20 скрытых слоев (L_1, \dots, L_{20}), каждый из которых состоит из 2048 нейронов с линейно-пороговой функцией активации (Rectified Linear Unit, ReLU). Выходной слой состоит из двух узлов 0 и 1 с обобщенной логистической функцией активации (Softmax). Таким образом, обученная сеть по значениям знаков задержек выдает ровно два значения: вероятность p_1 того, что это стабильная четверка-ответ ($\{0, 0, 1, 1\}$, $\{1, 1, 0, 0\}$, $\{X, X, 1, 1\}$, $\{1, 1, X, X\}$) и вероятность нестабильной четверки p_0 ($\{1, 1, 1, 1\}$ и другие значения), $p_0 + p_1 = 1$. Таким образом, этап 1 определяет, является ли стабильным запрос, поданный на вход сети.

В данном эксперименте были построены модели двух различных конфигураций АФНФ (24- и 128-разрядная). В каждом случае множество пар запрос-ответ было разбито на три подмножества: обучающая (80 % пар), валидационная (10 %) и экзаменационная выборки. Каждый из запросов был повторно подан на вход АФНФ $E = 100$ раз для определения стабильности каждой из четверок-ответов, наблюдаемых в эксперименте. В соответствии с алгоритмом мажоритарного выбора каждая четверка была помечена в обучающей выборке одним из пяти возможных вариантов: $\{0, 0, 1, 1\}$, $\{1, 1, 0, 0\}$, $\{X, X, 1, 1\}$, $\{1, 1, X, X\}$, $\{1, 1, 1, 1\}$. Если в результате эксперимента была получена четверка, отличающаяся от перечисленных выше, то данный запрос был отнесен к классу “остальные”. Проблема переобучения предлагаемой сети была решена с помощью регуляризации второго порядка, а также алгоритма сброса весов с вероятностью $p = 0,5$.

Множество пар запрос-ответ, используемое для обучения сети, содержало 2^{22} и 10^{10} пар для 24- и 128-разрядной АФНФ соответственно. Для каждой реализации АФНФ процент правильных классификаций составил 100 на обучающей, валидационной и тестовой выборках.

Второй этап алгоритма классификации имеет меньшую сложность, поскольку необходимо разделять стабильные ответы-четверки от метастабильных. В связи с этим нейронная сеть вместо 20 слоев содержала только три в отличие от первого этапа. Аналогично нейронной сети, состоящей из 20 слоев, данная модель также продемонстрировала 100 % правильных классификаций как для 24-разрядной, так и для 128-разрядной АФНФ.

Третий этап классификации был реализован с помощью линейного метода классификации (логистической регрессии). Поскольку основная масса четверок-ответов (86,94 %) относится к классам $\{0, 0, 1, 1\}$, $\{1, 1, 0, 0\}$, то использование простого алгоритма позволит сократить временные издержки как на обучение модели, так и на предсказание метки в процессе аутентификации.

Построение точной модели АФНФ с помощью методов машинного обучения позволило значительно снизить уязвимость предложенного протокола к криптографическим атакам.

IV. ГЕНЕРАТОРЫ СЛУЧАЙНЫХ ЧИСЛОВЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ НА ОСНОВЕ ФНФ

Предложенная авторами [47–48] структура генератора случайных числовых последовательностей (ГСЧП) состоит из трех компонентов: источника случайности, схемы сжатия и регистра случайного числа.

Источник случайности вырабатывает начальную последовательность, которая, как правило, не обладает необходимыми статистическими характеристиками, чтобы использоваться в криптографических приложениях. Схема сжатия применяется в генераторе для улучшения статистических свойств начальной случайной последовательности с целью применения ее в различных приложениях (в том числе криптографических). Регистр случайного числа предназначен для хранения элементов генерируемой случайной последовательности.

Результаты исследования вероятностных характеристик ГСЧП, построенных на основе ФНФ, позволяют сделать вывод о том, что практически любая реализация цифровой ФНФ на базе FPGA применима для генерирования случайных числовых последовательностей высокого качества. Однако всегда стоит принимать во внимание характеристику стабильности, высокое значение которой говорит о том, что ФНФ может быть эффективно использована для идентификации ПЛИС или проекта ПЛИС.

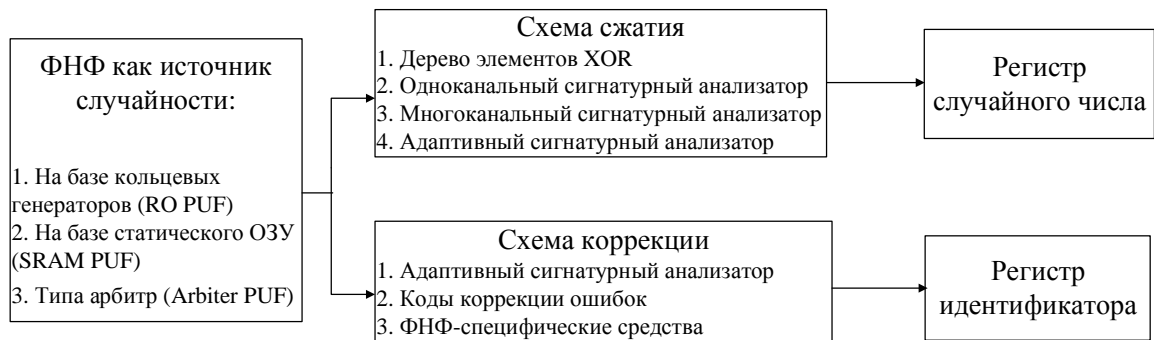


Рис. 14 – Общая структура устройства, работающего в двух режимах: идентификации и генерирования случайных числовых последовательностей

В качестве иллюстрации метода синтеза цифровых устройств на основе ФНФ, поддерживающих режимы идентификации и генерирования случайных числовых последовательностей, на рис. 14 приведена обобщенная структура устройств данного класса. Множество пар запрос-ответ ФНФ может быть условно разделено на два подмножества пар с высокой и низкой стабильностью. Идентификаторы цифрового устройства могут быть построены на основе достаточного числа максимально стабильных пар, которые также обладают высокой характеристической уникальностью. В свою очередь, пары с низкой стабильностью, которые по сути и характеризуют вариации технологического процесса изготовления ИС, следует использовать для генерирования случайных числовых последовательностей.

ФНФ типа арбитр является более эффективной как с точки зрения идентификации, так и ГСЧП в силу простоты реализации и меньших аппаратных затрат по сравнению с другими классическими ФНФ. По скорости генерирования последовательностей предлагаемые решения ограничены 1 Мбит/с в силу невысокой производительности используемого семейства ПЛИС, а также протокола UART. При реализации данных ГСЧП на ПЛИС Xilinx Zynq-7000 производительность может быть увеличена до 3 Мбит/с при использовании протокола UART и до 100 Мбит/с при использовании Ethernet различных стандартов.

Предложенные авторами ГСЧП успешно проходят тесты из статистических пакетов NIST [49] и Diehard [50], что позволяет сделать вывод о действительной случайности генерируемых ими числовых последовательностей.

V. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫХ ФУНКЦИЙ

Авторами предложен общий подход к исследованию характеристик ФНФ (уникальности,

стабильности). Например, для достоверной оценки уникальности требуется как минимум 10 идентичных ИС, содержащих реализацию ФНФ [51]. В связи с этим особенностью реализованной экспериментальной установки является возможность параллельно получать данные с каждой реализации ФНФ.

С другой стороны, реализованная АФНФ обладает расширенным выходным алфавитом за счет обнаружения метастабильных состояний, поэтому алгоритмы подсчета метрик качества были адаптированы для тернарных векторов. Более того, на каждой ПЛИС было реализовано несколько компонент ФНФ, что позволило оценить не только межкристальную, но и внутрикристальную уникальность.

Для исследования характеристик АФНФ была реализована экспериментальная установка, построенная на базе 10 ПЛИС Xilinx Artix-7, входящих в состав плат быстрого прототипирования Digilent Nexys-4, а также сервера Fujitsu PRIMERGY Econel 200 на базе двухъядерного процессора Intel Xeon 5050 с тактовой частотой 3 ГГц и объемом оперативной памяти 8 Гб под управлением операционной системы MS Windows Server 2012. Доступ к серверу осуществлялся удаленно с помощью утилиты Remote Desktop Connection из БГУИР (г. Минск) и НТУ (г. Сингапур). Общая схема эксперимента приведена на рис. 15.

Проектное описание АФНФ было создано на языке VHDL с использованием САПР Xilinx ISE System Edition 14.7. На каждой ПЛИС было реализовано по D компонент мультиарбитражных АФНФ. Разрядность МАФНФ N и, соответственно, количество арбитров задавалось на стадии проектирования. Целью данного эксперимента было повышение стабильности АФНФ за счет использования различных схем арбитра: на основе единственного D-триггера, четырех D-триггеров и асинхронного RS-триггера. Соответственно, тип арбитра также конфигурировался

в исходном коде проектного описания. Создание конфигурационного файла и программирование ПЛИС было осуществлено стандартными средствами САПР (модуль ISE iMPACT).

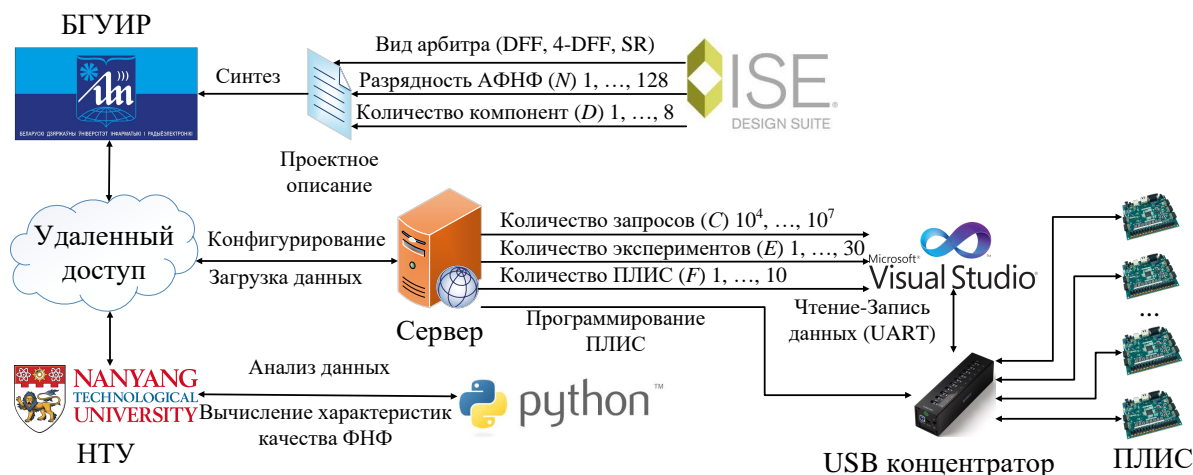


Рис. 15 – Обобщенная схема экспериментальной установки

ПЛИС Xilinx Artix-7 XC7A100T-1CSG324C изготовлена по 45 нм техпроцессу и содержит 15850 секций по шесть четырехходовых LUT-блоков и восемь синхронных D-триггеров, 4860 Кб встроенной статической памяти (BRAM), четыре блока цифровой обработки сигналов, 300 пользовательских блоков ввода / вывода. Плата быстрого прототипирования Digilent Nexys-4 содержит источник тактового импульса частотой до 450 МГц для синхронизации ПЛИС. Для проведения эксперимента платы были соединены в стойку и подключены к серверу через 10-портовый USB концентратор ST Lab U-500, как показано на рис 16.

Передача данных между сервером и ПЛИС была реализована по протоколу UART. Для реализации получения и передачи данных были использованы стандартные порты UART_TXD_IN, UART_RXD_OUT, UART_CTS, UART_RTS платы Nexys-4. Для генерирования запросов был реализован LFSR (Linear Feedback Shift Register) с разрядностью, совпадающей с количеством звеньев АФНФ (N). Значения запросов являлись слабокоррелированными, поскольку каждый из них вырабатывался только спустя N тактов после предыдущего. Ответы АФНФ, в свою очередь, хранились в регистровом файле.

Соответственно, передача данных от регистрового файла на интерфейс UART реализована в виде отдельного контроллера ФНФ. В свою очередь, программное обеспечение для передачи данных от сервера на АФНФ (ПЛИС) было разработано на языке C# в среде Microsoft Visual Studio. Количество ПЛИС (F), программируемых в данном эксперименте, можно было задать программно, как и число запросов (C). Некото-

рые из экспериментов были многократно повторены E раз. Данный параметр также задается с помощью программного обеспечения без необходимости перепрограммирования ПЛИС.

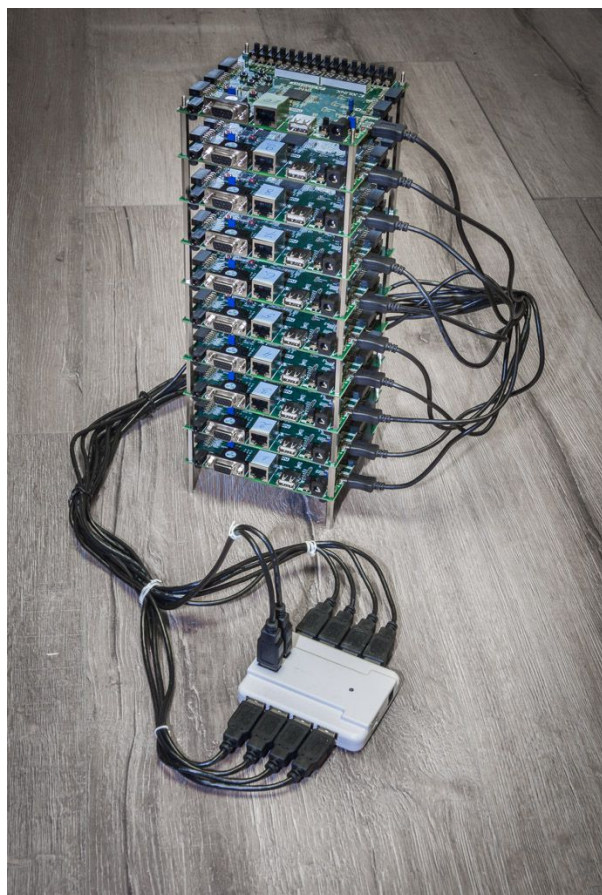


Рис. 16 – Стойка для 10 плат быстрого прототипирования Digilent Nexys-4 и USB концентратор

Данные, полученные в результате эксперимента, сохранялись в текстовые файлы специального формата, а затем анализировались с помощью алгоритмов расчета стабильности и уникальности на языке Python. В силу того, что выходной алфавит АФНФ тернарный, для проверки случайности метастабильные состояния были перекодированы в значения 0 и 1 с помощью генератора псевдослучайных числовых последовательностей, реализованного на языке Python. Пары запрос-ответ АФНФ анализировались с помощью пакетов NIST и Statistica.

Как было показано ранее, характеристика стабильности ФНФ может значительно ухудшаться при изменении условий функционирования ЦУ. В связи с этим метрика стабильности была измерена в условиях изменяющейся температуры окружающей среды от -40°C до 90°C . Исследование показателя стабильности под воздействием различных температур было осуществлено с помощью температурной камеры Thermotron[®] 8800, доступ к которой был предоставлен Центром исследования спутников Наньянгского технологического университета (Nanyang Technological University Satellite Research Centre) в Сингапуре. На рис. 17 приведена фотография температурной камеры и ПЛИС, находящейся внутри.



Рис. 17 – Температурная камера Thermotron[®] 8800

Чтение и запись $C = 10000$ запросов с четырьмя возможными комбинациями дополнительного старшего и младшего бит, сгенерированных повторно $E = 100$ раз, эквивалентны обработке $4 \cdot 10^6$ запросов классической АФНФ. Обработка данного количества запросов занимает порядка 40 минут. Процесс чтения и записи был повторен в диапазоне температур от -40 до 90°C с шагом 10°C . Таким образом, проведение эксперимента требует порядка 10 часов.

Программирование температурной камеры представляет собой настройку времени работы при каждой температуре, времени перехода между температурами, а также времени остывания (нагрева) камеры до комнатной температуры. Перечисленные параметры настраиваются с помощью передней панели камеры, управляемой касаниями стилуса, как показано на рис. 18. Конфигурационный файл, содержащий алгоритм изменения температуры, сохраняется в формате *.rgm. После нажатия кнопки Run на панели управления камера автоматически изменяет температуру внутри в соответствии с заданными временными интервалами.

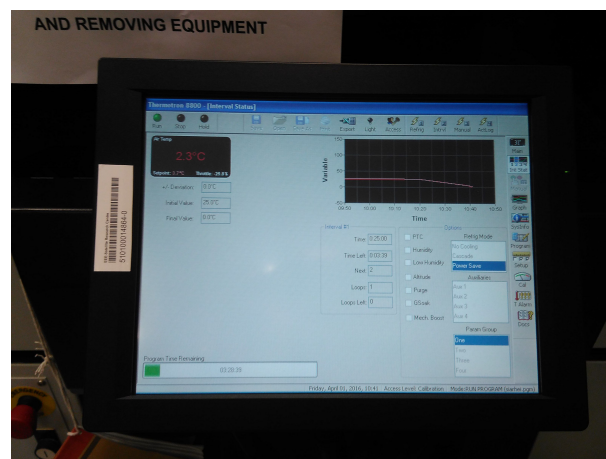


Рис. 18 – Панель настройки температурной камеры

Чтение и запись данных были осуществлены с помощью ноутбука MacBook Pro Mid 2015 на базе процессора Intel Core i5-5257U с объемом оперативной памяти 8 Гб. Кабели, соединяющие ноутбук и плату быстрого прототипирования, были помещены внутрь камеры и отделены от внешней среды с помощью предохранительной мембраны.

Проведение эксперимента в условиях изменяющейся температуры окружающей среды позволило подтвердить гипотезу о том, что устойчиво сильные запросы, выбранные в условиях комнатной температуры, сохраняют высокие характеристики стабильности.

VI. ЗАКЛЮЧЕНИЕ

В период с 2012 по 2019 годы профессором Иванюком А.А. и доцентом Заливако С.С. было опубликовано 8 статей в рецензируемых на-

учных изданиях, глава монографии в книге издательства Springer, 5 статей на международных научных конференциях (ISIC-2014 – Сингапур; ASPDAC-2016 – Макао, КНР; ISQED-2017 – Санта-Клара, США; ISCAS-2017 – Балтимор, США, PRIP-2019 – Минск, Беларусь) и более 20 тезисов докладов на международных и республиканских научных конференциях.

В настоящее время существует ряд открытых проблем, не решенных международным научным сообществом в области ФНФ: исследование и разработка новых архитектур ФНФ, реализация ФНФ на основе готовых интегральных схем, нахождение компромисса между стабильностью и предсказуемостью ФНФ, разработка точных и универсальных математических моделей ФНФ, возможности для использования ФНФ в коммерческих проектах.

Авторы в настоящее время ведут исследования в области разработки новых архитектур ФНФ на основе флэш-памяти, проектирования ФНФ на платформе ПЛИС с уменьшенными аппаратными затратами, повышения стабильности и уникальности неклонированных идентификаторов цифровых устройств, а также снижения уязвимости к криптографическим атакам с помощью машинного обучения.

Интерес международного научного сообщества и ведущих мировых производителей полупроводниковых устройств к тематике ФНФ подтверждает актуальность выбранного авторами научного направления.

VII. СПИСОК ЛИТЕРАТУРЫ

- State of the IoT 2018: Number of IoT devices now at 7B – Market accelerating [Electronic resource] / K. L. Lueth. – IoT Analytics, 2018. – Mode of access: <https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/> – Date of access: 14.10.2019.
- Internet of Things forecast [Electronic resource] – Ericsson, 2019. – Mode of access: <https://www.ericsson.com/en/mobility-report/internet-of-things-forecast> – Date of access: 14.10.2019.
- Ahmad M., A Review of Current Security Issues in Internet of Things / M. Ahmad [et al.] // Springer, 2019. – PP. 11–23.
- Chang, C. H. and Potkonjak, M. Secure System Design and Trustable Computing / C. H. Chang // Springer, 2016. – 537 P.
- Иванюк, А. А. Физическая криптография и защита цифровых устройств / А. А. Иванюк, С. С. Заливако // Доклады БГУИР – 2019. – № 2 (120). – С. 50–58.
- Chang, C. H. A Retrospective and a Look Forward: Fifteen Years of Physical Unclonable Function Advancement / C. H. Chang, Y. Zheng, L. Zhang // IEEE Circuits and Systems Magazine – 2017. – Vol. 17, № 3. – P. 32–62.
- Ярмолик, В. Н. Физически неклонированные функции / В. Н. Ярмолик, Ю. Г. Вашичко // Информатика – 2011. – № 2 (30). – С. 92–103.
- Tuyls, P. Security with Noisy Data / P. Tuyls, B. Skoric, T. Kevenaar. – Springer, 2007. – 344 P.
- Physical one-way functions / R. Pappu [et al.] // Science. – 2002. – vol. 297, № 5589. – P. 2026–2030.
- A technique to build a secret key in integrated circuits for identification and authentication applications / J. Lee [et al.] // Int. Symp. VLSI Circuits (VLSI'04). – Honolulu, USA, 2004. – P. 176–179.
- Anderson, J. H. A PUF design for secure FPGA-based embedded systems / J. H. Anderson // Proc. Asia and South Pacific Design Automat. Conf. 2010 (ASP-DAC'10). – Taipei, Taiwan, 2010. – P. 1–6.
- Silicon physical random functions / B. Gassend [et al.] // ACM Conf. on Comp. and Comm. Security (CCS'02). – New York, USA, 2002. – P. 148–160.
- The bistable ring PUF: A new architecture for strong physical unclonable functions / Q. Chen [et al.] // Proc. IEEE Int. Sympos. on Hardw. Orient. Secur. and Trust (HOST'11). – San Diego, USA, 2011. – P. 134–141.
- Holcomb, D. Initial SRAM state as a fingerprint and source of true random numbers for RFID tags / D. Holcomb, W. Burleson, K. Fu // Conf. RFID Security (RFID'07). – Malaga, Spain, 2007. – P. 1–2.
- DRAM-based intrinsic physically unclonable functions for system-level security and authentication / F. Tehranipoor [et al.] // IEEE Trans. on Very Large Scale Integr. (VLSI) Syst. – 2016. – № 99. – P. 1–13.
- The butterfly PUF protecting IP on every FPGA / S. S. Kumar [et al.] // Proc. Int. Workshop Hardware-Oriented Security and Trust (HOST'08). – Anaheim, USA, 2008. – P. 67–70.
- Uniqueness enhancement of PUF responses based on the locations of random outputting RS-latches / D. Yamamoto [et al.] // Cryptographic Hardware and Embedded Systems (CHES'11). – Nara, Japan, 2011. – P. 390–406.
- Highly reliable memory-based physical unclonable function using spintransfer torque MRAM / L. Zhang [et al.] // IEEE Int. Symp. on Circ. and Syst. (ISCAS'14). – Melbourne, Australia, 2014. – P. 2169–2172.
- Read-proof hardware from protective coatings / P. Tuyls [et al.] // Cryptographic Hardware and Embedded Systems (CHES'06). – Yokohama, Japan, 2006. – P. 369–383.
- Zheng, J. X. A digital PUF-based IP protection architecture for network embedded systems / J. X. Zheng, M. Potkonjak // Proc. ACM/IEEE Symp. on Archit. for Netw. and Comm. Syst. (ANCS'14). – Marina Del Rey, USA, 2014. – P. 255–256.
- Efficient Implementation of True Random Number Generator Based on SRAM PUFs / V. Leest [et al.] // Cryptography and Security: From Theory to Applications / ed. by D. Naccache. – Springer, 2012. – P. 300–318.
- Zalivaka, S.S. Arbiter PUF based FPGA chip identification and authentication methods with enhanced reliability and modeling attack resistance: PhD Thesis. – Singapore: NTU, 2018. – 162 P.
- Bolotnyy, L. Physically unclonable function-based security and privacy in RFID systems / L. Bolotnyy, G. Robins // Proc. IEEE Int. Conf. on Perv. Comp. and Comm. (PerCom'07). – White Plains, USA, 2007. – P. 211–218.
- Zhang, X. On-chip structures and techniques to improve the security, trustworthiness and reliability of integrated circuits / X. Zhang [Electronic resource]. – Mode of access: <http://digitalcommons.uconn.edu/cgi/viewcontent.cgi?article=6219&context=dissertations>. – PhD Thesis, 2013. – Date of access: 14.10.2019.

25. Maes, R. PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator / R. Maes, A. V. Herreweghe, I. Verbauwhede // Proc. Crypt. Hardw. and Emb. Syst. (CHES'12). – White Leuven, Belgium, 2012. – P. 302–319.
26. Verayo. Security and authentication solutions based on silicon physical unclonable functions (PUF) technology [Electronic resource]. – Mode of access: <http://www.verayo.com/>. – 2013, Verayo, Inc. – Date of access: 14.10.2019.
27. Samsung Introduces Exynos i T100 for Secure and Reliable IoT Devices with Short-Range Connectivity [Electronic resource] – Samsung, 2019. – Mode of access: <https://news.samsung.com/global/samsung-introduces-exynos-i-t100-for-secure-and-reliable-iot-devices-with-short-range-connectivity> – Date of access: 14.10.2019.
28. Altera Partners with Intrinsic-ID to Develop World's Most Secure High-End FPGA [Electronic resource] – Intel, 2015. – Mode of access: <https://newsroom.intel.com/news-releases/altera-partners-intrinsic-id-develop-worlds-secure-high-end-fpga/> – Date of access: 14.10.2019.
29. Xilinx to add PUF security to Zynq devices [Electronic resource] – EE News, 2016. – Mode of access: <https://www.eenewseurope.com/news/xilinx-add-puf-security-zynq-devices-0> – Date of access: 14.10.2019.
30. Applying circuit delay-based physically unclonable functions (PUFs) for masking operation of memory-based PUFs to resist invasive and clone attacks [Electronic resource] – Google Patents, 2013. – Mode of access: <https://patents.google.com/patent/US9787480> – Date of access: 14.10.2019.
31. SRAM PUF Solutions [Electronic resource] – Intrinsic ID, 2019. – Mode of access: <https://www.intrinsic-id.com/solutions/> – Date of access: 14.10.2019.
32. Accenture and Thales Demonstrate How Blockchain Technology Can Secure and Simplify Aerospace & Defense Supply Chains [Electronic resource] – Accenture, 2018. – Mode of access: <https://newsroom.accenture.com/news/accenture-and-thales-demonstrate-how-blockchain-technology-can-secure-and-simplify-aerospace-and-defense-supply-chains.htm> – Date of access: 14.10.2019.
33. Zalivaka, S. S. Multi-valued arbiters for quality enhancement of PUF responses on FPGA implementation / S. S. Zalivaka, A. V. Puchkov, V. P. Klybik, A. A. Ivaniuk, C. H. Chang // Proc. of IEEE Asia and South Pacific Design Automation Conf. (ASP-DAC'2016), – Macau, China – P. 533–538.
34. Kacprzak, T. Analysis of oscillatory metastable operation of an RS flip-flop / T. Kacprzak // IEEE J. of Solid-State Circ. – 1988. – Vol. 23, № 1. – P. 260–266.
35. Survey of main challenges (security and privacy) in wireless body area networks for healthcare applications / S. Al-Janabia [et al.] // Egyptian Informatics J. – 2017. – Vol. 18, № 2. – P. 113–122.
36. Highly-reliable feed through/filter capacitor and method for making same [Electronic resource]. – Mode of access: <https://www.google.com/patents/US4424551>. – 1991. – Date of access: 15.10.2019.
37. Stacey, W. M. Nuclear Reactor Physics / W. M. Stacey. – Wiley, 2007. – 707 p.
38. Nexys 4 DDR artix-7 FPGA: Trainer board recommended for ece curriculum [Electronic resource]. – Mode of access: <http://store.digilentinc.com/nexys-4-ddr-artix-7-fpga-trainer-board-recommended-for-ece-curriculum/>. – Digilent, Inc, 2017. – Date of access: 15.10.2019.
39. Заливако С. С., Иванюк А. А., Клыбик В. П. Метод увеличения стабильности физически неклонированной функции типа “арбитр” // Информатика. 2017. № 1 (53). – С. 31–43.
40. Zalivaka S. S., Ivaniuk A. A., Chang C. H. FPGA Implementation of Modeling Attack Resistant Arbiter PUF with Enhanced Reliability // Invited Paper at Special Session on IoT Security: Protocol, Implementation and Attacks, in Proc. 18th IEEE International Symposium on Quality Electronic Design (ISQED'17). Santa Clara, CA, USA, 13–15 March 2017. – P. 313–318.
41. Zalivaka S. S., Ivaniuk A. A., Chang C.-H. Reliable and Modeling Attack Resistant Authentication of Arbiter PUF in FPGA Implementation With Ternary Quadruple Response // IEEE Transactions on Information Forensics and Security. 2018. № 4 (14). – P. 1109–1123.
42. 7 series FPGAs configurable logic block [Electronic resource]. – Mode of access: https://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf. – Xilinx, Inc, 2016. – Date of access: 15.10.2019.
43. Theory and application of delay constraints in arbiter PUF / U. Chatterjee [et al.] // ACM Trans. on Emb. Comp. Syst. – 2016. – Vol. 15, № 1. – P. 1001–1020.
44. Efficient online and offline testing of embedded DRAMs / S. Hellebrand [et al.] // IEEE Trans. on Comp. – 2002. – Vol. 51, № 7. – P. 801–809.
45. Vicente, L. N. Implicitly and densely discrete black-box optimization problems / L. N. Vicente // Optimization Letters. – 2009. – Vol. 3, № 3. – P. 475–482.
46. Omidvar, M. N. A comparative study of CMA-ES on large scale global optimisation / M. N. Omidvar, X. Li // Australasian Joint Conf. on Art. Intell. – Adelaide, Australia, 2010. – P. 303–312.
47. Заливако С. С., Иванюк А. А. Использование физически неклонированных функций для генерирования действительно случайных числовых последовательностей // Автоматика и вычислительная техника. – 2013. № 3. – С. 61–72.
48. Заливако С. С., Иванюк А. А. Схемная реализация комбинированной физически неклонированной функции для генерирования действительно случайных числовых последовательностей // Докл. БГУИР. – 2013. № 7 (77). – С. 37–43.
49. A statistical test suite for random and pseudorandom number generators for cryptographic applications [Electronic resource]. – Mode of access: <http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf>. – NIST, 2010. – Date of access: 15.10.2019.
50. Diehard: A battery of tests of randomness [Electronic resource]. – Mode of access: http://stat.fsu.edu/_geo. – Florida State University, 1995. – Date of access: 15.10.2019.
51. Vijayakumar, A. On testing physically unclonable functions for uniqueness / A. Vijayakumar, V. C. Patil, S. Kundu // Proc. Int. Symp. on Qual. Electr. Design (ISQED'2016). – Santa Clara, USA, 2016. – P. 368–373.

ИНДУСТРИЯ ИГР БЕЛАРУСИ

Яцукевич В. В.
ведущий геймдизайнер ООО "Вайзор геймз".
Минск, Республика Беларусь
E-mail: blackard@vizer-interactive.com

В докладе освещаются вопросы, связанные с развитием гейминдустрии в Республике Беларусь. Мы рассмотрим, какие изменения произошли в индустрии за последние десять лет, какие новые профессии появились, как изменилась структура команды и какую роль в ней сегодня выполняет геймдизайнер.

ВВЕДЕНИЕ



Рис. 1 – Пример игровой графики

Глобальный рынок приложений и игр с каждым годом растёт, обгоняя даже самые смелые прогнозы прошлых лет (см. рис.2). Это связано с тем, что во всем мире у людей появляется всё больше свободного времени, которое они могут и готовы потратить на развлечения и развитие.



Рис. 2 – Анализ рынка игровых приложений

Вслед за мировыми тенденциями и в нашей стране закономерно начинают появляться компании, которые успешно участвуют в этом росте.

В этих компаниях на сегодняшний день работает около 3500 специалистов и почти все эти организации сейчас нуждаются в новых сотрудниках. Им нужно больше специалистов, которые смогут продолжать развивать и поддерживать существующие "выстрелившие" продукты, а также создавать новые успешные игры и приложения.

Анализ рынка игровых приложений

За последние десять лет очень сильно изменились и продукты гейминдустрии и подходы для их разработки.

Если раньше средняя команда состояла из пяти - семи специалистов и они были заняты в одном проекте от одного до шести месяцев, то сейчас средняя команда это 20-30 сотрудников и проект может длиться от двух до пяти лет. Это обусловлено тем, что пользователи с годами становятся более искушенными и разработчикам для привлечения большей аудитории приходится прикладывать достаточно серьезные усилия для поддержания своего игрового продукта в конкурентоспособном состоянии.

Серьезно поменялась гендерная составляющая: если десять лет назад на проектах были задействованы преимущественно лица мужского пола, то на данный момент соотношение в коллективах примерно 60% мужчин и 40% женщин (см. рис. 3).

Это всё результат изменения рынка игр, который позволяет работать маленьким компаниям и дает им возможность разрастаться в большие организации.

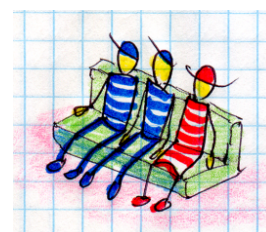
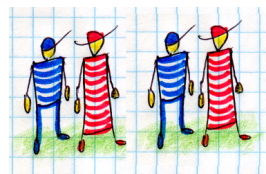


Рис. 3 – Команды разработчиков игровых приложений

В течении последних нескольких лет на рынке труда гейминдустрии появилась потребность в узкопрофильных специалистах и, соответственно, около шести новых востребованных профессиях: геймдизайнеры, маркетинг дизайнеры, комьюнити дизайнеры, специалисты по видео, аналитики, маркетологи. Все эти специалисты узкопрофильные и необходимы для поддержания развития игрового приложения, выпускаемого организациями гейминдустрии.

Это связано с увеличением объемов производства и тем, что человек, ранее совмещавший несколько профессий (например, художник, который совмещал еще и роль дизайнера), уже не в состоянии адекватно справиться с увеличенным объемом задач. Компании стали нуждаться в узкоспециализированных и высококвалифицированных сотрудниках, способных быстро реагировать на изменяющуюся конъюнктуру игрового рынка. Чтобы «вырастить» специалиста, который сможет создать конкурентоспособный продукт, необходимо потратить пять - семь лет и большое количество усилий.

Для максимальной эффективности этот обучающий процесс нужно начинать как можно раньше и сосредоточить процесс обучения на отдельной специализации, что даст возможность предприятиям получить в короткие сроки готового специалиста, а обучающемуся - некие гарантии для трудоустройства при успешном пройденном обучении.

Выводы

На данный момент существует необходимость в развитии образования в области гейминдустрии. Если некоторое время назад не было потребности в "готовых" специалистах, то сейчас, в рамках быстроразвивающейся сферы игрового бизнеса, компаниям нужен специалист, который может сразу влиться в группу разработчиков и стать в конкретную позицию исполнителя без длительного обучения основам профессии. Обучающиеся, которые проходят обучение по программам, связанным с геймдизайном, затем имеют возможность работать в игровой индустрии на должностях программиста, тестировщика, дизайнера, художника.

СПИСОК ЛИТЕРАТУРЫ

1. Новостной портал Онлайнер [Электронный ресурс] / Геймдизайнеры из белорусского IT о профессии: «Игры может делать строитель, экономист, архитектор и журналист». – Минск, 2019. – Режим доступа: <https://tech.onliner.by/2017/10/10/gejmdizajner>. – Дата доступа: 09.10.2019.
2. Новостной портал Dev.by / Вот, значит, какую овцу заказал — все твои мысли сразу вскрылись. – Минск, 2019. – Режим доступа: <https://dev.by/news/vizor-humor>. – Дата доступа: 09.10.2019.
3. Сайт Парка Высоких Технологий Park.by / В бизнес-инкубаторе ПВТ прошел семинар, посвященный game dev – Минск, 2019. – Режим доступа: <http://www.park.by/post-2191/>. – Дата доступа: 09.10.2019.

Секция «Системы управления»

Section «Control systems»

СИСТЕМА УЧЕТА РАБОТЫ БАНКОВСКОГО ОБОРУДОВАНИЯ

Акинфина М. А., Бондаренко С. П.

Кафедра информационных технологий, Белорусский государственный экономический университет

Кафедра технологий программирования, Белорусский государственный университет

Минск, Республика Беларусь

E-mail: {akinфина}@rambler.ru, {svetabon}@gmail.com

Авторами рассмотрены проблемы, связанные с анализом и учетом функционирования банковского оборудования. Предложено приложение, автоматизирующее процесс учета оборудования.

ВВЕДЕНИЕ

Современные банковские организации располагают обширным парком оборудования и программного обеспечения, который постоянно увеличивается и обновляется. Поэтому возникает необходимость в наведении порядка и в процессе учета оборудования и ПО и в эффективном контроле и управлении их жизненным циклом [1].

Цель работы состоит в разработке приложения для автоматизации управления банковского оборудования, которое реализует следующие функции [2]:

- учет оборудования, комплектующих и программного обеспечения;
- документирование комплектации, движения и эксплуатации оборудования, запасных комплектующих и программного обеспечения;
- автоматизация контроля выполнения ремонтных, модернизационных и регламентных работ, ведение истории работ;
- ведение истории эксплуатации оборудования;
- формирование отчетов по эксплуатации и размещению оборудования и программного обеспечения;
- определение эксплуатационных характеристик и характеристик надежности оборудования на основании истории эксплуатации.

I. ПРОЕКТ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Используемая система учета позволяет учитывать оборудование любого типа, в том числе и компьютерное оборудование, а также специальное программное обеспечение, установленное на нем [3].

В рамках жизненного цикла, состояние устройства (компьютера) описывается следующими характеристиками:

- расположение – одно из заданных в системе мест расположения;
- пользователь – один из заданных в системе сотрудников банка;
- состояние: резерв (устройство не эксплуатируется, но работоспособно, эксплуатация (устройство эксплуатируется), ремонт (устройство не эксплуатируется и неработоспособно), списано (устройство не эксплуатируется).

Различные комбинации значений характеристик определяют различные состояния устройства в жизненном цикле.

В процессе эксплуатации оборудование меняет состояние работоспособности. Устройство (компьютер) может находиться в одном из двух состояний: работоспособно и неработоспособно.

Состояние неработоспособно разбивается на несколько состояний: ожидание технического персонала, поиск неисправности, восстановление работоспособности.

Из состояния эксплуатации возможно два выхода: состояние ремонта – если устройство потеряло работоспособность и его пришлось заменить, и состояние резерва – если устройство было заменено, не потеряв работоспособности.

Разработанное приложение реализует следующие основные функции:

- документирование комплектации, движения и эксплуатации оборудования, запасных комплектующих и программного обеспечения;
- постановка на учет закупленного оборудования, комплектующих и ПО.
- организация (документирование) ремонта;
- формирование отчетов по эксплуатации и размещению оборудования и программного обеспечения;
- определение эксплуатационных характеристик и характеристик надежности оборудования.

При этом регистрируются следующие реквизиты:

- оборудование (тип, наименование, состояние, поставщик, срок гарантии, инвентарный номер, размещение, пользователь, комплектность (аппаратный и программный состав), перечень выполненных работ, список документов по данному оборудованию, сбои, отказы, восстановление работоспособности);
- комплектующие (тип, наименование, производитель, состояние, технические характеристики, срок гарантии, инвентарный номер, пользователь);
- программное обеспечение (тип, наименование, номер версии, производитель, язык интерфейса, ключ авторизации, пользователя);

– пользователи (фамилия, имя, отчество, отдел, должность, номер телефона и электронный адрес, используемое оборудование и программное обеспечение, перечень документов).

При учете движения оборудования, комплектующих и ПО выполняется ряд операций: регистрация значений учетных реквизитов оборудования на этапах жизненного цикла: поступления, ввода и вывода из эксплуатации, проведения модернизационных и ремонтных работ, перемещения, списания; создание формуляра для каждой единицы оборудования; документарный метод учета. Изменения учетных данных о движении, ремонте и техническом обслуживании оборудования регистрируются в программе только посредством оформления документов.

Документы могут быть представлены в печатном виде; автоматическая синхронизация данных о количестве и номенклатуре запасных комплектующих на складе при поступлении новой партии, выдачи со склада для выполнения ремонтных и модернизационных работ, возвращения на склад исправных и списании неисправных комплектующих.

При выполнении технического обслуживания ведутся истории ремонтных работ в разрезе пользователей, компьютеров, комплектующих и оргтехники.

Информация по эксплуатации оборудования может быть систематизирована и представлена в форме отчетов: о составе оборудования; о движении оборудования, комплектующих и ПО; об истории эксплуатации оборудования; о созданных документах.

Структурно приложение включает следующие основные объекты: справочники, документы, регистры сведений, регистры накопления, отчеты и формы.

Справочники служат для хранения первичной справочной информации, которую используют для регистрации операций с оборудованием и программным обеспечением.

Документы содержат все учетные данные о движении, ремонте и техническом обслуживании и регистрируются в программе посредством оформления документов. Пользователь заполняет поля документа, после чего запускает процедуру проведения регистрации, в которой выполняется основная работа по регистрации данных.

Основная задача регистра сведений – хранить существенную для прикладной задачи информацию, состав которой развернут по определенной комбинации значений и, при необходимости, развернут во времени.

Регистры накопления накапливают информацию о движении регистрируемых объектов в терминах приход-расход. Они позволяют быстро

получить актуальную информацию об их наличии (остатке) или оборотах за период.

Отчеты используются для получения сводной информации на основании данных, введенных в систему. Отчет содержит алгоритмы получения и обработки информации, формы, при помощи которых организуется интерфейс для ввода параметров алгоритмов и представления результатов их работы, а также макеты, используемые для вывода результатов выполнения алгоритма в табличный документ.

Формы служат для организации интерфейса пользователя, а именно, для отображения информации и получения информации от пользователя с помощью соответствующих элементов управления.

II. ЗАКЛЮЧЕНИЕ

Результатом работы является приложение для автоматизации учета оборудования, которое позволяет сократить затраты на обслуживание оборудования, управляя всем его жизненным циклом.

Результаты работы приложения являются не просто статической информацией, а отображением состояния оборудования в режиме реального времени.

Наряду с удобством эксплуатации и повышением эффективности работы всех подразделений компании, внедрение такого приложения является экономически выгодным вложением финансовых средств.

Данное приложение легко интегрируется с такими системами как «1С:Предприятие 8», «Галактика ERP». Поэтому может быть использовано для автоматизации управления не только банковским оборудованием, но и оборудованием предприятия [4].

1. Управление основными фондами: как автоматизировать ремонты и техническое обслуживание [Электронный ресурс]. – Режим доступа: <http://www.cio-world.ru>. – Дата доступа: 25.04.2019.
2. Акинфина, М.А. Система анализа и учета функционирования оборудования // М.А. Акинфина, С.П. Бондаренко. / Вести Института современных знаний. –2012. –№3(52). – С.95–102.
3. Акинфина, М.А. Контроль и управление в системах анализа и учета оборудования / М.А. Акинфина, С.П. Бондаренко. // Информационные технологии и системы 2012 (ИТС 2012): материалы международной научной конференции, БГУИР, Минск, Беларусь, 24 октября 2012 г. / – Минск:БГУИР, 2012. – С.24–26.
4. Акинфина, М.А. Автоматизация процесса учета оборудования предприятия / М.А. Акинфина, С.П. Бондаренко. // Проблемы современной экономики: глобальный, национальный и региональный контекст: Сборник научных статей, ГрГУ, Гродно, 2013 г.: В 2 ч. / Минск:ГрГУ, 2013. –Ч.1. – С.321–327.

АЛГОРИТМИЧЕСКОЕ РЕШЕНИЕ ОБРАТНОЙ ЗАДАЧИ ДИНАМИКИ ДЛЯ ПОДАВЛЕНИЯ КОЛЕБАНИЙ

Алешко Е. Ю., Хаджинов М. К.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: km_h@tut.by, yaleshko2@gmail.com

Опробовано алгоритмическое решение обратной задачи динамики для подавления колебаний. Обратная задача заключается в определении сил, приложенных в исполнительных механизмах для обеспечения заданной траектории их движения. Показана высокая эффективность модального управления электроприводами с нежесткой механикой по сравнению с традиционными регуляторами и способами управления.

ВВЕДЕНИЕ

Возможность получения систем автоматического управления (САУ), обладающих высокой чувствительностью к параметрическим возмущениям, при решении обратных задач динамики в частотной области методами модального управления (МУ), является известным фактом, поэтому вопросам обеспечения параметрической грубости таких систем в настоящее время уделяется достаточно пристальное внимание.

Известно, что чувствительность систем методами модального управления значительно возрастает в том случае, когда в статическом регуляторе состояния (РС) присутствуют положительные обратные связи.

В некоторых случаях оценить возможность получения САУ с высокой чувствительностью к параметрическим возмущениям и попытаться уменьшить ее можно еще на стадии составления математической модели объекта управления (ОУ) и задания желаемого характера движения замкнутой системы. Так, например, исключение из описания объекта полюсов, определяющих процессы более быстрые, чем обусловленные полюсами желаемого характеристического полинома, позволяет значительно повысить грубость замкнутой системы.

Опыт показывает, что особенно сильное влияние на чувствительность САУ, синтезированных методами МУ, может оказывать наличие нулей в передаточной функции ОУ от входа к управляемой координате. Наличие нулей ограничивает возможное, с точки зрения обеспечения параметрической грубости, быстродействие замкнутой системы.

Существующие системы управления автоматизированными приводами динамических систем не всегда могут обеспечить стабильность технологического процесса. Колебательные процессы имеют случайный характер и, как правило, очень зашумлены, поэтому электромеханические динамические системы в таких случаях должны быть малочувствительными к изменению основных характеристик элементов элек-

тропривода (момента инерции, коэффициента жесткости) в достаточно широком диапазоне, определяемом особенностями технологического процесса.

I. ЦЕЛЬ ИССЛЕДОВАНИЯ

Одним из наиболее перспективных методов синтеза управляющих устройств, используемых в системах автоматизации, является модальное управление, предоставляющее разработчику широкие возможности в обеспечении требуемых показателей качества протекания процессов: повышение быстродействия, устранение колебательности при наличии упругости и люфтов в кинематических связях. В связи с этим в последнее время вопросы анализа и синтеза систем модального управления (СМУ) электроприводами становятся все более актуальными, им уделяется большое внимание [1]. В зависимости от порядка системы таких коэффициентов может быть достаточно много, что повлечет за собой как сложность и громоздкость формул, так и последующих расчетов, а для высокого порядка системы — невозможность формирования такого аналитического представления, которое можно использовать в расчетах.

Несмотря на свой значительный потенциал, модальный метод проектирования регуляторов в настоящее время нечасто применяются в инженерной практике, поскольку:

— чаще всего реальные объекты нелинейные и в отдельных случаях даже не имеют постоянной структуры, и применение модального метода, изначально предназначенного для синтеза линейных систем, в разработке систем управления такими объектами до сих пор оставалось проблематичным;

— параметры модальных регуляторов (коэффициенты обратной связи) не несут достаточно очевидной смысловой нагрузки и потому при «уходе» параметров объекта не могут быть настроены непосредственно разработчиком, как в случае с традиционными регуляторами;

— будучи аналитическим, метод модального синтеза в большей степени, чем классические методы, зависим от точности модели системы. Применение методов модального управления в решении задач электроприводной техники — сравнительно молодое направление, применение в этой области различных интеллектуальных инструментов встречается довольно редко.

II. МОДАЛЬНЫЙ РЕГУЛЯТОР ЭЛЕКТРОПРИВОДА С НЕЖЕСТКОЙ МЕХАНИКОЙ

В рамках развития исследовательской работы [2] выполним синтез модального регулятора (МР) для двухмассового электропривода с упругим механическим звеном. Упругое звено с коэффициентом жесткости связывает две массы: ротора и рабочего органа с моментами инерции J_1 и J_2 . В подобной системе при определенном соотношении параметров возможен резко колебательный характер движения. Задача управления: обеспечение заданных показателей в статике и динамике, обеспечение устойчивости и стабилизации движения. Применение модального управления предоставляет разработчику широкие возможности в обеспечении требуемых показателей качества протекания процессов и устойчивости систем управления.

Главные преимущества модального управления электроприводом: синтезированная система модального управления объектом уже удовлетворяет требуемым показателям качества, не требует введения дополнительных корректирующих устройств и проверки на устойчивость.

Расчёты демпфирующего модального регулятора произведём для электропривода рольганга с нежесткой механикой широкополосного стана горячей прокатки мощностью 25 квт, 1000 об/мин [3], имеющий частоту крутильных колебаний нагрузки приблизительно 40 1/С. Разработанный авторами методика проектирования модального регулятора позволяла уменьшить показатель колебательности электропривода с 9 до 1.3. Т.е. авторам не удалось решить задачу кординально и свести показатель колебательности к единице. Нами для модели электропривода 5-го порядка в пространстве состояний были использованы стандартные операторы Матлаба для расчёта модальных регуляторов. В качестве желаемого характеристического полинома использовался полином с аппериодическими переходными характеристиками и перерегулированием 0.02 процента. Чтобы минимизировать механические нагрузки на электропривод, полином масштабировался частотой крутильных колебаний нагрузки. Расчёты производились для ши-

рокого диапазона отношений моментов инерции нагрузки J_2 и электродвигателя J_1 . Моделировались процессы по управлению и возмущению. Результаты моделирования электроприводов без модального регулятора и с демпфирующим модальным регулятором для соотношения $J_2/J_1 = 0.5$ приведены на рис.1. На рис.1 отображены графики скоростей нагрузки, электродвигателя и угла закручивания гибкого вала связи двигателя и нагрузки. Как видно из графиков, модальный регулятор полностью устраняет колебательность нагрузки по управлению и возмущению.

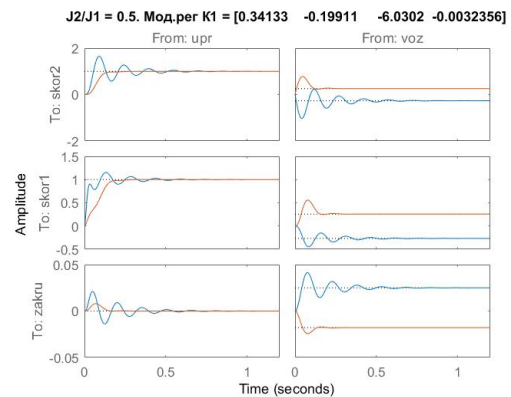


Рис. 1 – Отношение моментов инерции нагрузки J_2 и двигателя J_1

III. ЗАКЛЮЧЕНИЕ

Было опробовано алгоритмическое решение обратной задачи динамики для подавления колебаний. Показана целесообразность расчёта модального регулятора стандартными средствами Матлаб. Установлена высокая эффективность модального регулятора как демпфера крутильных колебаний двухмассового электромеханического объекта управления с гибкой кинематической связью. Полученные результаты способствуют совершенствованию электроприводов большой мощности в большинстве случаев имеющие нежесткую механику.

СПИСОК ЛИТЕРАТУРЫ

1. Тарарыкин С.В., Тютиков В.В. Определение размерности вектора состояния при синтезе управляемых динамических систем // Изв. вузов. Электромеханика, 1995, №1-2, С. 69-74.
2. Тарарыкин С.В., Тютиков В.В. Робастное модальное управление динамическими системами // Автоматика и телемеханика. — 2002. — № 5. — С. 41–55
3. Кочнева Т.Н., Кожевников А.В., Кочнев Н.В. Синтез модального регулятора и оценка эффективности модального управления для двухмассовых электромеханических систем / Т.Н. Кочнева, А.В. Кожевников, Н.В. Кочнев // Вестник Череповецкого госуниверситета. — 2013. - №4. — С. 15 – 22

DEVELOPMENT OF A ROBUST AIRCRAFT CONTROL SYSTEM IN CONDITIONS OF DISTURBANCES

Amirzhanova Z. B., Mashtaeva A. A., Satybaldina D. K.

Department of Systems Analysis and Management, Eurasian National University named after L.N. Gumilyov
Nur-Sultan, Kazakhstan

E-mail: zinara_amir@mail.ru, mashtayeva@mail.ru, satybaldinad@mail.ru

In this paper a mathematical model of the dynamics of an airplane in a vertical plane is developed. The features of the robust structural synthesis procedure as applied to the system of the type under study are presented. The optimization problem for the system under study was solved using the mixed sensitivity method. The modern approach to solving the problem of robust structural optimization is based on the formation of the desired frequency characteristics of the system by expanding the object by introducing weighted transfer functions.

INTRODUCTION

The problem of ensuring quality control of the aircraft, especially in the case of a disturbed atmosphere, is currently relevant. To solve this problem, robust regulators based on H_∞ – control theory are used. In recent years, methods for the synthesis of optimal robust controllers have been one of the central issues of control theory, which remains somewhat unresolved [1–3].

I. MATHEMATICAL DESCRIPTION OF THE CONTROL SYSTEM

Consider the dynamics of the aircraft in a vertical plane. The dynamic airplane model in the vertical plane is described by a 5th order equation with phase coordinates: x_1 – relative height, m , x_2 – translational speed, m/s , x_3 – pitch angle, $deg.$, x_4 – pitch angular velocity, $deg./s$, x_5 – vertical speed, m/s . The first three coordinates are measured. The control variables are: u_1 – spoiler angle, $deg. \times 0.1$, u_2 – translational acceleration, m/s^2 , u_3 – elevator angle, $deg.$

Taking into account the parameters of the aircraft, as well as the moments of inertia and their derivatives, the matrix coefficients are obtained:

$$A = \begin{pmatrix} 0 & 0 & 1.1320 & 0 & -1 \\ 0 & -0.0538 & -0.1712 & 0 & 0.0705 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0.0485 & 0 & -0.8656 & -1.0130 \\ 0 & -0.2909 & 0 & 1.053 & 0.6859 \end{pmatrix}$$

$$B = \begin{pmatrix} 0 & 0 & 0 \\ -0.12 & 1 & 0 \\ 0 & 0 & 0 \\ 4.42 & 0 & -1.665 \\ 1.575 & 0 & -0.0732 \end{pmatrix},$$

$$u = (u_1, u_2, u_3).$$

The system has poles of $-0.78 \pm 1.03j$, $-0.0176 \pm 0.1826j$, 0 . There are no zeros.

II. RESEARCH METHODS

Consider the method of weight functions, which allows you to improve the efficiency of the

control system [4–7]. This method is described below.

Let $z(s) = W(s)y(s)$, where $W_z(s)$ – is the matrix weight function, which can be selected depending on the characteristics of the systems. The block diagram of the system is shown in Figure 1.

The transfer function matrix T_{wz} of the closed-loop system has the form [6,7]:

$$T_{wz} = W_z(1 - GK)^{-1}W$$

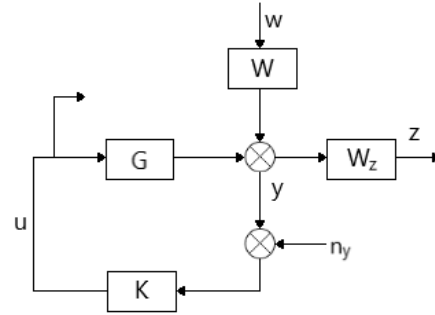


Fig. 1 – Block diagram of a system with weight functions

It can be noted that weight functions allow you to set requirements in the frequency domain for the characteristics of a closed system in terms of the quality of transients and robust stability [4,6].

Based on the flowchart (Figure 1), we can write $T_{wz} = W_z T_{wy}$, if W_z is reversible correct. Then it turns out $T_{wy} = W_z^{-1} T_{wz}$, and then

$$\|T_{wy}\|_\infty \leq \|W_z^{-1}\| \|T_{wz}\|_\infty$$

or

$$\|T_{wy}\|_\infty \leq \gamma_{min} \|W_z^{-1}\|_\infty \quad (1)$$

if W_z is a diagonal matrix and has the form:

$$W_z = \begin{bmatrix} W_{ze} & 0 & \dots \\ 0 & W_{ze} & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

where W_{ze} – correct stable transfer function, from (1) it follows:

IV. FINDINGS

$$\bar{\sigma}(T_{wy}(jw)) \leq \gamma_{min} |W_{ze}^{-1}|, \forall w, \quad (2)$$

where $\bar{\sigma}$ – maximum singular number, γ_{min} – optimality level.

Inequality (2) is considered the main one for choosing W_{ze} (as well as W_z).

III. RESEARCH RESULTS

Weighted frequency functions are selected as follows [8–9]:

$$W_1 = \frac{1}{s + 0.01} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 10(0.02s + 1) & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

$$W_3 = \frac{s^2}{k} \text{diag}(3)$$

Graphs of singular values of SV and frequency characteristics are shown in Figure 2.

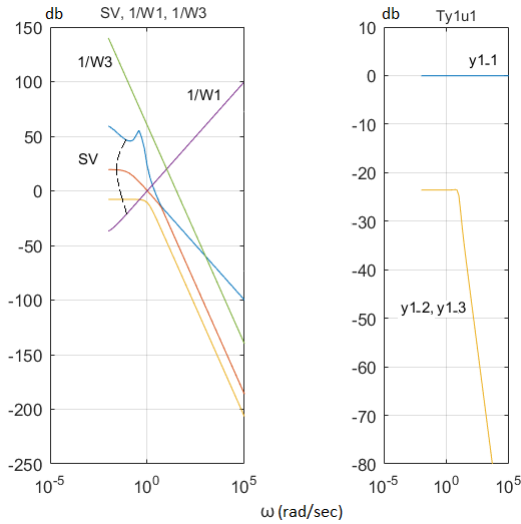


Fig. 2 – Graphs of singular values of SV and frequency characteristics

Figure 3 shows the graphs of transients with a step change in the task at each input. It can be seen that the processes are well damped, and the mutual influence of the channels is absent.

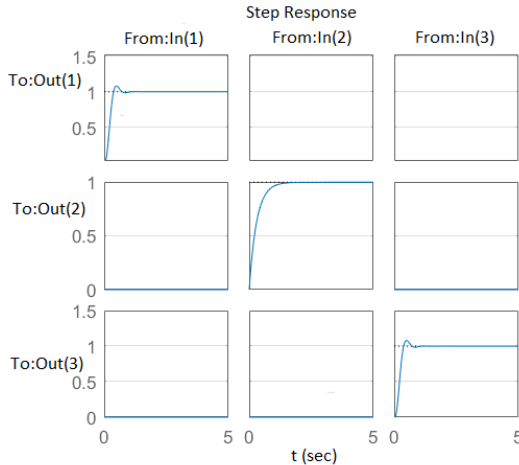


Fig. 3 – Transients of an aircraft in a vertical plane

The paper presents the main approaches to the robust structural synthesis of an aircraft control system. A mathematical description of the aircraft dynamics control system in the vertical plane is obtained taking into account wind disturbances. The movement of the aircraft in the vertical plane is a special case of the longitudinal movement of the aircraft and is described by a system of fifth-order equations. We consider the method of weight functions, which helps to improve the operability of the aircraft control system in conditions of wind disturbances. Matrices of weight transfer functions are selected that provide requirements for robust stability and the quality of transients to the characteristics of the system under study in the frequency domain. The effectiveness of the proposed method is confirmed by the results of modeling a synthesized system.

V. REFERENCES

1. Bulychev Yu. G. [i dr.] *Matematicheskiye aspekty opredeleniya dvizheniya LA.* – M.: 2000.
2. Polyak B. T., Shcherbakov P. S. *Robastnaya ustoychivost i upravleniye.* – M.: Izd-vo RAN Institut problem upravleniya. 2002. – 273 s.
3. Satybaldina D. K., Mashtayeva A. A., Smailov E. *Development of an Evaluation System of Orientation Angles of Maneuver Objects.* – *Engineering Computations*, No. 8(2). Volume 35. Emerald Group Publishing Ltd., 2018. – p. 3204–3214. – ISSN 0264-4401
4. Ali R. S. *Sintez robastnykh regulyatorov stabilizatsii transportnykh sredstv.* – Sankt-Peterburgskiy Gosudarstvennyy Politekhnikeskiiy universitet. – Sankt-Peterburg. 2002. – 164 s.
5. Gu D. W., Petkov P. Hr., Konstantinov M. M. *Robust control design with MATLAB.* London: Springer-Verlag, 2005. – 576 p.
6. Sushchenko O. A., Azarskov V. N. *Proyektirovaniye robastnykh sistem stabilizatsii oborudovaniya bespilotnykh letatelnnykh apparatov // Vestnik Samarskogo gosudarstvennogo aerokosmicheskogo universiteta.* – 2014. – № 1(43). – s. 80–90.
7. *Robust Control Toolbox. User's Guide.* The MathWorks. 2001. – 655 p.
8. Satybaldina D. K., Mashtayeva A. A., Zekenova G., Intymakov T. *Synthesis of Robust Control System for Aircraft // The VIII International Academic Congress "Fundamental and Applied Studies in EU and CIS Countries".* – United Kingdom, Cambridge, England, 2018. – p. 129–138.
9. Mashtayeva A. A., Amirzhanova Z. B., Satybaldina D. K. *Development of aircraft dynamics model in the vertical plane.* – *Informatsionnyye tekhnologii v nauke. upravlenii. sotsialnoy sfere i meditsine: sbornik nauchnykh trudov V Mezhdunarodnoy nauchnoy konferentsii.* – Tomsk. 2018. – Chast 1. – p. 18–21. – ISBN 978-5-4387-0845-2 (ch.1)

РОБАСТНАЯ УСТОЙЧИВОСТЬ СИСТЕМЫ С ОДНИМ ВХОДОМ И ОДНИМ ВЫХОДОМ В КЛАССЕ КАТАСТРОФ «ЛАСТОЧКИН ХВОСТ»

Ахметова С. О.

Кафедра системного анализа и управление, Евразийский национальный университет им. Л.Н. Гумилева

Нур-Султан, Казахстан

E-mail: saule_akhmetova_79@mail.ru

В статье предлагается новый подход к построению систем управления для объектов с неопределенными параметрами в форме трехпараметрических структурно-устойчивых отображений из теории катастроф, позволяющей синтезировать высокоэффективные системы управления, обладающие предельно широкой областью робастной устойчивости. Исследование робастной устойчивости систем управления базируется на новом подходе к построению функций А. М. Ляпунова. Излагается метод построения системы управления с повышенным потенциалом робастной устойчивости.

ВВЕДЕНИЕ

Проблема исследования устойчивости занимает одно из центральных мест при создании систем управления техническими объектами и технологическими процессами, которые широко применяются практически во всех сферах производства и техники: в машиностроении, энергетике, электронной, химической, биологической, металлургической и текстильной промышленности, транспорта, робототехнике, авиации, космических системах, высокоточной военной технологии и технике и т. д. В настоящее время общепризнано, что большинство реальных систем управления функционируют в условиях той или иной степени неопределенности. При этом неопределенность может быть обусловлена незнанием истинных значений отдельных параметров объекта управления и непредсказуемым изменением их во времени. Поэтому исключительно важную роль в теории управления динамическими объектами играет робастная устойчивость.

I. РОБАСТНЫЕ СИСТЕМЫ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Повышение эффективности работы технологических машин и установок требует применения современных систем управления. Одним из перспективных направлений совершенствования систем управления является робастное управление. Построение регуляторов для реальных физических объектов, описываемых линейными и нелинейными уравнениями, сопряжено с рядом значительных трудностей. Они обусловлены сложной природой изучаемых объектов, воздействием на них неконтролируемых внешних факторов, неполнотой априорной информации [1–3] и, в частности, неопределенностью параметров объекта управления и системы [4–8]. Существует много причин, которыми можно объяснить параметрическую неопределенность. При наладке системы управления приходится допускать некоторую неточность ее работы, например, из-за конечной точности изготовления ее компонент.

Проблема робастности на сегодняшний день является важнейшей проблемой в теории управления.

Целью синтеза робастной системы является гарантия требуемого качества независимо от погрешностей и изменения параметров модели. Система, обладающая свойством устойчивости и допустимыми изменениями качества при изменениях или неточности ее модели, называется робастной. От робастной системы требуется, чтобы она обладала низкой чувствительностью, сохраняла устойчивость и удовлетворяла требованиям, предъявляемым к ее качеству, в достаточно большом диапазоне изменения ее параметров. Робастность по сути дела характеризуется чувствительностью системы к факторам, которые не учитываются на этапах анализа и синтеза – например, к возмущениям, шуму датчика и не отраженным в модели системы параметрам, влияющими на ее динамику. Система должно быть способна противодействовать влиянию этих факторов при выполнении задач, ради которых она проектировалась. При этом введен термин «робастность», как правило, понимающий как способность сохранять устойчивость системы в условиях параметрической или непараметрической неопределенности в описании объекта управления. Но работоспособность системы характеризуется не только способностью сохранять устойчивость при изменении параметров в допустимой области, но важно и то, что показатели качества управления также не должны выходить за пределы некоторой допустимой области. При этом основное внимание уделяется проблеме робастных показателей качества управления.

II. КАТАСТРОФА «ЛАСТОЧКИН ХВОСТ» В КАЧЕСТВЕННОЙ ТЕОРИИ ДИНАМИЧЕСКИХ СИСТЕМ

Теория катастроф – часть качественной теории сложных нелинейных систем. Её основой является теория особенностей гладких (диффе-

ренцируемых) отображений, сформировавшаяся на стыке топологии и математического анализа, и являющаяся обобщением задач на экстремум в математическом анализе. Элементарная теория катастроф сводит огромное многообразие ситуаций к небольшому числу стандартных схем, которые можно детально исследовать. Анализ качественного поведения нелинейных динамических систем при изменении описывающих их параметров, позволяет описывать состояния, далёкие от равновесия, а также предсказывать резкую смену этих состояний. Теория катастроф – программа прогнозирования неустойчивости различных систем. Такое название она получила потому, что потеря устойчивости может быть катастрофична, даже если не приводит к гибели или разрушению системы, а лишь обуславливает переход к иной траектории развития. Катастрофа – скачкообразное изменение, возникающее в виде внезапного отклика системы на плавное изменение внешних условий, т.е. резкое качественное изменение объекта при плавном количественном изменении параметров, от которых он зависит. Теория катастроф – раздел математики, включающий в себя теорию бифуркаций, дифференциальных уравнений (динамических систем) и теорию особенностей гладких отображений. Одной из главных задач теории катастроф является получение нормальной формы исследуемого объекта (дифференциального уравнения или отображения) в окрестности «точки катастрофы» и построение классификации объектов. Теория бифуркаций динамических систем – теория, изучающая изменения качественной картины разбиения фазового пространства, в зависимости от изменения параметра (или нескольких параметров). Бифуркация – приобретение нового качества в движениях динамической системы при малом изменении её параметров. Бифуркация – раздвоение, разделение, разветвление чего-либо. Состояние процесса в динамической системе, при котором резко возрастают флуктуации, и выход из которого возможен по двум существенно различным трудно предсказуемым направлениям – хаотическому или упорядоченному. Динамическая система – математическая абстракция, предназначенная для описания и изучения эволюции систем во времени. Представляет собой математическую модель некоторого объекта, процесса или явления. Рассмотрим катастрофу ласточкин хвост. $x^5 + k^3x^3 + k^2x^2 + k^1x$ –катастрофа ласточкин хвост (swallowtail). графическое изображение в пространстве показано на рисунках 1.

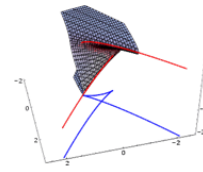


Рис. 1 – Катастрофа ласточкин хвост (swallowtail)

III. Выводы

В теории управления в последнее время получены исследования изменений поведения систем при больших конечных изменениях параметров и синтез законов управления в условиях большой неопределенности в знании свойств объекта, понимаемый как робастность, т. е. как способность системы сохранять устойчивость в условиях параметрической или непараметрической неопределенности в описании объекта управления. Важным прикладным результатом качественной теории динамических систем являются структурно устойчивые отображения и устойчивость соответствующей топологической структуры. Приведены, как основной прикладной результат теории катастроф, структурно устойчивых отображений и изложен обоснованный формализм анализа качественных изменений в динамических системах. Поставлена задача анализа и синтеза систем управления с повышенным потенциалом робастной устойчивости для динамических объектов с подходом к синтезу систем управления в классе функций катастроф ласточкин хвост, придающих системе управления свойство наибольшей робастной устойчивости.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Поляк, Б. Т., Щербаков П. С. Робастная устойчивость и управление. – М.: Наука, 2002. – 303 с.
2. Safonof, M. G. Stability and robustness of multivariable feedback systems. – Cambridge, MA: MIT Press, 1980.
3. Chen, M. J., Desoer, C. A. Necessary and sufficient for robust stability of linear distributed feedback systems // Intern. J. Control. – 1982. – V.35, No.2. – P.255–267.
4. Бесекерский, В. А., Небылов, А. В. Робастные системы автоматического управления. – М.: Наука, 1983. – 239 с.
5. Gregoire Nicolis, Ilya Prigogine. Exploring Complexity an Introduction. – New York (1989).
6. Постон, Г., Стюарт, И. Теория катастроф и ее приложения. –М.: Мир, 1980.
7. Гилмор, Р. Прикладная теория катастроф. Т.1. –М.: Мир, 1981.
8. Бейсенби, М. А. Методы повышения потенциала робастной устойчивости систем управления. – Астана, 2011. –352 с.
9. Бейсенби, М. А. Модели и методы системного анализа и управление детерминированным хаосом в экономике. – Астана, 2011. – 201 с.
10. Бейсенби, М. А., Ержанов, Б. А. Системы управления с повышенным потенциалом робастной устойчивости. – Астана, 2002. – 164 с.

АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС УЧЁТА ПОКАЗАТЕЛЕЙ ВОЗДУШНОЙ СРЕДЫ

Бысов С. А., Ивановская Я. М.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: s.bysov@mail.ru, yanaivanovskaya@mail.ru

Цель статьи заключается в описании аппаратно-программного комплекса обнаружения выбросов газов и учёта температуры и влажности воздушной среды. Особенное внимание уделено качеству и достоверности показателей, аргументированию выбора физических компонентов комплекса, а также энергосбережению.

ВВЕДЕНИЕ

Современный мир отличается высоким уровнем развития вычислительной техники, а также интеграции этой техники в различные сферы жизнедеятельности. Акцент делается на аспекте промышленного производства, а также сохранении и поддержании здоровья человека. При этом важнейшую роль в этих областях играет воздушная среда. Задачей данной статьи является описание процесса построения системы, которая бы позволяла осуществлять тщательный контроль за такими показателями воздушной среды, как температура, влажность и выбросы газов. Из этого следует, что полученная система нацелена на выполнение следующих функций: снимать показания датчиков о температуре и влажности, а также о выбросах газов метана, водорода и дыма, и отправлять собранные данные по сети интернет каждые 3-4 часа, или при повышении показателей воздушной среды выше допустимых пределов.

I. АППАРАТНАЯ ЧАСТЬ

Далее опишем необходимое аппаратное обеспечение для создания нашей системы.

В качестве основы для построения была выбрана плата Arduino Uno. Это объясняется тем, что данная плата имеет все необходимые интерфейсы взаимодействия с различными периферийными устройствами, в том числе с датчиками состояния воздушной среды. Кроме того, немаловажным фактом является её простота в использовании и написании программного кода. При сравнении с микроконтроллерами STM32 плата Arduino Uno может проигрывать в скорости взаимодействия, но в контексте решаемой задачи эти различия нивелируются.

Затем из всего множества датчиков газа, воздуха и температуры были выбраны наиболее подходящие нам. Модуль FC-22 предназначен для обнаружения изобутана, пропана, смесей сжиженных углеводородных газов (Liquefied petroleum gas (LPG)). Модуль построен на основе датчика MQ135. Датчик MQ135 относится к полупроводниковым приборам и может выяв-

лять NH_3 , NO_x , алкоголь, бензин, дым, CO_2 и некоторые другие газы, поэтому он удовлетворяет требования к системе контроля качества воздуха[2]. После подключения его к Arduino мы получили уровень загрязнения в PPM (parts per million). Принцип работы датчика основан на изменении сопротивления тонкопленочного слоя диоксида олова SnO_2 при контакте с молекулами определяемого газа, что изменяет уровень измеряемого напряжения, который интерпретируется в PPM. Чувствительный элемент датчика состоит из керамической трубки с покрытием Al_2O_3 и нанесенного на неё чувствительного слоя диоксида олова. Внутри трубки проходит нагревательный элемент, который нагревает чувствительный слой до температуры, при которой он начинает реагировать на определяемый газ. Чувствительность к разным газам достигается варьированием состава примесей в чувствительном слое[3].

Диапазон измерений датчика MQ135:

- Пропан: 200–5000 ppm
- Бутан: 300–5000 ppm
- Метан: 500–20000 ppm
- Водород: 300–5000 ppm

Для определения температуры и влажности воздуха был выбран датчик из семейства DHT, а именно DHT21. Цифровой датчик DHT21 является составным датчиком, который выдаёт каллиброванный цифровой сигнал с показаниями температуры и влажности. Сенсор включает в себя ёмкостной компонент измерения влажности и резистивный компонент измерения температуры с отрицательным температурным коэффициентом (NTC), которые подключены к высокопроизводительному АЦП. Каждый датчик DHT21 проходит калибровку на заводе изготовителе. Коэффициенты калибровки хранятся в однократно программируемой энергонезависимой памяти и используются во внутренних процессах обработки сигнала. Диапазон измерений датчика DHT21:

- Температура: $-40 \dots +80$ °C, погрешность: $\pm 0,3$ °C
- Влажность: 0–99,9 %, погрешность: ± 3 %

Для передачи данных по сети интернет был взят модуль NodeMCU. NodeMCU – плата на основе популярного модуля ESP8266. Модуль на чипсете ESP8266 – это простой и недорогой способ добавить в разрабатываемое устройство функцию беспроводной связи через Wi-Fi. Использование ESP8266 позволяет управлять устройством дистанционно или снимать показания с датчиков через интернет. Имеется возможность подключения гаджета к социальным сетям или реагирования на данные, получаемые через API от веб-сервисов. Управляющее устройство общается с ESP8266 через UART (Serial-порт) с помощью набора AT-команд[4]. Поэтому работа с модулем тривиальная для любой платы с UART-интерфейсом: возможно использование Arduino, Raspberry Pi, и т.д.

II. ПРОГРАММНАЯ ЧАСТЬ

В качестве средства разработки программного обеспечения аппаратно-программного комплекса использовалась Arduino IDE. Это позволило быстро прошивать плату Arduino Uno и при необходимости редактировать исходный код. Для передачи данных и представления их в удобном для пользователя виде было решено воспользоваться API популярного мессенджера Telegram и создать чат-бот. Это дает возможность пользоваться end-to-end шифрованием мессенджера, а также, при необходимости, добавлять различный функционал, так как разработка бота Telegram гораздо дешевле и быстрее, чем разработка собственного клиент-серверного приложения, и проще, чем работа с другими сервисами.

Общение между сервером Telegram и ботом происходит посредством передачи массива данных в формате JSON, в котором содержится подробное описание сообщения. Также при передаче данных указывается один из методов взаимодействия автономного аккаунта с пользователем[5].

Для создания автономного аккаунта необходимо зарегистрировать его в специальном боте, которого называют BotFather. При регистрации указываются: имя, отображаемое как основное, имя, по которому можно найти бота (так называемый “username”), а также описание. В username обязательно должна присутствовать метка “_bot”, для того, чтобы пользователи могли идентифицировать автономные аккаунты. После регистрации BotFather выдает токен. Это ключ, по которому можно обращаться к сервисам Telegram. На данном этапе регистрации можно считать завершенной и можно начать настраивать свой автономный аккаунт. Для удобства в аккаунте BotFather присутствуют различные настройки для зарегистрированного бота, включая настройки приватности, возможности включения бота в группу и множество других не менее важных возможностей[5].

После настройки можно воспользоваться двумя методами приема-передачи сообщения из/в автономный аккаунт: long pool и webhook. Различия между этими двумя методами достаточно значительны. В первом случае необходимо отправить запрос с методом GetUpdate на сервер Telegram для получения массива Update. В этом массиве находятся последние 500 сообщений (по умолчанию) полученных автономным аккаунтом. После получения необходимо просмотреть сообщения на предмет запросов. В случае с методом webhook Telegram переадресовывает каждое сообщение с аккаунта на сервер с программой управления ботом. И в этом случае появляются некоторые правила. Самое главное: принимающий сервер с программой должен иметь SSL-сертификат безопасности, для обеспечения работы по протоколу HTTPS. Вторым правилом, а точнее запретом, становится запрет на вызов метода GetUpdate, так как его функцию берет на себя Telegram[5].

Важная роль данных аккаунтов проявляется в том, что пользователь всегда и вовремя получает уведомления. И данное свойство как нельзя кстати подходит для их использования в связке с устройствами IoT. Говоря иными словами, можно приобрести устройства для управления домом/офисом/машиной и связать их с автономным аккаунтом. И постоянно быть в курсе событий. Более того, работа с ботом позволяет вести историю и статистику показаний датчиков.

III. ЗАКЛЮЧЕНИЕ

Таким образом, в итоге проведенной нами работы был спроектирован аппаратно-программный комплекс, который позволяет снимать показания с датчиков по заданной программе в реальном времени, и вести статистику показателей воздушной среды. Такой комплекс может найти применение в самом широком кругу областей: от интернет-вещей и элементов умного дома, до больших производственных решений, где нужен постоянный контроль над показаниями чистоты воздуха и выбросов различных газов.

IV. СПИСОК ЛИТЕРАТУРЫ

1. FC-22 [Электронный ресурс] – Режим доступа: <https://www.jahankitshop.com/getattach.aspx?id=3234&Type=Product>. – Дата доступа: 13.09.2019.
2. MQ-135 [Электронный ресурс] – Режим доступа: <https://www.olimex.com/Products/Components/Sensors/Gas/SNS-MQ135/resources/SNS-MQ135.pdf>. – Дата доступа: 13.09.2019.
3. DHT21 [Электронный ресурс] – Режим доступа: <https://kropochev.com/downloads/humidity/AM2301.pdf>. – Дата доступа: 13.09.2019.
4. ESP8266 NODEMCU [Электронный ресурс] – Режим доступа: https://www.handsontec.com/pdf_learn/esp8266-V10.pdf. – Дата доступа: 15.09.2019.
5. Telegram Bot API [Электронный ресурс] – Режим доступа: <https://core.telegram.org/bots/api>. – Дата доступа: 20.09.2019.

АНАЛИЗ И РАЗРАБОТКА МОДУЛЕЙ УЧЕБНОГО КУРСА ПО ПРОГРАММИРОВАНИЮ

Волков А. В.¹, Никифоров А. С.², Чижов И. В.³, Адасик А. В.⁴, Семенович А. С.⁴, Миранович Р. Б.⁴, Самардак Д. В.⁴, Кондратович М. М.⁴

¹Кафедра системного анализа и компьютерного моделирования, ²кафедра биофизики, ³кафедра физической электроники и нанотехнологий, ⁴кафедра информатики и компьютерных систем, Белорусский государственный университет
Минск, Республика Беларусь

E-mail: andrei@cybergizer.com, sky92033@live.com, fkknopsikk@mail.ru, anthonyadasiq@gmail.com, alexser1999@gmail.com, roman.miranovich@gmail.com, samardak.dasha31@gmail.com, minsk-maxim@mail.ru

В докладе представлено описание системы малых взаимодействующих друг с другом программируемых устройств (минироботов). Система призвана быть привлекательной для использования в факультативном учебном процессе старших классов средних общеобразовательных школ. Каждая программа для разрабатываемой платформы следует парадигме мобильного кода и имеет возможность перемещаться по сети. Набор минироботов позволяет обучающимся создать небольшую сеть, экспериментировать с этой сетью и реорганизовывать ее, что представляет возможность для изучения вычислительных процессов, а так же поведения алгоритмов, в группах.

ВВЕДЕНИЕ

Программируемые роботы постепенно становятся одной из значительных технологий в образовательном процессе 21-го века и используются для улучшения образования как в формальной так и в неформальной среде [1]. Данные роботы могут быть поделены на две категории: системы из множества независимых роботов [2,3] и одиночные роботы. Последняя категория образовательных роботов является наиболее традиционной и широкопротсраненной (одним из наиболее репрезентативных представителей является Bee-Bot [1]), однако не позволяет исследовать те процессы и явления, изучение которых возможно лишь при наличии множества независимых устройств.

I. ОПИСАНИЕ ПОДХОДА

В докладе представлено описание системы малых взаимодействующих друг с другом программируемых автономных объектов (в дальнейшем, минироботов). Данные минироботы призваны играть в учебном процессе роль как абстрактных, так и физических объектов. Основная идея, воплощенная в системе, заключается в том, что программа может существовать внутри сети, передаваясь от миниробота к минироботу. Программы — это отдельные логические объекты, так же как сами минироботы — это физические объекты. Программы переходят от миниробота к минироботу и могут взаимодействовать с другими программами, с которыми они встречаются во время перемещения по сети. Каждый миниробот представляет собой независимый, физический, осязаемый объект (недорогое и компактное устройство размером меньше ладони), который можно перемещать (рис. 1). Каждый миниробот может связываться с соседними минироботами, расположенными вдоль четырех сторон

света (направления Север, Восток, Юг и Запад). Минироботы могут быть организованы в двумерные узоры, в которых соседние минироботы связаны как логически так и физически (лишь смежные минироботы могут общаться). В итоге обучающиеся могут взаимодействовать с системой двумя способами: путем написания программ и путем реорганизации протсранственной конфигурации минироботов.



Рис. 1 – Четыре миниробота

II. ОПИСАНИЕ РЕАЛИЗАЦИИ

Представленный в работе подход является логическим продолжением подхода описанного в работе [2]. В работе [2] для реализации локальной коммуникации между автономными устройствами выбрано использование инфракрасного

канала (ИК) связи. В настоящей работе использован комплексный подход на основании использования инфракрасного и радио каналов связи. В итоге для определения соседних роботов разработан ИК модуль обнаружения соседей (рис. 2).

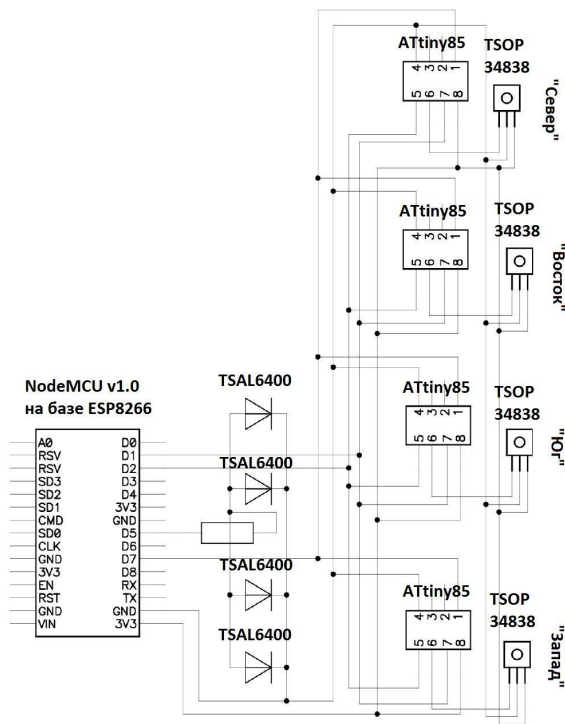


Рис. 2 – Принципиальная схема модуля обнаружения соседа

Передача идентификационного номера осуществляется по NEC протоколу [4]. При этом используется ИК диод TSAL6400 подключенный к цифровому выводу. Для приема модулированного ик сигнала используются фотоприемники TSOP34838 преобразующих принятый сигнал в цифровой. Микроконтроллер Attiny85 осуществляет декодирование цифрового сигнала с сохранением значения в буфер. Приемники и передатчики размещены в направлениях Север, Восток, Юг и Запад для обеспечения покрытия четырех сторон света (рис. 2). Использование четырех микроконтроллеров Attiny85 позволяет вести прием ИК сигнала во всех направлениях одновременно.

Блок-схема программируемого минибота представлена на рис. 3. ИК модуль подключен по I2C интерфейсу к головному контроллеру на базе системы на кристалле ESP8266 (считывание информации по I2C интерфейсу происходит по прерываниям). Вынесение обработки принимаемого ИК сигнала в отдельную систему позволяет снять данного рода вычислительную нагрузку с ESP8266 и обеспечить модульность. Использование ИК модуля позволяет осуществлять передачу идентификаторов в сети (MAC адресов) посредством NEC протокола, что представляет собой основу для передачи данных основного потока данных посредством протокола ESP-

Now [5]. Подсистема электроснабжения минибота реализована на базе повышающего DC/DC конвертора и Li-ion аккумулятора. Система световой индикации реализована с использованием четырех адресных светодиодов типа WS2812B.



Рис. 3 – Блок-схема программируемого минибота

III. ЗАКЛЮЧЕНИЕ

Предложена система малых взаимодействующих друг с другом программируемых автономных минироботов, которая предназначена для использования в качестве инструмента для обучения думать о результатах взаимодействия многочисленных независимых процессов. Возможность локальной коммуникации отдельных минироботов обеспечивается посредством передачи основного потока данных посредством протокола ESP-Now и передачи идентификаторов в сети (MAC адресов) посредством NEC протокола. Результирующая система позволяет реализовывать сложные взаимодействия с участием множества минироботов и нескольких программ. Направления будущих исследований включают разработку учебного курса на базе предложенной системы.

СПИСОК ЛИТЕРАТУРЫ

1. Caballero Gonzalez, Yen, García-Valcárcel, Ana. (2017). Educational robotics for the formation of programming skills and computational thinking in childish. 1-5. 10.1109/SIIE.2017.8259652.
2. Kramer, Kwindla "Moveable Objects, Mobile Code" Cambridge, MA: Master Thesis, Massachusetts Institute of Technology, 1998.
3. Özgür, Ayberk (2018). Cellulo: Tangible Haptic Swarm Robots for Learning (PhD). EPFL. doi:10.5075/epfl-thesis-8241
4. Novakovic, Djordje and Sovilj, Platon and Gazivoda, Nemanja. (2017). Multisensory platform based on NEC protocol. Serbian Journal of Electrical Engineering. 14. 23-33. 10.2298/SJEE1701023N.
5. ESP-Now Overview [Electronic resource]: description of ESP-Now protocol at Espressif Systems website – Mode of access: <https://www.espressif.com/en/products/software/esp-now/overview>. – Date of access: 10.10.2019.

ПРОГРАММНОЕ СРЕДСТВО АВТОМАТИЗАЦИИ ПРОЦЕССА ДИНАМИЧЕСКОГО ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ С РАСПОЗНАВАНИЕМ НОМЕРОВ НА ОСНОВЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ

Деменковец Д. В., Дубицкая К. Ю., Борисевич А. Н.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: Demenkovets@bsuir.by, dubitskaya.karalina@gmail.com, alexwelgum@gmail.com,

В докладе рассматривается состав программно-аппаратного комплекса, программное средство автоматизации процесса динамического взвешивания железнодорожных вагонов. Также описываются особенности алгоритма автоматического взвешивания вагонов в движении и модуль распознавания номера вагонов на основе искусственной нейронной сети.

ВВЕДЕНИЕ

Задача организации процессов учета грузов, перевозимых железнодорожным транспортом, является крайне востребованной и актуальной на сегодняшний день. Это определяется развитием сети железных дорог и большими объемами грузовых перевозок [1]. Таким образом, автоматизация процесса взвешивания железнодорожных вагонов, как в статическом режиме, так и в движении является одним из основных направлений развития данной отрасли.

I. ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС

В данное время широко применяются статодинамические весы, позволяющие производить взвешивание как в статическом режиме с остановкой каждого вагона на весовой платформе, так и в движении на скорости до 15 км/ч. Главным преимуществом автоматизированного динамического режима взвешивания вагонов является существенная экономия времени, затраченного на взвешивание состава вагонов. Тем не менее, в данном режиме требуется специальная настройка весового оборудования для минимизации погрешностей полученных данных о взвешиваниях.

На рисунке 1 представлена схема взаимодействия разработанного программного средства с оборудованием системы взвешивания предприятия.

При установке вагона на измерительную платформу параметры взвешивания передаются от весового контроллера на персональный компьютер оператора по интерфейсу Ethernet с помощью протоколов TCP/IP. Уведомление машиниста локомотива о статусе процесса взвешивания производится посредством отображением данных на информационном табло. Для перехода в режим динамического взвешивания оператор подает соответствующую команду программному средству.

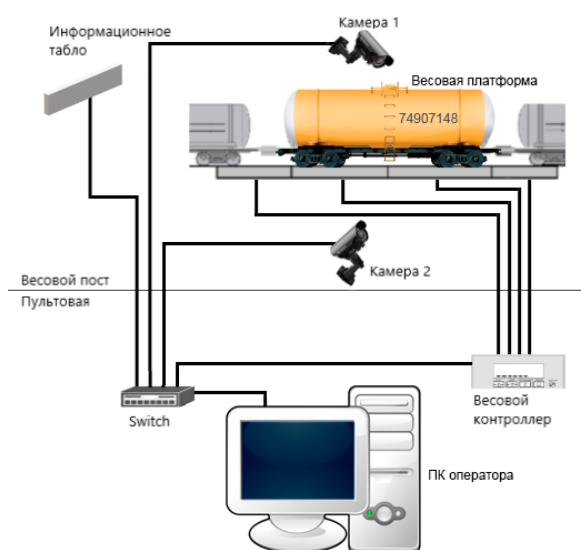


Рис. 1 – Схема взаимодействия программного средства с аппаратным комплексом предприятия

Данное программное средство разработано на языке CSharp. Для работы с видео-рядом использован VLC ActiveX компонент. Взаимодействие с информационным табло осуществляется посредством библиотеки VX-IV. Распознавание номера вагона на основе изображения из видео-ряда реализовано с помощью библиотек TensorFlow и OpenCV.

II. АЛГОРИТМ ДИНАМИЧЕСКОГО ВЗВЕШИВАНИЯ ЖЕЛЕЗНОДОРОЖНЫХ ВАГОНОВ

При составлении отчетов требуется сопоставление полученных значений результатов взвешиваний и конкретных вагонов. Вагоны идентифицируются номерами, также описывающими тип вагонов, осьность и основные характеристики [2]. Одним из важных этапов процесса автоматизированного динамического взвешивания является определение номера вагона.

Во время проезда подвижного состава по весам сохраняется видеозапись. После этого на ее

основе в моменты взвешивания конкретных вагонов вырезаются их изображения, корректируются и передаются модулю распознавания, разработанного на основе нейронной сети. Далее определяется номер каждого вагона.

Основным в реализации алгоритма автоматизации процесса динамического взвешивания является определение момента нахождения вагона на весоизмерительной платформе для точного распознавания номерного знака. Для получения данных о направлении движения состава и количестве осей, прошедших через каждую платформу весов по срабатыванию таймера каждые 20 миллисекунд на весовой контроллер посылается соответствующая команда. Как только значение количества осей, прошедших через первую платформу, сменяется с 0 на 1, начинается записываться видео с внешней видеокамеры. Как только значение количества осей, прошедших через платформы, перестает меняться в течение 5 секунд, что означает окончание взвешивания железнодорожного состава, видеозапись процесса взвешивания останавливается. Далее соответствующей командой запрашивается список протоколов взвешивания от весового контроллера и считывается соответствующий файл с данными о массе, осности и скорости каждого прошедшего через платформу вагона. Кроме того, в данном файле содержатся время начала взвешивания задней тележки на задней платформе (по ходу движения) T_1 и время окончания взвешивания задней тележки на задней платформе T_2 . Среднее значение величин T_1 и T_2 является наиболее вероятным моментом нахождения вагона приблизительно посередине задней платформы. Полученное значение может корректироваться в зависимости от местоположения видеокамеры и расположения номера вагона.

На основе видеозаписи процесса взвешивания подвижного состава в вычисленные моменты времени нахождения каждого вагона в наиболее оптимальном для распознавания положении сохраняются кадры, которые после предварительной обработки передаются в качестве входных параметров модулю распознавания номерного знака.

III. РАСПОЗНАВАНИЕ НОМЕРОВ ВАГОНОВ

Обработка фотографии происходит в несколько этапов: первоначально вырезается область изображения, в которой располагается номер вагона, после чего при необходимости удаляется эффект перспективы.

При разработке модуля распознавания были оценены результаты работы несколько архитектур моделей детекции: YOLOv2, FasterRCNN, SSD [3]. Наилучший результат по метрике mAP (mean average precision) показал детектор класса SSD (single shot detection). Исходя из балан-

са точности и требований была выбрана модель SSD-FPN, в качестве кодировщика признаков использовался ResNet101. Модель была обучена на реальных кадрах из видео-ряда, размеченных в инструменте CVAT (computer vision annotation tools). После распознавания символов на фотографии была использована кластеризация для получения местонахождения номерного знака с последующим выделением восьми наиболее релевантных чисел. Поскольку начальное количество кластеров было неизвестно, использовался алгоритм иерархической кластеризации.

На рисунке 2 представлен результат работы алгоритма детекции и кластеризации объектов. Распознанное значение номера вагона передается в основной модуль программного средства для последующей обработки полученных данных с целью учета транспортируемых грузов.



Рис. 2 – Результат работы алгоритма детекции

IV. ЗАКЛЮЧЕНИЕ

Разработанное программное средство позволяет организовать процесс взвешивания подвижных составов в динамическом режиме с обработкой полученных данных на основе распознанных номеров железнодорожных вагонов, что повышает эффективность процесса технологического учета грузов на крупных предприятиях.

V. СПИСОК ЛИТЕРАТУРЫ

1. Деменковец, Д. В. Система автоматической фото, видеофиксации и записи процесса взвешивания железнодорожных вагонов / Д. В. Деменковец // Информационные технологии и системы 2017 (ИТС 2017): Материалы международной научной конференции, БГУИР, Минск, Беларусь, 25 октября 2017 / БГУИР. – Минск, 2018. – 22–23 с.
2. Деменковец, Д. В. Алгоритм автоматического взвешивания железнодорожных вагонов / Д. В. Деменковец, К. Ю. Дубицкая // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / БГУИР. – Минск, 2019. – 94–95 с.
3. Борисевич, А. Н. Способы автоматического распознавания номеров вагонов железнодорожного транспорта / А. Н. Борисевич, Д. В. Деменковец // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / БГУИР. – Минск, 2019. – 87–89 с.

АНАЛИЗ ПРОИЗВОДИТЕЛЬНОСТИ ТЕХНОЛОГИЧЕСКОГО УЧАСТКА ВАКУУМНОЙ УСТАНОВКИ

Железнёв А. И., Доманов А. Т.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: zhelezonsany@gmail.com

Рассматривается структура технологического участка вакуумной установки. Анализируется вопрос производительности и загруженности его оборудования. Определяется зависимость производительности участка и загруженности его оборудования от числа тележек на технологическом участке.

ВВЕДЕНИЕ

Одним из способов получения тонкопленочных покрытий в промышленности является метод вакуумного осаждения материала. Для получения покрытий таким методом в условиях производства применяется система, управляющая перемещением подложек.

В вакуумной установке есть технологический участок, в котором происходит процесс осаждения материала на подложки. Этот участок является основным, поскольку в результате его работы получаем покрытие, готовое к поступлению рынок. Такой участок носит название технологическим, а камера - технологической. Исходя из производительности работы этого участка, можно судить о производительности работы установки в целом.

I. ОПИСАНИЕ ТЕХНОЛОГИЧЕСКОГО УЧАСТКА

Технологический участок состоит из транспортной и системы осаждения (см. рис. 1).

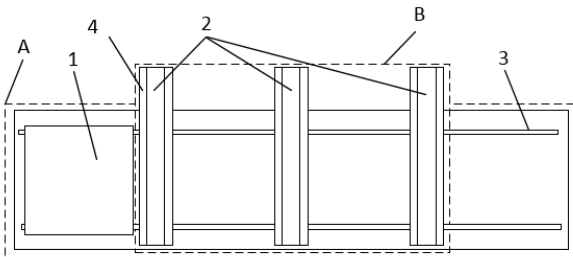


Рис. 1 – Структурная схема технологического участка (А – транспортная система, В – система осаждения, 1 – тележка, 2 – устройства осаждения, 3 – направляющие, 4 – технологическая защита)

Транспортная система включает в себя тележки которые перемещаются по направляющим. Система осаждения состоит из трех устройств осаждения материала, которые оснащены технологической защитой для предотвращения влияния устройства одной позиции на покрытие в другой.

Задачей этого участка является осаждение материала на подложки, расположенные на тележке. Тележка имеет длину l_T , количество под-

ложек $n_{\text{п}}$ перемещающиеся по участку с технологической скоростью v_T .

II. ПРОИЗВОДИТЕЛЬНОСТЬ УЧАСТКА

Рассмотрим вопрос производительности технологического участка при условии его бесперебойной работы и процесса с одним устройством осаждения. В таких условиях функционирования участка используются понятие цикловой производительности $Q_{\text{ц}}$ [1–3]:

$$Q_{\text{ц}} = k\eta_{\text{пр}}, \quad (1)$$

где k – технологическая производительность, а $\eta_{\text{пр}}$ – коэффициент производительности. Технологическая производительность представляет собой производительность участка вычисленную без учета потерь времени на холостые ходы, и характеризует возможности технологического процесса, положенного в основу работы участка. Величина коэффициента производительности технологического участка характеризует степень непрерывности процесса во времени.

Коэффициент производительности определяется отношением времени осаждения материала на подложку к периоду цикла осаждения:

$$\eta_{\text{пр}} = \frac{t_p}{T_{\text{ц}}}, \quad (2)$$

где t_p – рабочее время осаждения материала на подложку, а $T_{\text{ц}}$ – период цикла осаждения. Период цикла осаждения есть сумма рабочего времени и вспомогательного $T_{\text{ц}} = t_p + t_b$, где t_b – вспомогательное время на подачу тележки с подложками в рабочую зону и из её выхода.

Исходя из особенностей технологического процесса, технологическую производительность определим как:

$$k = \frac{n_{\text{п}}}{t_p}. \quad (3)$$

Учитывая, что тележки движутся непрерывно с технологической скоростью v_T , формула (3) примет следующий вид:

$$k = \frac{n_{\text{п}}v_T}{l_T}. \quad (4)$$

Как видно из (4) технологическая производительность для налаженного технологического процесса будет постоянной величиной. Тогда, анализируя (1), (2), (3), выходит что оптимизировать цикловую производительность можно путем изменения вспомогательного времени. Это время зависит от числа тележек, циркулирующих по технологическому участку, и расстояния между подложками, расположенных на тележке.

Для оценки загруженности устройств распыления введем в рассмотрение коэффициент загрузки оборудования k_3 . Он определяет степень использования оборудования во времени.

$$k_3 = \frac{t_{oc}}{T_{ц}}$$

где t_{oc} – время, в течении которого устройство распыления осаждало материал на подложки за один период цикла осаждения.

III. МОДЕЛИРОВАНИЕ

Для моделирования работы технологического участка, рассмотрим существующую установку компании ООО "Изовак". Длина тележек составляет 1200 мм, технологическая транспортная скорость – 10 мм/с, число подложек на тележке – 40, расстояние между подложками соседних тележек – 10 мм, расстояние между подложками на тележке – 2 мм, время возвращения тележки в технологический участок – 224 с.

Проводя компьютерное моделирование работы технологического участка получим зависимость коэффициента производительности и загруженности устройств распыления от числа тележек транспортной системы. Результаты моделирования (рисунок 2).

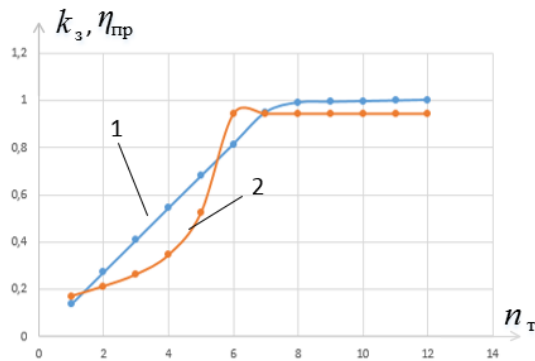


Рис. 2 – Зависимость значений коэффициентов от числа тележек в установке (1 – коэффициент загрузки оборудования, 2 – коэффициент производительности)

Как и следовало ожидать, при увеличении числа тележек на технологическом участке, увеличивается и коэффициент производительности, поскольку вспомогательное время уменьшается. Его величина будет отлична от нуля, так как существует расстояние между подложками на одной и соседних тележках (рисунок 3). Из графика видно, что при 7 тележках в технологическом участке, будет баланс между загрузкой участка и производительностью его оборудования.

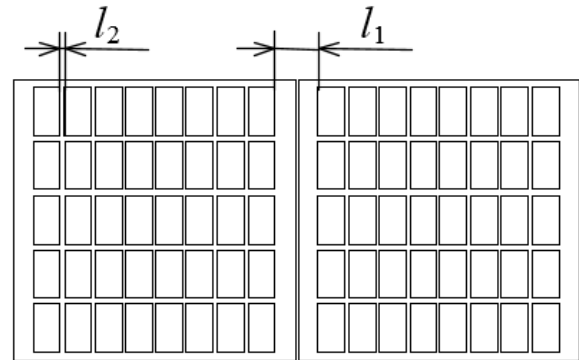


Рис. 3 – Схематичное расположение подложек на соседних тележках (l_2 – расстояние между подложками одной тележки, l_1 – расстояние между подложками соседних тележек)

ЗАКЛЮЧЕНИЕ

Результаты моделирования показали, что цикловая производительность технологического участка вакуумной установки нелинейно зависит от числа циркулирующих тележек в транспортной системе. Для обеспечения оптимального режима работы такого участка, необходимо дополнительное исследование загруженности устройств распыления. Оптимальное число транспортных тележек технологического участка находится там, где загруженность и цикловая производительность равны.

СПИСОК ЛИТЕРАТУРЫ

1. Артынов А. П. Автоматизация процессов планирования и управления транспортными системами / А. П. Артынов, И. И. Скалецкий. – М.: Наука, 1981. – 280 с.
2. Э. Э. Кольман-Иванов. Машины-автоматы химических производств. Теория и расчет. – М., «Машиностроение», 1972. – 296 с.
3. Болдин А. Н., Задиранов А.Н. Основы автоматизированного проектирования: Учебное пособие. – М.: МГИУ, 2006. – 104 с.

ANALYSIS OF MODEL PREDICTIVE CONTROL THROUGH A POWER CONVERTER IN A RENEWABLE ENERGY SYSTEM

Imad A. Elzein , Yury Petrenko
 International University of Beirut
 Department of computer science & information technology
 Beirut, Lebanon
 Belarusian National Technical University
 Department of electrical engineering
 Minsk, Belarus
 E-mail: Email: imad.zein@liu.edu.lb, ypetrenko@bntu.by

Predictive control developments for applications in the field of renewable energy systems are still under investigation. In this article, the fundamentals of predictive control are studied with a focus on model predictive control (MPC). Based on this techniques, a control strategy for flexible power supply can be developed which could be implemented in renewable energy systems, such as solar photovoltaic (PV) systems.

INTRODUCTION

Renewable energy power is an important solution to global warming. Solar energy generated from PV systems is one of the fastest and the most promising growing renewable energy types. Recently more power electronics converters have been used to integrate the energy sources into the AC and/or DC common buses in a distributed generation (DG) system. As the penetration and capacities of DG units increase, the power converters are required to operate more efficiently and effectively to maintain high power quality and dynamic stability. To fulfil these requirements, advanced control techniques have been intensively investigated in the last years. The main characteristic of predictive control is the use of the system model for the prediction of the controlled variables. Next, predefined optimized criterion selects the appropriate control set. The predictive controller aims for system-level control using control horizons of several minutes or even hours, but it fails to consider the discrete-time models and behaviours of power converters that act as power electronic interface between the renewable energy sources and the grid. In this paper we extend and explore the feasibility of predictive control by suggesting appropriate control strategies for renewable energy systems.

I. APPLICATION OF PREDICTIVE CONTROL IN RENEWABLE ENERGY SYSTEMS

For PV system, several useful topologies have been studied and applied. Figure 1 shows a typical configuration of PV system where several strings are interfaced with their own DC-DC converter for voltage boosting and then connected to a common DC bus. After that, a common DC-AC inverter is used for grid interfacing. Usually the MPPT is implemented on the DC-DC converter, while the grid synchronization and power regulation are achieved by the grid-side inverter.

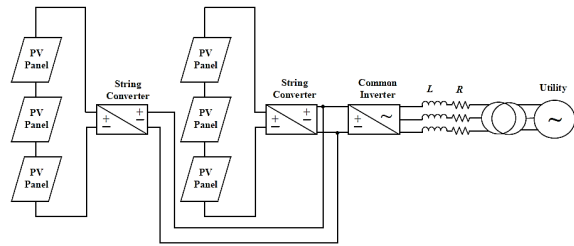


Рис. 1 – A typical configuration of PV system.

In this paper we concentrate on the control of the grid-side common inverter of the PV system (figure 1). Grid-connected PV systems should be controlled to regulate active and reactive powers flexibly for voltage support and power quality improvement. In this sense, flexible power regulation capability for a DG unit becomes more and more significant. For two-level inverters, there are eight possible voltage vectors generated by the inverter (six active vectors and two null vectors), and the $\alpha - \beta$ plane is divided into six sectors, as shown in Figure 2.

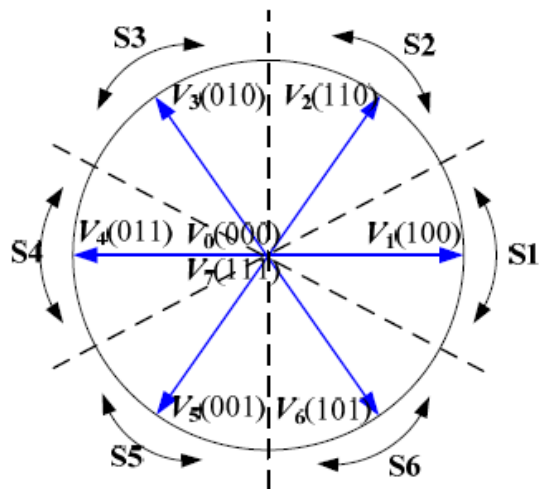


Рис. 2 – Possible voltage vectors generated by the inverter and sector division.

According to the equivalent circuit in figure 1, the system mathematical model can be expressed as:

$$V_i = V_g + IR + L \frac{d_i}{d_t} \quad (1)$$

where V_i and V_g are the inverter output voltage vector and grid voltage vector, respectively; I the line current vector; L the filter inductance; R the filter resistance. The instantaneous active and reactive powers exchanged between the PV and the utility grid can be expressed as:

$$P = \frac{3}{2} \text{Re} \{V_g I^*\} = \frac{3}{2} (V_{g\alpha} I_\alpha + V_{g\beta} I_\beta) \quad (2)$$

$$Q = \frac{3}{2} \text{Im} \{V_g I^*\} = \frac{3}{2} (V_{g\beta} I_\alpha - V_{g\alpha} I_\beta) \quad (3)$$

where a and b represent the real and imaginary components of the space vector expressed in the stationary frame. According to Equations (2) and (3), the active and reactive power derivatives can be calculated as:

$$\frac{dP}{dt} = \frac{3}{2} \left(\frac{dV_{g\alpha}}{dt} I_\alpha + V_{g\alpha} \frac{dI_\alpha}{dt} + \frac{dV_{g\beta}}{dt} I_\beta + V_{g\beta} \frac{dI_\beta}{dt} \right) \quad (4)$$

$$\frac{dQ}{dt} = \frac{3}{2} \left(\frac{dV_{g\beta}}{dt} I_\alpha + V_{g\beta} \frac{dI_\alpha}{dt} - \frac{dV_{g\alpha}}{dt} I_\beta - V_{g\alpha} \frac{dI_\beta}{dt} \right) \quad (5)$$

Considering sinusoidal and balanced line voltage, one can obtain:

$$\frac{dv_{g\alpha}}{dt} = -\omega V_{g\beta} \quad (6)$$

$$\frac{dv_{g\beta}}{dt} = \omega V_{g\alpha} \quad (7)$$

Thus, the inverter output active and reactive power derivatives can be obtained by substituting Equations (1), (6) and (7) into Equations (4) and (5) as:

$$\frac{dp}{dt} = -\frac{R}{L} P - \omega Q + \frac{3}{2L} \left(\text{Re} (V_g V_i^*) - |V_g|^2 \right) \quad (8)$$

$$\frac{dQ}{dt} = \omega P - \frac{R}{L} Q + \frac{3}{2L} \text{Im} (V_g V_i^*) \quad (9)$$

$$P^{k+1} = T_s \left[-\frac{R}{L} P - \omega Q + \frac{3}{2L} \left(\text{Re} (V_g V_i^*) - |V_g|^2 \right) \right] + P^k \quad (10)$$

$$Q^{k+1} = T_s \left[\omega P - \frac{R}{L} Q + \frac{3}{2L} \text{Im} (V_g V_i^*) \right] + Q^k \quad (11)$$

Now the predictive model has been obtained mathematically with Equations (10) and (11). Figure 3 depicts the block diagram of the proposed

MPC strategy for grid-connected PV systems. After the power is predicted, the next step is to evaluate the effects of each voltage vector on active and reactive powers and then select the one producing the least power ripple according to a specific cost function. Here, the cost function is defined as follows considering the same weighting priority for P and Q :

$$J = (P^* - P^{k+1})^2 + (Q^* - Q^{k+1})^2 \quad (12)$$

Once the optimal voltage vector is determined, it will be applied during the next sampling period to control the inverter.

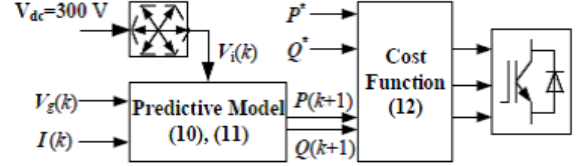


Рис. 3 – Block diagram of MPC strategy of PV systems.

II. CONCLUSIONS

This paper has reviewed the most important type of predictive control approaches, namely model predictive control which can be employed in renewable energy systems such as PV power generation. Application example has been described. With the increasing level of renewable energy sources penetration in existing power system, new challenges have been posted to the control of these distributed generation units (DGs). The DGs are not only controlled to injected power into the main grid, also required to participate in grid support by flexible power regulation.

REFERENCES

1. A. Elzein, I. Model Predictive Control for Positioning and Navigation of Mobile Robot with Cooperation of UAV/ Moustafa M Kurdi, Imad A Elzein and Alex K Dadykin// Journal of Communications on Applied Electronics (CAE), vol. 6, No.7, pp. 17–25, February 2017, USA.
2. A. Elzein, Imad. Adapting Model Predictive Control for a PV Station and Evaluating two different MPPT Algorithms P&O and FLC/ Imad Elzein, Y. N. Petrenko// Advances in Science, Technology and Engineering Systems Journal (ASTESJ), Vol. 2, No.3, P.741–748, 2017.
3. Elzein, Imad. Аналитические методы определения параметров режима максимальной выходной мощности солнечных батарей / Imad Elzein, Ю. Н. Петренко // Информационные технологии в образовании, науке и производстве : IV Международная научно–техническая интернет–конференция, 18–19 ноября 2016 г. [Электронный ресурс]. – [Б. и.], 2016.
4. Кузнецов А.П., Марков А.В., Шмарловский А.С., Гаврилик Т.В. Shaping – алгоритмы подавления колебаний объектов управления // Доклады БГУИР.– 2011.– № 6.– С. 5–11.
5. Ревотюк М.П., Кароли М.К., Хаджинова Н.В. Быстрый поиск кратчайших путей на графах с предопределенными решениями // Доклады БГУИР.– 2014.– № 4(82). – С. 73–79

ПЛАНИРОВАНИЕ ТРАЕКТОРИЙ ПРОМЫШЛЕННОГО РОБОТА-МАНИПУЛЯТОРА FANUK710iC С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

Кожевников М. М., Илюшин И. Э., Юркина А. А., Демиденко Е. Ю.
 Кафедра автоматизации технологических процессов и производств
 Могилевский государственный университет продовольствия
 Кафедра технологии машиностроения, Белорусско-Российский университет
 Могилев, Республика Беларусь
 E-mail: kmmk@mail.ru, iliue@yandex.by

Предложен новый эффективный алгоритм планирования траекторий роботов-манипуляторов, основанный на использовании модифицированной нейронной сети типа NRG. Разработанный алгоритм реализован на примере промышленного манипулятора Fanuc710iC.

ВВЕДЕНИЕ

Наиболее трудоемкой задачей при внедрении и эксплуатации роботизированных технологических комплексов (РТК) для процессов сборки, сварки, резки является планирование траекторий робота-манипулятора с учетом ограничений и столкновений с препятствиями. Для решения данной задачи обычно используются методы на основе динамического программирования [1, 2]. В работах [3, 4] для планирования траектории предложено использовать гармонические функции. Проведенный анализ показал, что существующие алгоритмы планирования траекторий в недостаточной мере учитывают специфику технологических операций, выполняемых роботом-манипулятором.

I. ПОСТАНОВКА ЗАДАЧИ

Задача планирования траекторий в конфигурационном пространстве сводится к нахождению последовательностей точек-конфигураций, образующих непересекающиеся с препятствиями и ограничениями траектории. При этом на конфигурационном пространстве шестиступенчатого робота Fanuc710iC налагаются ограничения вида

$$q_1 \in [q_1^- q_1^+], q_2 \in [q_2^- q_2^+], \dots, q_6 \in [q_6^- q_6^+], \quad (1)$$

где $q_i^-, q_i^+, i = 1..6$ – нижнее и верхнее предельные значения обобщенных координат. Внешняя граница конфигурационного пространства определяется ограничениями (1), а его внутренние границы соответствуют множеству точек столкновения:

$$CO_M \equiv \{ \mathbf{q} \in C_{spaceM} | (M)_q \cap B \neq \emptyset \}. \quad (2)$$

В общем случае внутренние границы могут быть несвязными. Критерий «качества» траектории зададим в виде:

$$J(\mathbf{q}(R)) = \sum_{R=0}^{N-1} \Phi(\mathbf{q}(R)), \quad (3)$$

где $\Phi(\mathbf{q}(R))$ – значение весовой функции на конфигурации $\mathbf{q}(R)$. Тогда задача планирования

траектории робота-манипулятора формулируется следующим образом: среди всех последовательностей, существующих в заданной свободной области конфигурационного пространства, границы которой определяются в соответствии с (1) и (2), и соединяющих стартовую \mathbf{q}_{start} и целевую \mathbf{q}_{goal} конфигурации, найти последовательность, на которой достигает экстремума критерий качества (3).

II. АЛГОРИТМ ПЛАНИРОВАНИЯ ТРАЕКТОРИЙ

Предложен новый алгоритм планирования траекторий, основанный на использовании модификации нейронной сети типа NRG и реализованный в три этапа. На первом этапе формируется карта конфигурационного пространства, ограниченная в соответствии с (1) и (2), которая буферизируется в слой пространственной памяти NRG. На втором этапе выполняется расчет унимодального потенциального распределения на выходе нейронной сети с точкой максимума в целевой конфигурации. На третьем этапе производится градиентный поиск траекторий на унимодальном потенциальном распределении. Элементарной ячейкой предлагаемой нейронной сети является трехмерная структура, образованная шестью нейронами. Выход каждого a -го нейрона поступает на вход каждого из b соседних нейронов. Таким образом, каждому нейрону ставится в соответствие узел дискретной карты, задающей конфигурационное пространство. Положению нейрона в координатной системе NRG соответствует единственная конфигурация робота. На вход каждого нейрона поступает также внешний сигнал от узла, задающего слой пространственной памяти.

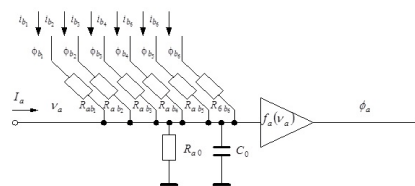


Рис. 1 – Схематическое изображение модели узла NRG

Рассмотрим аналоговую модель трехмерной структуры для предлагаемой модификации NRG (рис.1). Применяя к узлу a первый закон Кирхгофа, получаем

$$\sum_{b=1}^{N^6} i_{ab} + I_a - C_0 \frac{d\nu_a}{dt} - \frac{\nu_a}{R_{a0}} = 0, \quad (4)$$

где i_{ab} – ток, текущий от узла b к узлу a , I_a – ток от внешнего источника, ν_a – значение потенциала на входе нейрона a , а суммирование ведется по всем $b = 1, \dots, N^6$, так как проводимости между всеми узлами, кроме соседних, приняты равными нулю. Уравнение (4) может быть также представлено в виде

$$C_0 \frac{d\nu_a}{dt} = \sum_{b=1}^{N^6} \frac{\phi_b - \nu_a}{R_{ab}} - \frac{\nu_a}{R_{a0}} + I_a. \quad (5)$$

Тогда распределение потенциала в NRG задается системой уравнений

$$\tau_a \frac{d\nu_a}{dt} = \sum_{b=1}^{N^6} T_{ab} \phi_b - T_{a0} \nu_a + V_a, \quad (6)$$

$$a = 1, 2, \dots, N^6; \phi_a = f_a(\nu_a),$$

при

$$T_{ab} = \frac{R_{ab}^{-1}}{\sum_{b=1}^{N^6} R_{ab}^{-1}}, \tau = \frac{C_0}{\sum_{b=1}^{N^6} R_{ab}^{-1}}, T_{a0} = 1 + \frac{R_{a0}^{-1}}{\sum_{b=1}^{N^6} R_{ab}^{-1}}.$$

Функции активации нейронов $f_a(\nu_a)$ подбираются таким образом, чтобы для $f_a(\nu_a)$, а также для их производных, существовали обратные функции, т.е. $\nu_a = F_a(\phi_a)$.

ЗАКЛЮЧЕНИЕ

Эффективность предложенного алгоритма исследована с использованием роботизированной ячейки для плазменной резки с роботом-манипулятором Fanuc710iC (рис. 2). Параметры нейронной сети выбраны исходя из условия обеспечения асимптотической устойчивости. В ходе экспериментального исследования показано, что градиентный поиск сходится для нейронных сетей с различной размерностью скрытого слоя. Условием сходимости являлось существование хотя бы одной траектории между стартовой и целевой локациями. Поэтому выход из цикла поиска траектории осуществлялся в случае если траектория найдена, либо число итераций в ходе поиска достигло максимального значения.

СПИСОК ЛИТЕРАТУРЫ

1. LaValle, S. M. Planning Algorithms / S. M. LaValle. – Cambridge University Press, Cambridge, U.K., 2006. – 1023 p.
2. Springer Handbook of Robotics / ed. B. Siciliano and O. Khatib. Springer-Verlag, 2016. – 2227 p.
3. Lahijanian, M. A. Sampling-Based Strategy Planner for Nondeterministic Hybrid Systems / M.A. Lahijanian, L. E. Kavraki, M. Y. Vardi // International Conference on Robotics and Automation, Hong Kong, China. – 2014. – P. 3005-3012.
4. Генерация программных траекторий для роботоманипуляторов точечной контактной сварки в среде с препятствиями / М. М. Кожевников [и др.] // Доклады БГУИР. – 2017. – №7 (109). – С. 32-39.

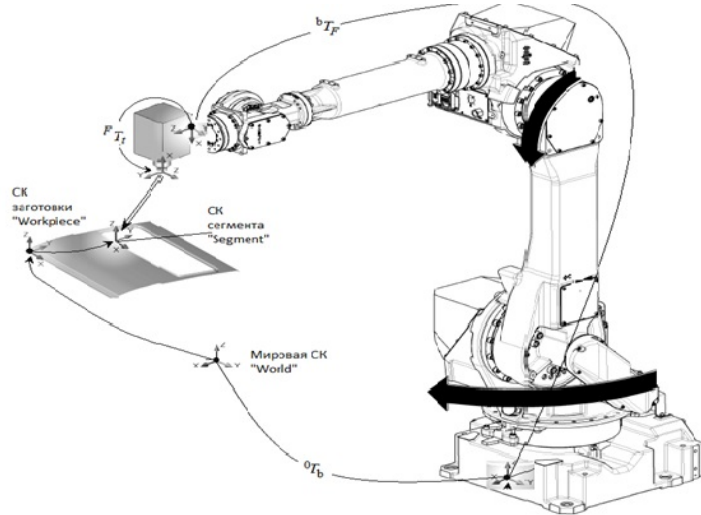


Рис. 2 – Роботизированная ячейка для плазменной резки с роботом-манипулятором Fanuc710iC

SYNTHESIS OF ROBUST TRACKING CONTROL SYSTEM UNDER PERTURBATION CONDITIONS

Kalmaganbetova Zh. A., Uashov E. K., Satybaldina D. K.

Department of Systems Analysis and Management, L.N.Gumilyov Eurasian National University

Nur-Sultan, Kazakhstan

E-mail: {kundyz_ai_92, Satybaldinad}@mail.ru, uashov.yer@gmail.com

In this paper a mathematical model of a rotating mechanical system consisting of two inertial masses connected by an elastic coupling is developed. Model parameters correspond to the parameters of one of the robots. For the transfer matrix of the control system, singular values are found that determine the attenuation of disturbances and ensure the stability margin of the control system. In the process of synthesis of the system under study based on heuristic approaches, expressions for weight transfer functions were obtained.

INTRODUCTION

The constant growth of requirements to the quality of industrial products, defines the appropriate requirements for accuracy and speed of working out the given laws and trajectory tracking of Electromechanical control systems. Tracking systems are used to control radar antennas, radio telescopes, artillery installations on mobile platforms, as well as to regulate the synchronicity and in-phase rotation of the shafts of the drive and driven engines when they are located at a sufficiently large distance from each other[1,2]. This determined the increased requirements for electric drives. Thus, the actual problem today is the construction of electric drive systems that provide requirements for accuracy and speed of working out the given laws and trajectories of the tracking Electromechanical control systems.

I. MATHEMATICAL DESCRIPTION OF THE CONTROL SYSTEM

We will consider a rotating mechanical system consisting of two inertial masses J_1, J_2 , connected by an elastic coupling [2] (fig.1).

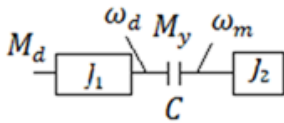


Fig. 1 – Mechanical system with elastic coupling

The moment M_y transmitted by the coupling, which is the moment of resistance for the first inertial mass and the driving moment for the second, is proportional to the difference in angular displacements of both masses with a coefficient of proportionality C . The control is the driving torque of the first mass (usually the torque of the drive motor) M_d , and the output is the speed of its rotation ω_d , since the speed sensor is usually installed on the shaft of the drive motor. For simplicity, the moment of resistance is assumed to be zero. System equations:

$$\frac{d\omega_d}{dt} = (M_d - M_y) / J_1 \quad (1)$$

$$\frac{dM_y}{dt} = C (\omega_d - \omega_m) \quad (2)$$

$$\frac{d\omega_m}{dt} = M_y / J_2 \quad (3)$$

We take into account in the considered system (1) – (3) the moment of loading M_c and assume that the purpose of regulation is to reduce the speed deviation of the mechanism ω_m under the action of M_c .

II. METHOD OF RESEARCH

There are a number of methods for designing robust systems, we will consider the methods that are used in the Robust Control Toolbox, which is part of the Matlab computing system. These methods are based on operations with frequency characteristics of systems. What role does Amplitude-frequency characteristic (AFC) play in assessing robustness: the smaller it is, the greater the change in the parameters of the object can be allowed without loss of stability [3,4]. Since the singular value $S(j\omega)$ determines the perturbation attenuation, the required perturbation attenuation can be given as

$$\sigma_1(S(j\omega)) \leq |W_1^{-1}(j\omega)|$$

With this in mind, the boundaries for the rest of the sensitivity function are given as:

$$\sigma_1(R(j\omega)) \leq |W_2^{-1}(j\omega)|.$$

$$\sigma_1(T(j\omega)) \leq |W_3^{-1}(j\omega)|.$$

The condition must be met

$$\sigma_1(W_1^{-1}(j\omega)) + \sigma_1(W_3^{-1}(j\omega)) > 1.$$

where W_1, W_2, W_3 are weight transfer functions; S, R, T – sensitivity functions for a given signal and control, as well as a complementary sensitivity function [5].

All the requirements for the system to mitigate disturbances and ensure the stability margin are

reduced to a single requirement for the norm

$$\|T_{y1u1}\|_{\infty} \leq 1,$$

where

$$T_{y1u1} = \begin{bmatrix} W_1 S \\ W_2 R \\ W_3 T \end{bmatrix}$$

- the so-called cost function of the mixed sensitivity method.

III. RESEARCH RESULT

Initially we will consider the position control system. Let's form weight frequency characteristics [6 – 8]. The characteristic W_2 is assumed to be a small constant. Characteristic W_3 is accepted in the form

$$W_3 = K_{f1} S^2 / 100,$$

where K_{f1} – is a configurable parameter and characteristic W_1 in the form

$$W_1 = \frac{K_f b (as^2 + 2z_1\omega_0\sqrt{as} + \omega_0^2)}{bs^2 + 2z_2\omega_0\sqrt{bs} + \omega_0^2}$$

The characteristics of the system with a full-order regulator while minimizing the H_{∞} norm are shown in figure 2. It is seen that the modulus of the transfer function T_{y1u1} in the essential frequency range is 1.

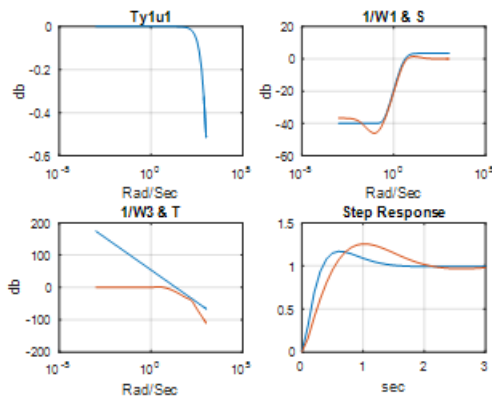


Fig. 2 – Characteristics of a system with a full-order regulator while minimizing the norm H_{∞} (position regulator)

This characteristic is called «all-pass». Now consider the speed control system. The resulting curves are shown in figure 3. It can be seen that the system requirements are satisfied.

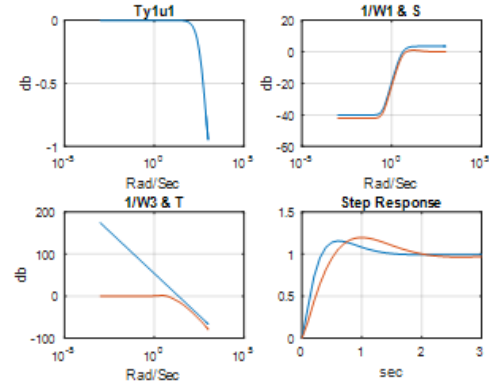


Fig. 3 – Characteristics of a system with a full-order regulator while minimizing the norm H_{∞} (speed regulator)

IV. SUMMARY

The paper presents the main approaches to robust structural synthesis of a rotating mechanical system consisting of two inertial masses connected by an elastic coupling. The model parameters correspond to the parameters of one of the robots. In developing the synthesis of a robust tracking system, methods based on operations with the frequency characteristics of systems are considered. The modeling of the system under study is carried out and the characteristics of the system are given. As a result of synthesis all requirements to the system are satisfied.

V. LIST OF REFERENCES

1. German-Galkin, S. G. Matlab Simulink. Proektirovanie mekhatronnyh system na PK. – SPb.: KORONA –Vek, 2008. – 368 s.
2. Kazancev, V. P., Dadenkov, D. A. Sintez diskretno-nepreryvnyh system upravleniya elektroprivodami s uprugimi svyaziyami // Elektrotehnika. – 2012. – № 11. – S. 24a –28.
3. Polyak, B. T., SHERBAKOV, P. S. Robustnaya ustojchivost' I upravlenie. – M.: Izd-vo RAN Institut problem upravleniya, 2002. – 273 c.
4. Pupkov, K. A. Metody robustnogo, nejro-nechytokogo i adaptivnogo upravleniya / K. A. Pupkov, N. D. Egupov, A. G. Gavrilov, V. YU. Zverev. M.: Izd-vo MGTU im. N.E. Baumana, 2002. 744 s.
5. Gu, D. W., Petkov, P. Hr., Konstantinov, M. M. Robust control design with MATLAB. London: Springer – Verlag, 2005. – 576 p.
6. Robust Control Toolbox, User's Guide, The MathWorks, 2001. – 655 p.
7. Dana Satybaldina, Aida Mashtayeva, Gulsanat Zekenova, Turakty Intymakov. Synthesis of Robust Control System for Aircraft // The VIII International Academic Congress „Fundamental and Applied Studies in EU and CIS Countries“0. – United Kingdom, Cambridge, England, 2018. – S.129-138.
8. Satybaldina, D. K., Zekenova, G. Z., Kalmagambetova, ZH. A. Koliq kuralyn robasty baskaru zhyiesin kuru // Vestnik Evrazijskogo nacional'nogo universiteta im. L. N. Gumileva. Seriya „Tekhnicheskie nauki i tekhnologii“0. – Astana, 2019. – №.1(126). – S.53 – 59. – ISSN 2616 –7263

РАСПРЕДЕЛЕННАЯ ТРАССИРОВКА

Кива В. С., Брановицкий А. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: knjiaz@gmail.com, art7yom@gmail.com

В статье описан один из способов решения проблемы диагностики производительности программного обеспечения в тестовой или рабочей среде, проблема локализации сбоев или необходимость построения графа зависимостей распределенной системы. И одним из способов решить эти проблемы – внедрение распределенной трассировки в существующую систему.

ВВЕДЕНИЕ

Распределенная трассировка, также называемая распределенной трассировкой запросов, представляет собой метод, используемый для профилирования и мониторинга приложений, особенно тех, которые построены с использованием архитектуры микросервисов. Распределенная трассировка помогает точно определить, где происходят сбои и что приводит к снижению производительности программного обеспечения.

I. СЛОЖНОСТИ ВНЕДРЕНИЯ ТРАССИРОВКИ

Внедрение распределенной трассировки является сложной задачей, потому что инструментарий должен распространять контекст трассировки как внутри, так и между процессами. Выполнение внедрения затрагивает почти каждую часть приложения. В частности, контекст трассировки должен быть передан через:

- Автономные сервисы программного обеспечения с открытым исходным кодом (NGINX, Cassandra, Redis и т. д.);
- Пакеты программного обеспечения с открытым кодом, связанные с пользовательскими службами (gRPC, ORM и т. д.).

Самое сложное в распределенной трассировке – это распределенная часть. Любая система трассировки нуждается в способе понимания причинно-следственной связи между действиями во многих отдельных процессах, независимо от того, связаны ли они через формальные структуры RPC, системы подписчик-издатель, общие очереди сообщений, прямые HTTP-вызовы, UDP-пакеты или иным способом.

Некоторые распределенные системы отслеживания (например, Project5 от 2003 г., WAP5 от 2006 г. или The Mystery Machine из 2014 г.) выводят трассировку за границы процесса (подход черного ящика). Что касается качества, то OpenTracing является явным стандартом инструментов распределенной трассировки, и он намного лучше соответствует подходам, таким как X-Trace от 2007 года, Dapper от 2010 года, или многочисленным системам трассировки с открытым исходным кодом, таким как Zipkin или Jaeger (среди прочих).[1]

Нецелесообразно просить все службы программного обеспечения с открытым исходным кодом, внедрить одного поставщика трассировки в их пакеты и в весь код приложения. Тем не менее, если у них нет общего механизма для описания и распространения спанов, трассировка нарушается, а спаны не являются полными. Поэтому нужен единый стандартный механизм для описания поведения систем.

II. ФОРМАТ ТРАССИРОВКИ

Для упрощенного интегрирования между библиотеками программного обеспечения по трассировке запросов введен один из общих форматов – OpenTracing. OpenTracing состоит из спецификации программных интерфейсов, интерфейсов и библиотек, которые реализовали спецификацию и документацию проекта. OpenTracing позволяет разработчикам добавлять трассировочный инструментарий в код своего приложения с помощью программных интерфейсов, которые не привязывают их к какому-либо конкретному продукту или поставщику.

OpenTracing стандартизирует:

- Управление «спанами»: программный интерфейс приложения для запуска, завершения и декорирования временных операций;
- Межпроцессное распространение: программный интерфейс приложения для передачи контекста трассировки за границы процесса;
- Управление активным спаном: программный интерфейс приложения для хранения и извлечения активного спана в одном процессе;
- Внутрислое контекстное кодирование: спецификация точного формата кодирования для отслеживания контекста, передаваемого вместе с данными приложения между процессами;
- Кодирование данных внеполосной трассировки: спецификация того, как кодируются декорированные данные трассировки и спана, когда они направляются к поставщику распределенной трассировки.

Все специфичные для языков программирования программные интерфейсы OpenTracing

имеют некоторые основные понятия и терминологию. Эти концепции настолько важны, что у них есть свой собственный репозиторий на github.[1]

III. СПАН

Одним из фундаментальных понятий OpenTracing является «Спан». «Спан» – это основной строительный блок распределенной трассировки, представляющий отдельную единицу работы, выполненную в распределенной системе. Каждый компонент распределенной системы вносит свой спан – именованную временную операцию, представляющую часть рабочего процесса.

Спаны могут (и обычно содержат) ссылки на другие спаны, что позволяет объединять несколько спанов в одну полную трассировку и визуализировать сроки действия запроса при его перемещении по распределенной системе.

В настоящее время OpenTracing определяет два типа ссылок спанов: ChildOf и FollowsFrom. Оба ссылочных типа специально моделируют прямые причинно-следственные связи между дочерним спаном и родительским спаном. В будущем OpenTracing может также поддерживать ссылочные типы для спанов с не причинно-следственными связями (например, спаны, которые объединены вместе, спаны, которые ждут в одной очереди и т. д.).[2]

Ссылки ChildOf: спан может быть ребёнком родительского спана. В ссылке ChildOf родительский спан в некоторой степени зависит от дочернего спана. Примеры ChildOf:

- Спан, представляющий серверную часть RPC, может быть ChildOf спана, представляющим клиентскую сторону этого RPC;
- Спан, представляющий вставку SQL, может быть ChildOf спана, представляющим метод сохранения ORM;
- Множество спанов, выполняющие параллельную (возможно, распределенную) работу, могут все по отдельности быть ChildOf одного родительского спана, который объединяет результаты для всех дочерних элементов, которые возвращаются в установленные сроки.

Каждый спан инкапсулирует следующее состояние в соответствии со спецификацией OpenTracing:

- Название операции;
- Метка времени начала и окончания;
- Наборы ключ-значение, содержащие теги;
- Наборы ключ-значение, содержащие логи.

IV. НАБОРЫ КЛЮЧ-ЗНАЧЕНИЕ

Теги – пары ключ-значение, которые позволяют задавать пользователем аннотации диапазонов для запроса, фильтрации и уточнение данных трассировки.

Логи являются парами ключ-значение, которые полезны для сохранения сообщений журнала, относящихся к спан, и других отладочных или информационных выходных данных самого приложения. Логи могут быть полезны для документирования определенного момента или события в промежутке (в отличие от тегов, которые должны применяться к промежутку в целом).

V. ПОТОКИ

В любом конкретном потоке есть «активный» спан, в основном отвечающий за работу, выполняемую окружающим кодом приложения, который называется активным спаном. Программный интерфейс приложения OpenTracing позволяет активировать только один спан в потоке в любой момент времени. Это управляется с помощью области видимости, которая управляет описывает активацию и деактивацию спана.[3]

Другие спаны, которые связаны с тем же потоком, будут удовлетворять одному из следующих условий:

- Начался;
- Не закончен;
- Не активный.

Например, в одном потоке может быть несколько спанов, если они:

- В ожидании ввода / вывода;
- Заблокированы на дочернем спане;
- Вне критического пути.

Если область видимости существует, когда разработчик создает новый спан, он будет выступать в качестве его родительского, если программист не вызовет специальный метод по игнорированию активного спана во время его создания или не укажет родительский контекст явно.

VI. ПОТРЕБИТЕЛИ

Команды информационно-коммуникационных технологий и администраторов используют OpenTracing для мониторинга приложений. Распределенная трассировка особенно хорошо подходит для отладки и мониторинга современных архитектур распределенного программного обеспечения, таких как микросервисы. Разработчики могут использовать распределенную трассировку для отладки и оптимизации своего кода.

1. Vendor-neutral APIs and instrumentation for distributed tracing [Электронный ресурс] / Documentation Home – Режим доступа: <https://opentracing.io/docs>. – Дата доступа: 29.09.2019.
2. GitHub is where people build software [Электронный ресурс] / Documentation Home – Режим доступа: <https://github.com/opentracing/specification>. – Дата доступа: 28.09.2019.
3. Medium, a place where words matter [Электронный ресурс] / Distributed Tracing in 10 Minutes – Режим доступа: <https://medium.com/opentracing/distributed-tracing-in-10-minutes-51b378ee40f1>. – Дата доступа: 28.09.2019.

СИСТЕМА ТОЧНОГО ПОЗИЦИОНИРОВАНИЯ НИЗКОСКОРОСТНЫХ МИНИРОБОТОВ

Коников А. Д., Самаль Д. И.

Кафедра электронных вычислительных машин, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: skonikov@mail.com, samal@bsuir.by

Автономные мобильные роботы применяются во многих областях жизнедеятельности человека. Для уменьшения габаритов и стоимости системы может применяться множество идентичных компактных самоорганизующихся роботов. В данной статье рассмотрена система перемещения компактного робота при помощи одного пьезоэлемента, обоснована необходимость применения обратной связи, рассмотрен способ определения взаимно расположения множества роботов на основе триангуляции радиосигнала, показаны преимущества прецизионного позиционирования роя и способы применения данного свойства.

Сфера автономных мобильных роботов набирает высокую популярность и получает всё большее распространение. Согласно прогнозам наибольшее развитие она получит в течение ближайших 5-10 лет [1]. Одним из наиболее приоритетных направлений в данной области являются исследования коллективного интеллекта, иными словами роевого поведения. Подразумевается, что множество независимых относительно компактных роботов могут объединяться для решения одной большой и сложной задачи, непосильной для её полного выполнения одним роботом, но при координации действий нескольких – вполне. Такой задачей является, например, перемещение крупного и тяжёлого объекта. При подобном подходе и соответствующей координации становится принципиально возможной распараллеливание между роботами и иных – простейших задач, с соответствующим уменьшением времени на каждую из них.

Наиболее подходящими для реализации компактных роботов являются электрические, в том числе и вибрационные двигатели, т.к. они обладают очень высоким отношением мощность/вес, имеют компактные размеры, обладают низким тепловыделением, уровнем шума, а также высоким КПД по сравнению с двигателями внутреннего сгорания, что позволяет существенно увеличить время автономной работы робота. В рамках проводимой исследовательской работы применяется вибрационный двигатель основанный на применении пьезоэлемента, разработанный и запатентованный профессором В.Т. Минченей [2].

Платформа, служащая основанием для робота, представляет собой равнобедренный треугольник с загнутыми краями, изготовленный из нержавеющей стали, к которому приклеен пьезоэлемент (см. рис.1). Подача переменного напряжения с определённой частотой, определяемой эмпирически, на пьезоэлемент, вызывают его колебания. Так как пьезоэлемент жёстко соединён

с платформой при помощи цианокрилата, то все колебания пьезоэлемента передаются на стальное основание.

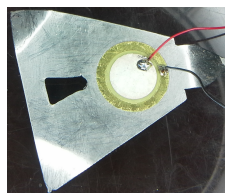


Рис. 1 – Платформа с пьезоэлементом

Волны механических колебаний распространяются по металлу и вызывают пружинящие колебания (изгибания) очень малой амплитуды ножек платформы. При попадании в частоту резонанса, из-за различных расстояний от пьезоэлемента до концов ножек платформы в определённый момент вектор суперпозиции сил оказывается ненулевым и направленным в определённую сторону, что вызывает движение платформы. Сужающиеся углы треугольника вызывают существенный рост амплитуды волн колебаний на концах ножек основания, что обеспечивает ещё большее смещение платформы. Отверстие в платформе расположено в точке наибольших механических напряжений и вызывает переотражение и усиление колебаний.

При движении платформы, из-за шероховатости поверхности, по которой она движется, возникают неконтролируемые отклонения движения робота от намеченного курса. С целью устранения отклонений в движении необходимо вводить корректировки, рассчитываемые на основании системы обратной связи, построенной на базе оптического сенсора компьютерной мыши.

В идеале, пьезоэлемент должен быть приклеен ровно посередине платформы и над отверстием, однако из-за необходимости подключения оптического сенсора компьютерной мыши пьезоэлемент пришлось сместить, а отверстие расширить, чтобы оптический сенсор мог реагировать на поверхность, по которой движется робот. При-

чиной таких изменений стало то, что центр тяжести использованной мыши находится приблизительно в области оптического сенсора и требовалось, чтобы центр тяжести находился над платформой, а не за её пределами для обеспечения устойчивости платформы и равномерной нагрузки на её ножки. Равномерность распределения нагрузки является одним из необходимых условий для приведения данной системы в легко контролируемое движение.

Основные достоинства данной платформы - простота и дешевизна конструкции, позволяющая создать множество экземпляров для изучения роевого поведения роботов.

Главными недостатками являются сложность при изготовлении стальной платформы, т.к. требуется точно соблюсти пропорции, а также необходимость подбора частот необходимых для приведения платформы в движение.

Первый недостаток может быть устранён путём изготовления вырубного и формовочного штампов, что кроме повышения точности изготовления позволит ускорить и удешевить процесс производства больших партий.

Второй недостаток устраняется алгоритмически. Так как у пьезоробота имеются органы обратной связи о перемещении, например основанные на оптическом сенсоре, который применяется в компьютерной мыши, то перебирая все частоты генератора автоматически в определённом диапазоне и при этом регистрируя перемещения платформы можно получить сведения о том, какие частоты отвечают за движение в определённом направлении. Форма сигнала на выходе генератора представлена на рис. 2 и соответствует частоте 15 кГц.



Рис. 2 – Сигнал поступающий на пьезоэлемент

Применение оптического сенсора мыши в системе обратной связи позволяет не только снизить стоимость системы в целом, но также и добиться очень высокой точности позиционирования. При использовании недорогого оптического сенсора с разрешением в 600 dPi (dots per inch) точность позиционирования достигает 0,042мм, а при использовании более прогрессивного сенсора с разрешением в 1600 dPi точность позиционирования достигает 0,015875мм.

При реализации алгоритмов роевого поведения на множестве роботов возникает необходимость координации их взаимодействия между собой. При взаимодействии кроме обмена данными требуется определять так же и взаимное расположение. Существует несколько возможных реализаций данной системы, например на основе технологии Bluetooth или Wi-Fi. Роботы подключаются в Mesh сеть, благодаря этому появляется

возможность обмениваться данными. Обе технологии позволяют определять уровень сигнала передатчика. Таким образом, появляется возможность определить расстояние до соседнего робота. Зная расстояние между множеством роботов можно вычислить их взаимное расположение путём использования триангуляции радиосигнала, по принципу, схожему с принципом работы системы GPS, однако в данном случае расстояние до другого робота (спутника) определяется не на основании разницы во времени, а на основании затухания радиосигнала. Минимальным необходимым количеством роботов в данном случае является 3, т.к. только таким образом можно определить положение робота в пространстве путём пересечения 3 окружностей, радиус которых пропорционален затуханию сигнала передатчика (см. рис.3).

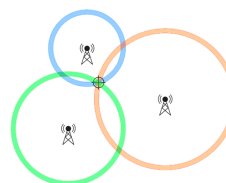


Рис. 3 – Триангуляция радиосигнала

Одной из перспектив развития данного проекта является использование роботов на основе пьезоэлементов в дефектоскопии: после того, как робот приведён в движение и продолжает двигаться по инерции, при помощи пьезоэлемента можно регистрировать затухающие колебания и анализируя их, можно судить о наличии микрорезультатов на поверхности, по которой двигался робот. Компактные размеры робота позволят анализировать поверхности в труднодоступных местах, например внутреннюю поверхность труб.

Конечной целью данного исследования является построение робототехнической платформы для исследования роевого поведения. Точное позиционирование и определение взаимоположения роботов между собой позволяет реализовать так называемое чувство «боковой линии», которое наблюдается у рыб. Таким образом появляется возможность абсолютно синхронного перемещения роботов, что является зрелищным процессом, например и может использоваться для популяризации науки среди молодёжи.

СПИСОК ЛИТЕРАТУРЫ

1. Gartner Hype Cycle [Электронный ресурс]. – Режим доступа : <https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/>
2. Цифровые системы управления технологическим оборудованием : уч. пособие / В. Т. Минченя [и др.]. – Минск : Энциклопедикс, 2016. – 108 с. : ил., табл.
3. Схемотехника цифровых преобразователей перемещений: Справочное пособие/ В.Г. Домрачев, В.Р. Матвеевский, Ю.С. Смирнов. – М.: Энергоатомиздат, 1992. - 392с.: ил.

WEB-ПРИЛОЖЕНИЕ ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССА ОБРАБОТКИ ЗАКАЗОВ

Конончик О. Н.

Кафедра программного обеспечения информационных систем и технологий, Белорусский национальный технический университет
Минск, Республика Беларусь
E-mail: o.kononchik@mail.ru

В настоящее время автоматизация процесса обработки заказов играет важную роль в любой области производства и оказания услуг. Не все из существующих программных решений, позволяющие автоматизировать большинство задач и процессов, связанных с обработкой заказов, эффективны для конкретной компании. Для автоматизации процесса обработки заказов было разработано web-приложение, предназначенное для предприятий, имеющие ограниченные ресурсы и работающие в условиях повышенной нагрузки. Основой web-приложения является механизм отбора заказов, цель которого – формирование лучшего набора заказов из большого числа поступающих заказов.

ВВЕДЕНИЕ

В наше стремительное время в условиях развития рыночных отношений, непрерывно и быстро меняющихся реформ и реконструкций, когда одно нововведение спешит сменить другое, всё больше компаний осознают важность перемен в сфере автоматизации своего бизнеса. Как следствие разработано большое количество информационных систем, позволяющих автоматизировать большинство задач и процессов, характерных для современных служб управления компаний. [1]

Основным процессом, который играет важную роль в любой области производства и оказания услуг является процесс обработки заказов. Но в ходе этого процесса возникает большое количество проблем, связанных с несовершенством применяемых программных и технических средств или с отсутствием поддержки этих бизнес-процессов в программном обеспечении (ПО), а также с недостаточной автоматизацией основных процессов, что приводит к различным ошибкам и сбоям, а, главное, к снижению экономических показателей компании. [2]

Решением таких проблем может стать программная поддержка процесса обработки заказов, которая заключается в использовании программного обеспечения для автоматизации основных этапов процесса обработки заказов. С помощью автоматизации процесса обработки заказов можно существенно облегчить работу сотрудникам компании и сэкономить их время, путем сокращения большого количества рутинных операций, что приведет к повышению экономических показателей и скорости работы всей компании в целом. [3]

1. ОПИСАНИЕ WEB-ПРИЛОЖЕНИЯ

В настоящее время разработано большое количество информационных систем, позволяющих автоматизировать большинство задач и процессов, связанных с обработкой заказов. Однако

не все из них одинаково эффективны для конкретной компании.

Большинство из существующих программных средств предоставляются, как правило, в виде отдельного специализированного приложения, которое устанавливается и настраивается на каждом компьютере в отдельности. Такое приложение невозможно использовать одновременно на устройствах, работающих на разных операционных системах. Поэтому, в последнее время такие специализированные приложения используются компаниями все реже и реже.

Для автоматизации процесса обработки заказов было разработано web-приложение на основе инфраструктуры интернет (см. рис. 1) – где:

1. создание заказа;
2. добавление заказа;
3. запрос заказов на исполнение;
4. запрос лучшего набора заказов;
5. запрос всех доступных заказов;
6. отправка всех доступных заказов;
7. отправка лучшего набора заказов;
8. отправка лучших заказов на исполнение.

Разработанное приложение предназначено для предприятий, которые имеют ограниченные ресурсы и работают в условиях большой нагрузки. Основное назначение web-приложения – это автоматизация приема, управления и исполнения заказов с целью максимизации прибыли.

Приложение предоставляется конечному потребителю в виде пользовательского (графического) интерфейса, доступ к которому осуществляется посредством web-браузера через сеть интернет. Созданная система обеспечивает разделение прав доступа для различных категорий пользователей, которые создаются системным администратором. [4]

Для повышения эффективности поддержки и сопровождения приложения, а также его дальнейшего расширения при создании web-приложения использовалась трёхуровневая архитектура построения программных систем и

клиент-серверная архитектура. В качестве уровня доступа к данным используется СУБД MySQL. Реализация команд выборки данных, контроль целостности и непротиворечивости данных осуществляется с помощью соответствующих хранимых процедур, триггеров и других объектов, предоставляемых сервером. Уровень бизнес-логики развернут на сервере приложений и представляет собой ядро системы. [4]

Разработанное web-приложение обладает следующими возможностями:

- регистрация и аутентификация пользователей;
- создание, управление и обработка заказов;
- управление личным счетом клиента;
- просмотр сделанных заказов;
- просмотр информации о ходе исполнения заказов;
- формирование истории операций.

II. ОТБОР ЛУЧШИХ ЗАКАЗОВ

Основой разработанного web-приложения стал механизм отбора лучших заказов из большого числа поступающих заказов, для дальнейшей передачи их на исполнение. Целью данного механизма является формирование лучшего набора заказов из большого количества заказов, например – набор заказов, с меньшим объемом и с более высокой стоимостью по отношению к другим. Для этого разработана оптимизационная модель процесса обработки заказов.

Лучшие заказы отбираются на основе заданных критериев оптимальности. Каждый поступающий заказ характеризуется тремя показателями:

- количество выполнений (С);
- стоимость выполнения (Р);
- скорость выполнения (V).

Соответственно и весь набор заказов характеризуется этими тремя показателями, а показатели С, Р, V – выступают в качестве критериев оптимальности.

Механизм отбора лучших заказов работает следующим образом. На вход модуля поступает N заказов. Из N отбирается k лучших заказов. Из

всех входящих заказов формируются так называемые наборы по k заказов. Каждый набор – это k различных заказов. Для нахождения возможных наборов заказов используется метод «Тасование Фишера-Йетса». А для нахождения общего числа таких наборов заказов используется формула вычисления сочетаний без повторов.

Для каждого поступающего заказа вычисляется обобщенный критерий, а для каждого набора заказов вычисляется сумма обобщенных критериев входящих в этот набор заказов. Набор, который имеет максимальную сумму обобщенных критериев устанавливается как оптимальный, а все заказы, входящие в этот набор немедленно отправляются на исполнение.

ЗАКЛЮЧЕНИЕ

Основным достоинством разработанного web-приложения можно считать то, что оно не нуждается в установке, а значит работать с ним можно с любого клиентского устройства под любой операционной системой. Благодаря разработанному web-приложению, предприятия, имеющие ограниченную пропускную способность и большой поток заказов, смогут существенно максимизировать свою прибыль. Достигается это за счет интегрированного в web-приложение модуля для отбора заказов, посредством которого, из большого потока заказов на первоочередное исполнение отправляются только лучшие заказы.

СПИСОК ЛИТЕРАТУРЫ

1. A. Noroozi, M. Mahdavi Mazdeh, M. Rasti-Barzoki. Coordinating Order Acceptance and Batch Delivery for an Integrated Supply Chain Scheduling. International Journal of Engineering, Vol. 30, No. 5, (May 2017) 700-709.
2. Обработка заказов – «Мой склад». [Электронный ресурс] – Режим доступа: <https://support.moysklad.ru/hc/ru/articles/203054596-Обработка-заказов,свободный>. – Загл. с экрана.
3. Автоматизация процессов обработки заказов. [Электронный ресурс] – Режим доступа: <http://blog.web100.biz/matematika-rasskazhet-ob-avtomatizacii-b2b-processov-luchshe-chem-slova/>, свободный. – Загл. с экрана.
4. Колесниченко Д. Н. PHP 5/6 и MySQL 6. Разработка Web-приложений. - СПб.: БХВ-Петербург, 2011.



Рис. 1 – Схема функционирования web-приложения

ОБ АГРЕГИРОВАННОМ ПОДХОДЕ ВЫБОРА КОМАНД ИТ-ПРОЕКТОВ В СИСТЕМАХ ПРИНЯТИЯ РЕШЕНИЙ

Котковец А. А., Поттосина С. А.

Кафедра экономической информатики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: kotkovets.a.a@gmail.com, s.pottosina@gmail.com

Одной из самых распространенных форм организации труда в ИТ-компаниях является проектная. При данной форме организации труда вся ответственность ложится на проектной команде. Поэтому важным вопросом является подбор такой команды, которая полностью соответствовала бы требованиям проекта и могла бы самостоятельно решать возникающие вопросы.

ВВЕДЕНИЕ

Проектная команда – это человеческие ресурсы, привлекаемые на постоянной или временной основе для работы над выходной продукцией проекта. Члены проектной команды обязаны понимать суть задания, которое предстоит выполнить; детально спланировать порученные работы; завершить их в пределах бюджета, в заданные сроки и качественно; а также информировать руководителя проекта о возникающих вопросах, изменениях объема, рисках и проблемах качества [1].

I. Задачи исследования

В современном мире управление в области ИТ – особая область управления, применение которой дает ощутимые результаты. Профессионалы в этой области высоко ценятся (например, в США это третья профессия после юристов и врачей по средней величине оплаты), а методологии управления ИТ-проектами стали фактическим стандартом управления на многих предприятиях и применяются в той или иной степени практически во всех организациях [1].

Каждый проект требует временной и финансовой оценки, приоритизации, поскольку без данной работы над каждым проектом эффективность управления может быть подвержена значительным рискам. Ошибки в оценке и приоритизации неизбежно приведут к ошибкам в планировании и управлении проектом, что, как правило, сказывается на качестве выполняемых работ, на зарплате сотрудников, приводит к потере прибыли и снижению авторитета организации [3]. Потому очень важно уделять необходимое время и ресурсы процессам оценки проектов в общей структуре процесса управления ИТ-проектами в каждой организации. Проблема оценки проектов является актуальной не только на начальном этапе, предшествующем внедрению проекта, но и на последующих этапах, например, при корректировке ранее принятых решений [4].

Сказанное выше позволяет говорить об актуальности исследований в области моделирования автоматизированных решений оценки про-

ектов и проектных команд путем привлечения средств искусственного интеллекта, и, в первую очередь, систем и методов поддержки принятия решений. В настоящее время разработаны и успешно применяются на практике множество систем поддержки принятия решений (СППР).

Основными задачами данной работы являются:

- изучить и проанализировать теоретические работы по формированию проектных команд;
- провести сравнение и анализ существующих систем принятия решений с целью выделения набора критериев, требующих улучшения;
- разработать агрегированные показатели с целью устранения выявленных недостатков.

II. СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

В традиционных СППР модель знаний (когнитивная модель) о предметной области создается с привлечением инженеров по знаниям и ориентируется, как правило, на конкретные задачи. К основным инструментальным средствам, позволяющим выполнять когнитивное моделирование, относятся такие СППР, как: КоСМОС; Канва; PolyAnalyst; Deductor; Fuzzy Thought Amplifier; Cope; NIPPER; Gismo; iThink, Hyper; RESEARCH; ATLAS/ti; Metamorph; KANT; Meta design; Гипердок; FCM Analyst [4].

Рассмотрим упомянутые системы в сравнении. В целях осуществления сравнительного анализа данных систем был проведен сравнительный анализ функциональных критериев, которые являются наиболее важными и часто используемыми в процессе оценки проектов. Данный анализ был проведен на основе исследованных основ и принципов оценки и приоритизации ИТ-проектов, и в результате его проведения были выделены следующие критерии:

1. Развитый графический интерфейс, понятный для любого пользователя системы.
2. Возможность учета особенностей человеческой системы измерения, переработки и

- оценки информации, а также возможность коррекции заблуждений экспертов, производящих оценку.
- Объединение в единой модели факторов, имеющих числовые и логические, либо лингвистические типы значения.
 - Возможность генерации системой возможных альтернативных решений.
 - Возможность моделирования сценариев развития ситуации в зависимости от принимаемого решения.
 - Возможность отбора альтернатив.
 - Возможность одновременного коллективного использования системы с функциональностью поддержки согласования групповых решений.
 - Возможности построения новой модели, а также генерации новых альтернатив в случае расхождения прогнозируемых и фактических данных.

Таблица 1 – Обзор возможностей СППР

СППР	1	2	3	4	5	6	7	8	С
КоСМоС	+	-	+	+	+	+	+	+	7
Канва	+	-	+	+	+	+	-	+	6
Poly-Analyst	-	+	-	+	+	+	-	-	4
Deductor	+	+	-	+	+	+	-	-	5
iThink	-	+	-	+	+	+	+	+	6
FCM-Analyst	+	-	+	+	-	-	-	+	4
ИГЛА	+	+	-	+	+	+	+	+	7

III. АГРЕГИРОВАННЫЕ ПОКАЗАТЕЛИ

Проанализированные СППР позволяют выполнять детальный расчет сценариев развития ситуации в зависимости от принимаемого решения, а также учитывать возможные альтернативы. Необходимо отметить, что каждая система имеет свои минусы:

- сложный и интуитивно непонятный графический интерфейс;
- непрозрачность и неочевидность процесса получения итоговых результатов;
- сложность механизма оценки и выбора альтернатив;
- высокая цена использования продуктов.

На основании этой информации было принято решение о разработке подхода с использованием агрегированных показателей, способного расширить возможности рассмотренных систем, а именно, производить прозрачный автоматический расчет трудовых ресурсов проектных команд с представлением обоснованных и экономически выгодных альтернатив.

Агрегированный показатель проекта построен на основании его расчетной стоимости, сложности, длительности и приоритета для организации, а агрегированный показатель сотрудника – на основании уровня полезности для ор-

ганизации, опыта, производительности, уровня качества выполняемых работ, дополнительных знаний, навыков и других параметров сотрудника.

Очевидно, что данные два показателя являются независимыми по отношению друг к другу. Однако в рамках разных проектов комплектация команды для разработки может и должна дифференцироваться, т.к. навыки и опыт одного и того же сотрудника, дающие определенную эффективность от привлечения его на одном проекте, могут отличаться от тех же показателей на другом проекте. Каждый проект организации, который берется в разработку, имеет свою оценку сроков реализации и приоритетность для данной организации, потому важно рассматривать агрегированные показатели сотрудника и проекта в корреляции для каждого конкретного случая, поскольку в рамках проектов разного уровня приоритета и стоимости каждый сотрудник может играть разную роль и влиять на исход проекта с разным уровнем эффективности.

Поскольку I_c не зависит от проекта, а является характеристикой сотрудника только с учетом его полезности для организации, уровня опыта и уровня качества его работы, необходимо ввести новый показатель, который будет выявлять полезность и эффективность привлечения конкретного сотрудника на конкретный проект. Данный показатель будет служить проектным агрегированным показателем сотрудника.

Расчет трудовых ресурсов IT-проектов осуществляется путем максимизации суммы проектных агрегированных показателей сотрудников проектных команд.

По результатам проведенного исследования, использование разработанного метода на проектах четырех организаций, в среднем, дало улучшение качества проектных команд на 7%, что соответствует средней экономии времени на разработку проектов на 8% и среднему снижению затрат на 7.3%.

Расчитанные показатели эффективности позволяют сделать вывод о том, что применение метода расчета трудовых ресурсов IT-проектов с использованием агрегированных показателей является целесообразным и экономически выгодным.

СПИСОК ЛИТЕРАТУРЫ

- Новиков, Д.А. Управление проектами: организационные механизмы / Д. А. Новиков. – М. : ПМСОФТ, 2007. – 140 с.
- В. Либерзон – «Основные понятия и принципы управления в области IT» [Электронный ресурс]. – Режим доступа: <http://www.osp.ru/cio/2000/03/170815>.
- Jesse Russell «Jira» / J. Russell – Книга по требованию, 2013. – 174 с.
- Система управления проектами Redmine [Электронный ресурс]. – Режим доступа: <http://nadir.kz/2014/02/27/redmine/>.

ОБ ОСНОВНЫХ АСПЕКТАХ РАЗРАБОТКИ ИНТЕРНЕТ-ПРИЛОЖЕНИЯ «ONLINE GALLERY»

Козловская Е. А., Рудикова Л. В.

Кафедра современных технологий программирования, Гродненский государственный университет имени
Янки Купалы

Гродно, Республика Беларусь

E-mail: liza.kozlovskaya.2011@gmail.com, rudikowa@gmail.com

В статье описываются этапы разработки системы «Online gallery». Приводится описание приложения, его цели и назначение, а также средства для реализации приложения.

ВВЕДЕНИЕ

В последние годы в мире наблюдается значительный рост пользователей в сети Интернет, а также разнообразных образовательных, информационных и развлекательных ресурсов. К числу образовательных ресурсов можно отнести и виртуальные арт-галереи. Тема переноса реальных галерей в режим «онлайн» является актуальной на сегодняшний день, так как современный темп жизни всё меньше и меньше даёт возможностей посещать такие культурные места: знакомиться с новыми лицами в мире искусства, узнавать новое об уже всеми известных произведениях, обмениваться впечатлениями с единомышленниками; однако такое приложение сможет решить эти проблемы и любой человек в любом месте земного шара сможет узнать что-то большее о различных произведениях искусства.

Данное приложение позволит собрать информацию о таких видах искусства как: живопись, архитектура, скульптура, фотография, кинематография и современное искусство. Благодаря многообразию видов творчества любой пользователь такой системы сможет найти для себя что-то по душе и обсудить в кругу единомышленников.

Интернет-приложение «Online gallery» также будет предоставлять информацию об уже популярных, но и новых лицах в искусстве, а также даст возможность приобрести на аукционе понравившееся произведение новых авторов.

1. ЭТАПЫ РАЗРАБОТКИ ПРИЛОЖЕНИЯ

При разработке приложения было решено разделить проект на семь этапов, который будут подробнее описаны ниже, для обеспечения создания всех необходимых функциональностей.

1. Анализ ресурсов схожей тематики и поиск источников информации о произведениях искусства;
2. Разработка архитектуры веб-приложения;
3. Создание макета пользовательского интерфейса;
4. Создание логической модели базы данных;
5. Проектирование физической модели базы данных и заполнение базы данных информацией о произведениях искусства;

6. Разработка серверной части приложения;

7. Разработка клиентской части приложения.

Первым этапом в разработке интернет-приложения является анализ ресурсов схожей тематики для выявления достоинств и недостатков, а также необычных функциональностей, которые будут учитываться при разработке веб-приложения «Online gallery». Также на данном этапе осуществляется поиск источников с подробным описанием произведений искусства. Так как приложение будет основано на наличие различного рода произведениях, то и поиск данных будет осуществляться из нескольких источников: личные галереи авторов, сайты знаменитых галерей, музеев и выставок, информационные сайты посвященные искусству.

Следующим этапом в разработке веб-приложения будет разработка архитектуры. Более подробно о выбранных технологиях и разработке архитектуры будет описано во второй главе.

Создание макета будущего пользовательского интерфейса приложения будет являться третьим этапом в разработке. Под макетом интерфейса понимается визуальное представление интерфейса пользователя. На этом этапе мы распределяем все важные элементы управления на веб-странице и формируем их поведение на пользовательское взаимодействие. При размещении элементов также будут проанализированы UX-составляющие: какое впечатление должен будет получать пользователь от работы с разрабатываемым интерфейсом, удастся ли ему достичь цели и на сколько легко или сложно будет ориентироваться на сайте.

Четвертым этапом в разработке является разработка логической модели базы данных. С помощью логической модели будут получено графическое представление исследуемой предметной области. Логическая модель даёт возможность выделить и графически представить сущности изучаемой предметной области, а также взаимоотношения между собой.

Проектирование физической модели базы данных и заполнение базы данных информацией о произведениях искусства, является пятым этапом разработки. Итогом этого шага является готовая полноценная база данных, которая будет

использоваться при разработке серверной части приложения.

Предпоследним шагом в разработке веб-приложения «Online gallery» является создание серверной части: разработка функции по обработке пользовательских запросов и формирование ответа, функции для работы с данными и т.д.

Создание клиентской части приложения будет финальным шагом создания приложения. Данный шаг основан на создании полноценного пользовательского интерфейса с помощью спроектированного макета интерфейса на третьем этапе

Таким образом в ходе всех выше упомянутых семи шагов было создано интернет-приложение «Online gallery».

II. ПОДХОДЫ К РЕАЛИЗАЦИИ АРХИТЕКТУРЫ ПРИЛОЖЕНИЯ

Приложение «Online gallery» является веб-приложением или клиент-серверным приложением, где пользователь «общается» с сервером посредством браузера. Таким образом обмен информацией между клиентом и сервером происходит по сети, что является плюсом в вопросе адаптации приложения, так как этот вид приложений является межплатформенным и не зависит от операционной системы пользователя.

В качестве платформы разработки приложения была выбрана ASP.NET: модель ASP.NET MVC Framework для создания веб-приложений, которая основана на шаблоне «Модель-Представление-Контроллер». Данная схема позволит разделить все данные на три части (бизнес-логика, данные и представление данных), которые могут модифицироваться независимо друг от друга.

Приложение состоит из следующих частей:

- Базы данных, где будут храниться вся информация о произведениях искусства, авторах, пользователях системы, а также данные аукциона;
- Серверной части приложения. Позволяет взаимодействовать пользователю с базой данных посредством пользовательского интерфейса;
- Клиентской части приложения с помощью которого предоставляется доступ к функциональным возможностям приложения.

Для создания полноценной клиентской части приложения были использованы: язык гипертекстовой разметки HTML5 для создания каркаса страниц, каскадные таблицы стилей CSS3 для оформления внешнего вида веб-страниц и JavaScript для придания веб-страницам приложения динамичности.

Выбранной СУБД для создания интернет-приложения «Online gallery» является Microsoft SQL Server 2017. Приложение «Online gallery» использует 15 таблиц.

Некоторые таблицы приведены ниже:

- Таблица «Artwork». Содержит в себе описание произведения искусства: название, оригинальное название, дату создания и др.;
- Таблица «Author». Хранит информацию об авторе произведения: имя, псевдоним, краткую биографию и др.;
- Таблица «GenreArtwork». Содержит все возможные жанры произведений искусства;
- Таблица «User». Содержит в себе данные всех зарегистрированных пользователей;
- Таблице «RoleUser». Содержит роли пользователей в приложении.

Для работы с базой данных был использован ORM-фреймворк Entity Framework на базе .NET для работы с данными. Концепцией данного фреймворка является понятие сущности, которая представляет собой набор данных, ассоциированных с определенным объектом. Поэтому данная технология предполагает работу не с таблицами, а с объектами и их наборами свойств.

Как упоминалось ранее приложение «Online gallery» будет иметь несколько типов ролей пользователей: «Администратор», «Модератор», «Пользователь», «Автор произведения» и «Гость». При регистрации автоматически присваивается роль «Пользователь». Каждая из ролей имеет свои возможности и дополнительные функции. Например, пользователь имеющий роль «Автор» имеет возможность самостоятельно добавлять свои работы в каталог, редактировать свою авторскую страницу, а также подавать заявку на участие работы в аукционе. В тоже время «Администратор» может раздавать роли другим пользователям, рассматривать заявки на участие в аукционе, а также при завершении аукциона связываться с покупателем. В возможности «Модератора» входит редактирование произведений в каталоге, ответы на вопросы пользователей под произведениями, а также осуществлять контроль над сайтом в целом.

III. Выводы

Таким образом в результате проделанной работы было разработано приложение, которое позволяет отслеживать современные тенденции в искусстве, а также знакомиться с работами новых мастеров. Одним из весомых преимуществ такой интернет-галереи является тот факт, что она позволяет предложить пользователю многообразие произведений искусства, которые могут привлечь как новичкам, так и профессионалам.

СПИСОК ЛИТЕРАТУРЫ

1. ASP.NET MVC 4. Разработка реальных веб-приложений с помощью ASP.NET MVC / Д. Чедвик, Т. Снайдер, Х. Панда. – 2013. – С. 94–100.
2. ASP.NET MVC 5 с примерами на C# 5.0 для профессионалов / А. Фримен. – 2015. – С. 25–28.

УПРАВЛЕНИЕ МЕХАТРОННЫМИ СИСТЕМАМИ ПЕРЕМЕЩЕНИЙ С ИСПОЛЬЗОВАНИЕМ ТЕХНОЛОГИИ ETHERCAT

Кузнецов В. В., Марко А. Ф.

Кафедра высшей математики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: mmts@bsuir.by

Рассмотрены подходы к управлению мехатронными многокоординатными системами в режиме реального времени, основанные на технологии EtherCAT. Управление с помощью протокола EtherCAT позволяет осуществить необходимое быстрое действие передачи данных обеспечивая при этом быстрое и качественное формирование заданных пространственных траекторий многокоординатной системы перемещений.

Объединение узлов точной механики с электронными, электрическими и компьютерными компонентами позволило осуществлять проектирование и производство качественно новых координатных модулей и мехатронных систем с интеллектуальным управлением, работающих в режиме реального времени [1–2]. Традиционные системы перемещений используют ПЛК-технологии для выполнения задачи управления и включают в свой состав некоторые аппаратные и программные элементы: ПЛК для визуализации, ПЛК с различными сопроцессорами, ввод/вывод через полевую шину, управление движением через параллельный интерфейс LPT, операционную систему и различные языки программирования.

В настоящей работе анализируется возможность использования технологии EtherCAT для управления мехатронными многокоординатными системами перемещений в режиме реального времени.

Существует множество примеров, когда осуществлялись попытки организовать управление в реальном времени через Ethernet [3]. Например, процедура доступа CSMA/CD (множественный доступ с контролем данных и обнаружением конфликтов) запрещает доступ через уровень протокола и разрешает управление через разделение времени или “pooling”. Другие подходы используют специальные ключи, которые пересылают сообщения по сети в точно определенное контроллером время. Эти решения обеспечивают передачу данных для узла Ethernet более-менее быстро и надежно, однако скорость передачи данных ограничена. Более того, время начала обмена строго фиксировано для устройств переадресации и зависит от состава и количества устройств в сети, что не позволяет эффективно управлять сложными устройствами (например, контроллерами управления движением различных механизмов).

Также в некоторых случаях рекомендуется использовать дополнительный канал управле-

ния, в частности для устройств управления вводом/выводом, для реализации быстрого и качественного алгоритма управления данными. При данном подходе передача данных для источника осуществляется быстро и синхронизировано, но с определенными задержками, что не всегда применимо для сложных устройств мехатроники.

С помощью рассматриваемой в статье технологии EtherCAT можно преодолеть перечисленные выше ограничения. Обработка данных практически не превышает время получения и обработки полученного Ethernet сообщения. Каждый управляемый контроллер читает данные, предназначенные для выбранного устройства, в момент прохождения сообщения через него. Пока сообщение проходит через контроллер (задержка всего лишь на несколько бит), контроллер распознает данные, предназначенные для выбранного устройства, и выполняет необходимые действия.

Ответные данные в соответствии с командами для устройства, также вставляются контроллером в Ethernet-сообщение «на лету», то есть в момент прохождения обратного сообщения по каналу Ethernet. Данный алгоритм обработки выполняется на аппаратном уровне и не зависит от реализации протокола обмена. Последнее устройство в сети EtherCAT возвращает полностью обработанное ответное сообщение для управляющего устройства, в соответствии с протоколом обмена по сети Ethernet.

С точки зрения Ethernet, EtherCAT-устройство представляет собой большое Ethernet устройство, которое принимает и посылает Ethernet-сообщения и представляет собой множество управляемых EtherCAT-устройств, подключенных к одному Ethernet контроллеру или переключателю Ethernet. В случае, когда управляемые EtherCAT-устройства подключаются непосредственно к контроллеру Ethernet, получается отдельная EtherCAT-система. Аппаратное обеспечение для управления EtherCAT-системами обеспечивается при помощи стандарт-

ных контроллеров управления доступом в среде передачи данных или сетевых интерфейсных плат. Устройство прямого доступа к памяти используется только для передачи данных в компьютер. Это означает, что доступ и обработка данных в EtherCAT-сетях не влияет на производительность компьютера.

Протокол EtherCAT оптимизирован для передачи управляющих данных посредством стандартной Ethernet телеграммы или вставляется непосредственно в UDP/IP пакет данных (рис. 1).

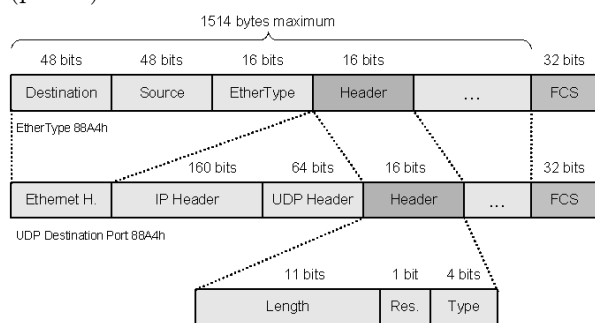


Рис. 1 – Структура пакета данных EtherCAT

Порядок передачи данных не зависит от расположения управляемых EtherCAT-устройств в системе, и они могут быть адресованы в любом порядке. Возможны любые варианты передачи управляющих данных: всем устройствам; одному или нескольким устройствам; между двумя устройствами EtherCAT.

Управляемое устройство сети EtherCAT представляет собой интеллектуальный контроллер, который обеспечивает прием и обработку данных, поступающих с управляющего устройства. По протоколу предусмотрена также и передача параметров устройства, которые обычно пересылаются в режимах, не требующих четкой фиксации времени обмена. Передача данных осуществляется при помощи почтовых ящиков (Mailbox).

Структура и значения параметров определяются посредством CANopen стандарта в части описания профиля устройства, которое поддерживается множеством производителей контроллеров во всем мире. Протокол так же поддерживает стандарт IEC 61800-7-204, который функции устройств управления серводвигателями, и используется для многих устройств управления перемещением под именем SERCOS.

В дополнение к обмену данными в сети по принципу «управляющий – управляемый», EtherCAT протокол так же очень удобен для организации обмена данных по принципу «управляющий – управляющий». Свободно адресуемые сетевые переменные для выполняемых данных и различные сервисы для параметрических данных, диагностики, программирования и удаленного управления создают необходимые условия для несложной организации обмена между управляющими контроллерами. При этом интер-

фейс передаваемых данных остается без изменения.

Технология EtherCAT открывает новые возможности для увеличения производительности обмена данными по сети. Время обработки данных для 1000 устройств ввода/вывода составляет только 30 мкс. До 1486 байтов управляющих данных могут быть обработаны в EtherCAT сети в составе одной Ethernet-телеграммы, что соответствует почти 12000 единичных сигналов ввода/вывода. При этом время обмена составляет всего 300 мкс.

Обмен управляющими данными между 100 сервомоторами занимает 100 мкс. В течение этого времени, координатные сервомоторы и гибридные шаговые двигатели мехатронных систем перемещений получают параметрические и управляющие данные и формируют ответное сообщение с данными о состоянии и реальном положении координатных модулей. Поддержка техники распределения времени позволяет синхронизировать управление приводами с точностью меньше 1 мкс, что необходимо для организации качественного управления координатными модулями по заданной траектории движения [1].

Чрезвычайно высокая скорость обмена данными позволяет использовать концепцию управления, которую нельзя реализовать на обычных сетевых протоколах. Быстрый цикл обработки управляющих данных, таким образом, может быть выполнен на уровне системы EtherCAT.

Функции обработки данных, которые рекомендовалось выполнять на уровне аппаратного обеспечения, могут быть перенесены на уровень программного обеспечения, что делает построение системы более гибким и прозрачным. Огромная пропускная способность EtherCAT системы позволяет получать все необходимые данные о состоянии каждого узла мехатронной системы за время, которое позволяет организовывать управление в режиме реального времени соответствующих многокоординатных приводов прямого действия. Технология EtherCAT обмена управляющими данными сопоставима по производительности обработки данных, с современными промышленными персональными компьютерами.

СПИСОК ЛИТЕРАТУРЫ

1. Карпович, С.Е. Системы многокоординатных перемещений и исполнительные механизмы для прецизионного технологического оборудования : моногр. / С.Е. Карпович [и др.] ; под. ред. д-ра техн. наук, проф. С.Е. Карповича. – Минск : Бестпринт, 2013. – 208с.
2. EtherCAT Technology Group: EtherCAT – шина Ethernet. [Электронный ресурс]. Режим доступа <http://www.ethercat.org/en/technology.html>. – Дата доступа: 20.12.2016.
3. Szigeti T. End-to-End QoS Network Design: Quality of Service for Rich-Media & Cloud Networks, 2nd Edition // Szigeti T., Hattingh C., Barton R., Briley K. – Cisco Press, 2013. – 1090с.

ПОВЫШЕНИЕ НАДЕЖНОСТИ АВТОМОБИЛЕЙ НА ЭТАПЕ ПРОЕКТИРОВАНИЯ

Куль В. А., Курулёв А. П., Островский В. А.

Кафедра теоретических основ электротехники

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: viktoriyakul7@gmail.com, alexparakuru@yahoo.com, ostrovskiy565@gmail.com

Целью настоящей работы является описание методов повышения надежности на этапах проектирования, комплекса мероприятий для каждого из этапов, а также рассмотрение наиболее важных факторов, влияющих на надежность автомобиля.

I. ВВЕДЕНИЕ

Современные автомобили включают в себя все больше электроники от блоков управления двигателем до антиблокировочных систем. Эти требования к комфорту и безопасности становятся настоящей головной болью для инженеров. Отказ автомобиля вследствие отказа детали или компонента недопустим, особенно если этот отказ может привести к травме или даже смерти.

Надежность становится одним из главных приоритетов на самых быстрорастущих рынках полупроводников, включая автомобильные, промышленные и облачные вычисления. Но вместо замены чипов каждые два-четыре года ожидается, что некоторые из этих устройств прослужат до 20 лет, даже при более интенсивном использовании в иногда экстремальных условиях окружающей среды.

II. ФАКТОРЫ, ВЛИЯЮЩИЕ НА НАДЕЖНОСТЬ

Получение повышенной надежности означает правильное проектирование для предполагаемых условий использования. Одно из основных направлений – это повышение стойкости изделий к внешним воздействиям, таким как: экстремальные температуры, вибрация, удары, влажность, туман. Типичный температурный диапазон автомобиля составляет от -40°C до 125°C . Требования к вибрации могут составлять до 3 г, а уровни удара – до 20 г. Они могут быть даже выше, если компонент устанавливается непосредственно на блок двигателя. Одним из аспектов окружающей среды, который упоминается не так часто, является проблема электромагнитных помех. С ростом использования электронных систем, защита от электромагнитных помех становится все более важным аспектом дизайна автомобиля.

Проблемы электромагнитного излучения и помехоустойчивости могут варьироваться от простого статического шума в радио до такой серьезной, как отказ антиблокировочной системы. Электромагнитные помехи генерируются каждой системой и это может привести к нарушению работы близлежащих или других взаимосвязанных систем. Основным источником большинства электромагнитных помех является встроенный

микропроцессор и его схема с малым временем нарастания сигналов. Также помехи могут создавать кабели, провода и печатные платы. Тем не менее, столь же серьезной проблемой являются внешние электромагнитные помехи, источниками которых являются вышки сотовой связи или воздушные линии электропередачи высокого напряжения. Другим источником электромагнитных помех является электрическая система автомобиля – различные реле, соленоиды, система зажигания.

Еще одним источником электромагнитных помех является электростатическое напряжение. Данное напряжение создается в результате взаимодействия водителя с салоном автомобиля. Электростатическое напряжение может вызвать излучение электромагнитного поля, которое может повредить электронику автомобиля.

III. МЕТОДЫ РЕАЛИЗАЦИИ НАДЕЖНОСТИ АВТОМОБИЛЯ НА РАЗЛИЧНЫХ УРОВНЯХ ПРОЕКТИРОВАНИЯ

Для каждой из проблем, описанных выше, существует предпочтительный метод защиты в зависимости от конкретной системы, ее конструкции и источника помех. Во многих случаях для обеспечения необходимого уровня защиты от электромагнитных помех необходимо использовать более одного метода. Эти методы также могут быть реализованы на одном или нескольких уровнях проектирования. К таковым уровням относятся: уровень микросхемы, печатной платы, модуля или корпуса и уровень межсоединений.

IV. УРОВЕНЬ МИКРОСХЕМ

Микросхема обычно является источником большинства шума, поэтому имеет смысл искать решение проблемы на этом уровне. Для получения лучшей производительности производители микросхем стремятся получить более высокие частоты. Они внедряют тактовые технологии с расширенным спектром, добавляя заземляющие плоскости и используя внутренние развязывающие конденсаторы на уровне микросхемы, чтобы повысить защищенность от электромагнитных помех. Если же микросхема не имеет

собственной защиты от помех, – необходимо использовать экранирование на уровне платы.

V. УРОВЕНЬ ПЕЧАТНОЙ ПЛАТЫ

Методы включают в себя разбиение, использование изолирующих переходов и маршрутизацию. Другие методы, которые включают в себя некоторые дополнительные затраты на компоненты, включают высокочастотное заземление платы, методы фильтрации и использование экранирования для изоляции компонентов, создающих помехи. Разработка экранирования на этапе проектирования ведет к минимальным затратам.

Если же говорить о лучших методах при создании печатных плат, то следует в первую очередь сгруппировать критические схемы (тактирование) и функции, что приведет к наименьшей длине трасс между компонентами. Также следует рассмотреть использование многослойных плит с использованием множества наземных плоскостей, надлежащих компонентов фильтрации и развязки. Не будет лишним разместить фильтрующие компоненты.

VI. УРОВЕНЬ МОДУЛЯ

На уровне модуля можно использовать экранирование. Однако это считается довольно таки дорогим способом защиты. В настоящее время используются пластмассы, краски или покрытия. Однако, существуют и более дешевые решения: экраны на уровне плат, термоформованная пластмасса. Тем не менее во многих случаях использование экранированного корпуса может быть единственным жизнеспособным решением для определенных типов проблем, связанных с электромагнитными помехами. Листовой металл может обеспечить затухание более чем на несколько сотен дБ даже в виде тонких слоев проводящего покрытия (например, до 60 дБ). Таким образом, использование прокладок становится необходимым. Выбор нужного типа прокладки зависит от окружающей среды. Если требуется герметизация окружающей среды,

следует использовать электропроводящую прокладку. Если проблемой является коррозия, то следует использовать кремний или неопрен.

VII. УРОВЕНЬ МЕЖСОЕДИНЕНИЙ

Кабели особенно уязвимы, когда заходит речь о весе и стоимости. Кабели, вероятно, являются следующим источником электромагнитного излучения после печатной платы. Однако добавление экранирования для проводов приведет к существенному увеличению веса и стоимости. Можно использовать фильтры, но это тоже дорого. Можно использовать ферриты, однако, это тоже является дорогостоящим решением, помимо этого увеличивается вес автомобиля, а также появляются проблемы с вибрацией. Если требуется фильтрация, лучшим подходом является добавление фильтров на интерфейс ввода-вывода платы. Для очень высоких частот можно также использовать поглощающее покрытие для кабеля.

VIII. ЗАКЛЮЧЕНИЕ

В большинстве случаев характеристики надежности нового автомобиля на этапе проектирования определены лишь приблизительно, т.к. невозможно гарантировать соблюдения их значений при эксплуатации.

Для решения главных задач надежности необходимо произвести ряд мероприятий на этапе проектирования: расчет и прогнозирование надежности автомобиля, регламентацию и обеспечение показателей, нормирование скоростей протекания процессов старения, резервирование отдельных элементов, разработку мероприятий по повышению восстанавливаемости.

СПИСОК ЛИТЕРАТУРЫ

1. Баженов Ю.В., Основы теории надежности машин / - Изд-во Владим. гос. ун-та, 2006.
2. Малкин В.С., Основы теории надежности и диагностики автомобилей / - Тольятти: ТГУ, 2005.
3. Курулёв А.П., Надёжность изделий промышленной электроники / - Минск, БГУИР, 2014.

ОСОБЕННОСТИ КЛАССИФИКАЦИИ ДЕФЕКТОВ ПОЛУПРОВОДНИКОВОЙ ПЛАСТИНЫ ПРИ АВТОМАТИЗИРОВАННОМ КОНТРОЛЕ

Лапко А. В.

Кафедра инженерной психологии и эргономики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: alexander.lapko@tut.by

В работе рассматриваются особенности классификации дефектов, возникающих при производстве полупроводниковой пластины, а также классификации, основанной на учете размеров дефекта, влияния дефекта на сформированную топологию пластины, причине возникновения дефекта или на цифровых параметрах, необходимых для автоматизированных систем, которые отличают дефектные области от бездефектных.

ВВЕДЕНИЕ

В процессе производства, полупроводниковые пластины подвергаются многочисленным воздействиям. Среди всего прочего, слои пластины могут добавляться, отшлифовываться, полироваться, удаляться, вытравливаться и очищаться. После каждого процесса обработки пластины, она должна быть проверена, чтобы убедиться, что предыдущий процесс был завершен с допустимым уровнем погрешности. Многочисленные операционные переменные (такие как длительность обработки, давление газа, концентрация веществ, температура обработки и т.д.) каждого процесса производимого над пластиной должны быть сохранены, чтобы при необходимости любые изменения какой-либо переменной могли быть быстро идентифицированы и проверены на предмет потенциальной взаимосвязи с возникшим дефектом при контроле пластины после обработки [1].

Характеристики дефектов могут быть использованы, чтобы определить их влияние на процессы полупроводникового производства и на показатели готового устройства (тепловые характеристики, электрические показатели устройства, параметры производительности).

I. БАЗОВАЯ РАЗМЕРНАЯ КЛАССИФИКАЦИЯ ДЕФЕКТОВ

Одновременно с совершенствованием технологического процесса и уменьшением формируемых на пластине элементов, уменьшаются и возникающие погрешности. Вследствие случайного положения таких «физических» дефектов, вызываемые ими искажения топологического рисунка могут быть весьма разнообразными. Таким образом, по размерам и характеру распределения по площади пластины дефекты могут быть подразделены на микродефекты и макродефекты.

Микродефекты – это дефекты, связанные с нарушением топологического рисунка слоя на площади, характерный размер которой меньше

или сравним с минимальным (или проектным) размером топологического элемента на данном слое или с расстоянием между соседними топологическими элементами. Распределение микродефектов по площади пластины носит случайный характер.

Макродефекты – это обширные нарушения топологии, вызванные одной причиной и локализованные на площади, характерный размер которой существенно превышает минимальный размер топологического элемента на данном слое [2].

II. ТОПОЛОГИЧЕСКАЯ КЛАССИФИКАЦИЯ ДЕФЕКТОВ

Рассматриваемая далее классификация в некоторой степени не учитывает причины возникновения дефектов и описывает конечный результат взаимодействия «физических» дефектов с топологией слоя, т.е. направлена на описание свойств объекта (кристалла) после проведения технологической операции формирования топологического рисунка. Она позволяет независимо от топологии слоя характеризовать состояние дефектности схем различной степени интеграции и в то же время для каждого конкретного слоя позволяет установить взаимосвязь «физического» уровня дефектности с конкретными и характерными топологическими дефектами.

Микродефекты можно разбить на следующие виды в зависимости от их топологических особенностей:

- замыкание соседних элементов топологии;
- обрыв (нарушение сплошности) топологического элемента;
- нарушение формы топологического элемента, включая выступы и вырывы;
- включения – инородные включения в материал слоя;
- внешнее загрязнение в виде инородного материала на поверхности слоя;
- поры или проколы в материале слоя;

- ложные элементы – остатки материала слоя в вытравленных областях;
- отсутствие элементов (элемента) слоя.

Следует отметить, что дефекты, связанные с отклонением размера от номинала и с превышением допуска по совмещению, т.е. изменения элементов, происходящие по всей площади кристалла, к локальным дефектам не относятся и считаются технологическим браком.

III. КЛАССИФИКАЦИЯ ДЕФЕКТОВ ПО ПРИЧИНЕ ВОЗНИКНОВЕНИЯ

Макродефекты проявляются, как протяженные области площади кристалла, где в той или иной степени нарушена топология элементов. Их классификация может быть основана на причинах, их вызвавших. Эти дефекты, как правило, связаны с особенностями или нарушениями при проведении технологических операций формирования топологического рисунка [2]. К ним можно отнести:

- царапины;
- крупные инородные включения или загрязнения в виде тонких пленок;
- перетравы и недотравы приводящие к изменению формы и/или линейных размеров элементов.

IV. КЛАССИФИКАЦИЯ ДЕФЕКТОВ ПРИ АВТОМАТИЗИРОВАННОМ КОНТРОЛЕ ПЛАСТИНЫ

С точки зрения автоматизированного обнаружения дефектов необходимо перейти от приведенной выше «физической» классификации дефектов к классификации, основанной на признаках или характеристиках, отличающих дефектные области от бездефектных. Анализ природы дефектов полупроводниковых пластины показывает, что можно предложить следующие признаки, характеризующие дефекты:

- наличие «аномалий» интенсивности; (как по диапазону длин волн, так и по пространственному распределению)
- наличие «аномалий» ориентации контуров элементов;
- наличие «аномалий» в спектре пространственных частот.

Под аномалией понимается наличие на фрагменте изображения неразрешенных значений указанных характеристик. Например, наличие областей с коэффициентом отражения света отличным от коэффициентов отражения топологического слоя и подстилающей поверхности, наличие контуров с ориентацией, недопустимой по нормам проектирования и т.д. В качестве примера на рисунках 1–2 показаны типичные дефекты двух классов. Предложенная классификация дефектов в одинаковой мере применима к макро- и микродефектам. Макродефектом может быть внешнее загрязнение на рисунке 1. Такой макродефект может быть обнаружен по отклонению интенсивности. Микродефектом может быть на-

рушение геометрии типа «вырыв» (рисунок 2), который обнаруживается по аномальной ориентации контура элемента.

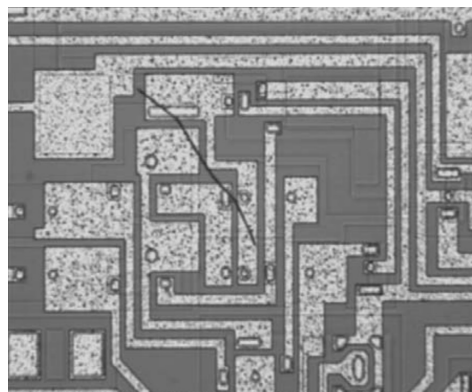


Рис. 1 – Пример дефекта класса «Аномалия интенсивности» типа «внешнее загрязнение»

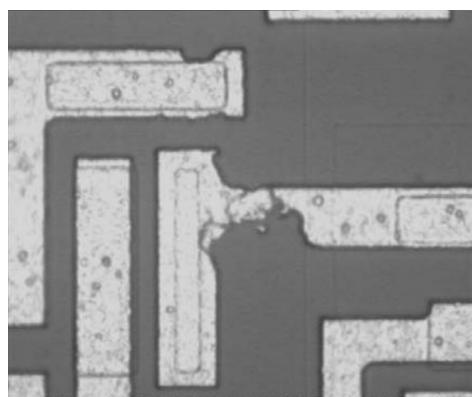


Рис. 2 – Пример дефекта класса «Аномалия контуров», нарушение геометрии типа «вырыв»

ЗАКЛЮЧЕНИЕ

Точно сформированная классификация дефектов полупроводниковых пластин способна систематизировать возникающие погрешности, в зависимости от текущего этапа производства, вида производимых измерений, особенностей технологического процесса и множества других параметров. Что в свою очередь делает более простым процесс идентификации источника дефекта и позволяет увеличить процент выхода годных изделий, а также формирует наиболее точные требования для разработки контрольно-измерительного оборудования.

СПИСОК ЛИТЕРАТУРЫ

1. User interface for wafer data analysis and visualization / Lam Research Corp. [Электронный ресурс]. – Режим доступа: <https://patents.google.com/patent/US7945085>. – Дата доступа: 25.09.2019.
2. Аваков, С. М. Методы детектирования и классификации макродефектов поверхности полупроводниковых пластин с топологией и их реализация в оборудовании для автоматического контроля // Сборник тезисов докладов «Прецизионное оборудование и технологии производства изделий микро- и радиоэлектроники». – Минск, Беларусь. – 2004. – № 1. – С. 18–21.

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ГИБКОГО ШТАМПОВОЧНОГО ПРОИЗВОДСТВА

Лукиянец С. В., Снисаренко С. В.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: kafsu@bsuir.by

Приводятся результаты имитационного моделирования гибкого производства деталей из листа на базе роботизированной четырехпрессовой линии, работающей в различных режимах.

ВВЕДЕНИЕ

При модернизации штамповочного производства важно оценивать зависимости производительности от размеров партий деталей и их номенклатуры, а также значения коэффициентов использования оборудования и коэффициентов занятости работников за плановый период. Рассмотрению этих вопросов посвящена работа [1], в которой исследовано функционирование четырехпрессовой линии при одинаковой сложности всех партий деталей. Представляет интерес работа линии в других режимах: наряду с изготовлением сложных деталей при использовании четырех прессов производство деталей средней сложности на трех прессах и простых – на двух прессах. Выявлению упомянутых зависимостей при функционировании такой линии в перечисленных режимах и посвящена настоящая работа. В ней используется подход, применявшийся в упомянутой статье и в работе [2].

I. КОМПОНОВОЧНАЯ СХЕМА И ФУНКЦИОНИРОВАНИЕ ЛИНИИ

Компоновочная схема линии, обеспечивающей изготовление деталей различной степени сложности, показана на рис. 1.

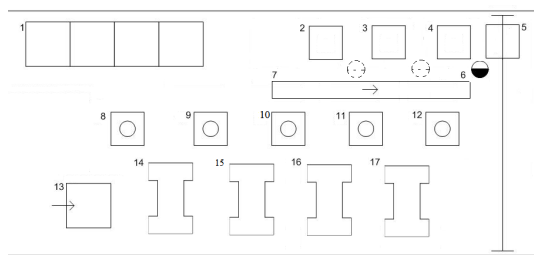


Рис. 1 – Компоновочная схема линии

На этом рисунке: 1 – склад; 2, 3, 4 – позиции загрузки в тару деталей соответственно простых, средней сложности и сложных; 5 – кран; 6 – оператор, находящийся на позиции, соответствующей уровню сложности деталей; 7 – лента, перемещающая готовые детали от пресса к оператору; 8, 9, 10, 11, 12 – роботы; 13 – позиция подачи заготовок в линию; 14, 15, 16, 17 – прессы.

Отличие предлагаемой компоновки от рассмотренной в [1] состоит в том, что лента обес-

печивает доставку готовых деталей к позициям, соответствующим работе двух, трех или четырех прессов. Работа линии осуществляется с учетом отмеченной особенности, в основном повторяя последовательность операций, изложенную в [1]. Назовем вариант, рассмотренный ранее, базовым.

II. ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ РАБОТЫ ЛИНИИ

Алгоритм моделирования работы линии при производстве на четырех прессах сложных деталей совпадает с алгоритмом моделирования базового варианта. Исключая из схемы алгоритма блоки, соответствующие работе и оснащению пресса 17 и обслуживанию этого пресса транспортом, и учитывая нахождение оператора на позиции 3, получена схема моделирования трехпрессовой линии. Поступая далее аналогичным образом, исключая из схемы алгоритма также блоки, связанные с работой пресса 16 при учете работы оператора на позиции 2, получена схема моделирования двухпрессовой линии. В соответствии с разработанными алгоритмами составлены программы их реализации на языке моделирования GPSS [3].

Особенности кода программы базового варианта [1] сохраняются и в кодах моделирования трехпрессовой и двухпрессовой линий. Окончательные коды этих программ получены из кода базового варианта путем исключения из них операторов, моделирующих блоки, удаленные из базового алгоритма при переходе к трехпрессовой и двухпрессовой линиям.

Объем выпуска деталей N за плановый период определялся количеством обработанных партий (номенклатурой деталей) и размером каждой партии, а номенклатура деталей - исходя из времени, затраченного на выпуск каждой партии, и длительности планового периода.

Для сопоставимости результатов моделирования работы линии в различных режимах исходные данные с учетом сделанных для базового варианта выводов о минимальной емкости тары под готовые детали и приемлемых размерах партий заготовок, приняты в основном прежними и приведены в табл.1.

Таблица 1 – Исходные данные

№ п/п	параметр/ед. изм.	Значение
1	Время установки/снятия штампа,с	3000 ± 300
2	Время доставки краном одного груза,с	300 ± 30
3	Время штамповки одной детали каждым прессом,с	3.8 ± 0.2
4	Время доставки одной заготовки на позицию 13,с	5
5	Время доставки одной детали лентой,с	6
6	Время работы каждого робота на одной операции,с	2.9 ± 0.2
7	Время работы оператора на одной операции,с	3.5 ± 0.5
8	Размеры тары,шт	200;400
9	Размер партии деталей,шт	800;1200;4000; 8000;12000; 16000;20000; 24000;28000; 32000

В результате компьютерного моделирования получен ряд зависимостей, характеризующих работу линии в различных режимах. Зависимости значений номенклатуры изделий от размеров партии заготовок представлены на рис. 2.

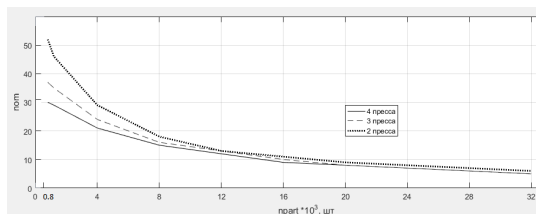


Рис. 2 – Зависимости значений номенклатуры изделий от размеров партии заготовок

Зависимости производительности линии от номенклатуры изделий при штамповке простых, средних и сложных деталей показаны на рис. 3. Из их анализа следует, что максимальная производительность линии во всех режимах достигается при значениях номенклатуры, не превышающих 10. При этом для простых деталей она выше на 20%, чем в других режимах.

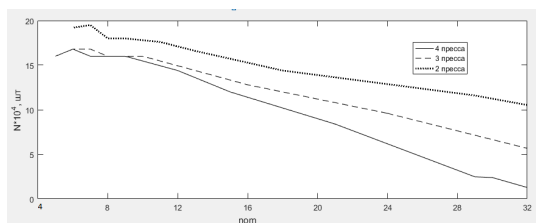


Рис. 3 – Зависимости производительности линии от номенклатуры изделий

При увеличении номенклатуры изделий до 20 единиц производительность двухпрессовой линии снижается на 27% от максимального значения;

при таком же увеличении номенклатуры изделий средней сложности снижение производительности составляет около 20% от своего максимального значения, а для сложных изделий – около 50%. Совместное рассмотрение рис. 3 и рис. 4 показывает, что во всех режимах наибольшей производительности соответствуют размеры партий деталей, превышающих 20000 единиц, при этом номенклатура изделий не превышает 10.

Исследованы зависимости коэффициентов загрузки прессов и крана, а также бригады, роботов и оператора от размера партий деталей. Результаты этих зависимостей для трех режимов работы линии при размерах партий деталей, превышающих 20000 шт., следующие. При переходе от сложных деталей к простым при изменении размеров партий от 20000 до 32000 шт. коэффициент загрузки прессов повышается от 60...86% до 93...95%, а коэффициент загрузки крана увеличивается от 58...68% до 68...74% для $n_{tar}=200$ и от 32...34% до 36...38% для $n_{tar}=400$. Коэффициент загрузки бригады уменьшается от 17...25% до 10...15%. Коэффициенты загрузки роботов и оператора увеличиваются: роботов от 58...63% до 66...68%, а оператора от 70...74% до 80...83%.

Результаты исследований целесообразно учитывать при организации работы гибкой линии штамповки в различных режимах.

ЗАКЛЮЧЕНИЕ

Выполнено имитационное моделирование гибкого производства деталей из листа на базе роботизированной четырехпрессовой линии, работающей в режимах выпуска простых, средних и сложных деталей. Особенностью производства является использование в качестве транспортного средства крана и бригады для оснащения прессов. На основе разработанных алгоритмов моделирования и программ для их реализации проведены компьютерные эксперименты по выявлению основных соотношений между производительностью, загрузкой оборудования и работников, размерами партий деталей и их номенклатурой за плановый период. Полученные результаты направлены на эффективное использование штамповочного производства рассмотренной конфигурации.

СПИСОК ЛИТЕРАТУРЫ

1. Лукьянец, С. В. Имитационное моделирование четырехпрессовой линии штамповки / С. В. Лукьянец, С. В. Снисаренко, М. А. Лишай // Докл. БГУ-ИР. 2018. №2(112) с. 92–97.
2. Лукьянец, С. В. Имитационное моделирование гибкого участка штамповки деталей из листа /С. В. Лукьянец, А. В. Павлова // Докл. БГУИР. 2008. №2(32).с. 105–110.
3. Боев, В. Д. Моделирование систем. Инструментальные средства GPSS World. / В. Д. Боев // Спб. – 2004. 368 с.

ДАТЧИК-ИНДИКАТОР СОСТОЯНИЯ ЭЛЕКТРИЧЕСКОЙ СЕТИ И ПОТРЕБИТЕЛЯ ЭЛЕКТРИЧЕСКОЙ ЭНЕРГИИ ДЛЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ

Малашенков А. А.

Кафедра электронных вычислительных средств, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: antlik2018@gmail.com

В статье описан принцип построения датчика-индикатора состояния электрической сети и потребителя электрической энергии. Устройство предназначено для индикации нормального состояния объекта управления (потребителей электрической энергии), определение основных неисправностей, как в сети энергоснабжения, так и в самом потребителе электрической энергии, передача данных о состоянии объекта управления на контроллер.

ВВЕДЕНИЕ

В электроснабжении многих узлов очень важно контролировать наличие входного и выходного напряжения, а также наличие потребления электрической энергии, которая непосредственно связана с протеканием тока через объект. Особенно это важно для объектов, которые нуждаются в бесперебойной подаче электроэнергии. Так как на большинстве объектах, используемых в настоящее время, уже смонтированы распределительные шкафы с традиционными автоматическими выключателями, то для решения задач управления требуется контроль состояния того или иного объекта для создания компьютеризированных, автоматизированных систем управления. Зачастую это связано с задачами диспетчеризации.

I. ДАТЧИК-ИНДИКАТОР НАПРЯЖЕНИЯ

Датчик-индикатор напряжения выполнен на базе оптрона PC817, которая с одной стороны обеспечивает гальваническую развязку между контроллером и датчиком, с другой стороны служит индикатором наличия напряжения в требуемом диапазоне (см. рис. 1).

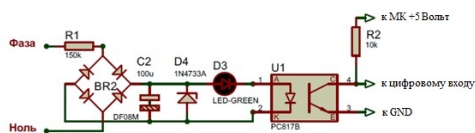


Рис. 1 – Схема датчика-индикатора напряжения для микроконтроллера

Датчик-индикатор напряжения формирует на выходе уровень логического «0» при наличии напряжения питания 220 или 230 В и уровень логической «1» при его отсутствии. Схема работает следующим образом. При наличии переменного напряжения на входе светодиод питается через выпрямитель собранный на диодном мосте BR1 (DF08M) и ограничительном резисторе R1

(150кОм) [1]. Светодиод HL1 обеспечивает визуальную индикацию. Далее если светодиод оптрона зажат, следовательно, транзистор оптрона открыт и на выходе схемы логический «0». Конденсатор является балластным сопротивлением для стабилитрона D2(1N4733A на 5.1 вольт) в сетевом выпрямителе. Если переменное напряжение на входе отсутствует, ток через светодиод не протекает, он погашен, следовательно, транзистор закрыт и на выходе схемы логическая «1». R2-выступает в качестве подтягивающего резистора.

II. ДАТЧИК-ИНДИКАТОР ТОКА

Выход датчика тока подключается к контроллеру через преобразователь ток-напряжение. В контроллере осуществляется аналогово-цифровое преобразование. Диапазон аналогово-цифрового преобразователя (АЦП) устанавливается согласно диапазону измеряемого действительного значения тока, протекающего через потребителей электрической энергии (см. рис. 2)[2].

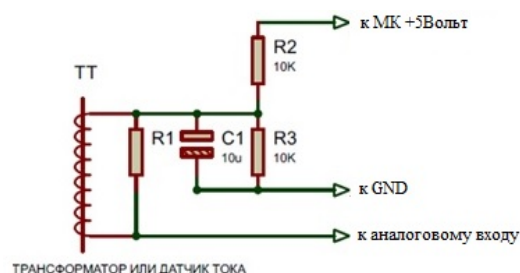


Рис. 2 – Схема подключения трансформатора тока через делитель напряжения для микроконтроллера

R1- нагрузочный резистор. R2,R3- делитель напряжения в данном случае номиналы резисторов 10кОм. C1- емкость 10 мкФ используется для сглаживания помех, а также размещение конденсатора на нижнем резисторе в делителе потенциала дает возможность уменьшить полное сопротивление источника без увеличения напряже-

ния смещения постоянного тока. Основная часть скрипта для расчета тока протекающего через потребитель приведена ниже:

```
SensorSensitivity это чувствительность трансформатора тока или датчика тока равная 37.0 // (185 мА для 5 ампер, 1 ампер 37.0)
unsigned long startTime = millis(); //количество миллисекунд с момента начала выполнения программы
while ((millis() - startTime) < 70) { // Выполняем считывание показаний до 70 раз
    readVal = analogRead(pin); // Считываем значение датчика
    if (readVal > maxVal) { //Проверяем, если значение датчика больше 0,
        maxVal = readVal; } // то записываем новое значение
    if (readVal > minVal) { //Проверяем, если значение датчика меньше 1023,
        minVal = readVal; } // то записываем новое значение
    maxVal -= maxVal * 0.00195; // Корректируем максимальное значение
    minVal += minVal * 0.00195; //Корректируем минимальное значение
    sensRMS = (((maxVal) - minVal) /2.0) / SensorSensitivity) * 0.707; // где 0.707 это корень из 2. По данному расчету получаем результат действующего значения тока
```

III. ПРИМЕР СТРУКТУРНОЙ СХЕМЫ СИСТЕМЫ УПРАВЛЕНИЯ ТЕХНОЛОГИЧЕСКИМ ОБОРУДОВАНИЕМ ПРЕДПРИЯТИЯ С ИСПОЛЬЗОВАНИЕМ ДАТЧИКОВ -ИНДИКАТОРОВ



Рис. 3 – Пример структурной схемы системы управления технологическим оборудованием предприятия с использованием датчиков -индикаторов

Из рисунка 3 видно, что датчики подключаются через мультиплексор в случае необходимости многоканального контроля. Например можно используется 16 канальный аналоговый мультиплексор, который собран на микросхеме CD74HC4067 и позволяет значительно расширить количество каналов для контроля, задействуя при этом небольшое количество выводов портов ввода/вывода микроконтроллера.

Подключение системы датчиков -индикаторов осуществляется в распределительном щите. Перед вводным автоматом подключа-

ется токовая петля (трансформатор тока T_{Tin} – вводной). Фазный контакт этого же автомата подключается к датчику -индикатора напряжения датчику присваивается имя Vin. Аналогичным образом подключаются выходные клеммы автоматических выключателей каждой группы к своим источникам нагрузки. Каждому датчику присваивается имя Vout. Блок -схема алгоритма работы датчиков -индикаторов состояния электрической сети представлена ниже (см. рис. 4).

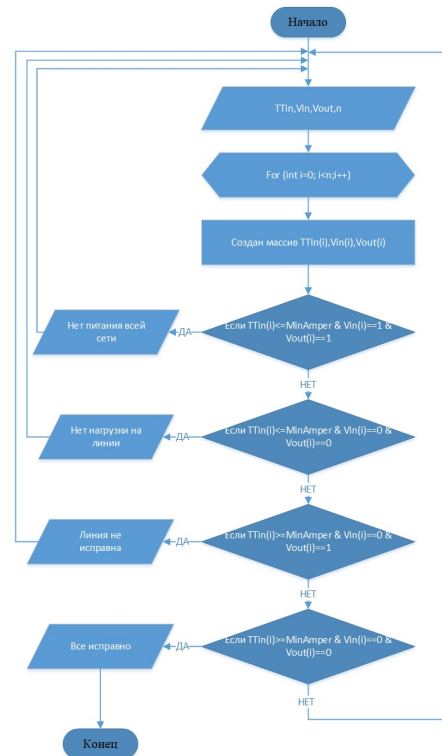


Рис. 4 – Блок -схема алгоритма работы датчиков -индикаторов состояния электрической сети

С датчиков поступают сигналы: T_{Tin}, Vin, Vout. Показания датчиков хранятся в массиве данных. В результате анализа состояния датчиков формируется данные о состоянии энергосетей и потребителей электрической энергии:

Нет питания всей сети, нет нагрузки на линии, линия не исправна, все исправно.

Преимущество предлагаемой системы заключается в отказе от использования релейных схем и возможности контроля тока потребителя электрической энергии. Предлагаемое устройство может найти широкое применение в автоматизированных системах управления технологическим оборудованием предприятия.

СПИСОК ЛИТЕРАТУРЫ

1. 1000 и одна микронтроллерная схема. Вып.1/ С. М. Рюмик. –М.: Додэка-XX1, 2010. –356 с.: ил. + СЭ. –(Серия «Программируемые системы»), - 18ВЫ 978-5-94120-211-9.
2. CT Sensors - Interfacing with an Arduino [Электронный ресурс] – Режим доступа: <https://learn.openenergymonitor.org/electricity-monitoring/ct-sensors/interface-with-arduino?redirected=true> – Дата доступа: 09.10.2019.

АНАЛИЗ РАСПРЕДЕЛЕНИЯ НАГРУЗКИ УЧЕБНЫХ ЛАБОРАТОРИЙ УО БГУИР

Марков А. Н., Савицкая Д. Г., Парамонов А. И.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: anm@bsuir.by, savitskaja@bsuir.by, anton_paramonov@tut.by

В работе рассматриваются результаты опытной эксплуатации внутреннего частного облака на базе технологии VMware vCenter решением VMware Horizon. Выполнен анализ показателей производительности нагрузок учебных лабораторий УО БГУИР.

Облачные вычисления сегодня активно применяются в различных отраслях. Их потребители могут значительно уменьшить расходы на инфраструктуру информационных технологий и гибко реагировать на изменения вычислительных потребностей. В тоже время, балансировка нагрузки в облачных вычислительных системах все еще является значительной проблемой. В работе предлагается оценить возможности внедрения облачных вычислений в организацию работы учебных лабораторий.

С 1 сентября 2019 года на базе Центра информатизации и инновационных разработок для общеуниверситетских компьютерных классов запущено внутреннее частное облако на базе технологии VMware vCenter решением VMware Horizon [1–2]. Технология подразумевает работу студентов по различным дисциплинам посредством доступа к полноценной операционной системе и приложениям посредством обращения к серверу через тонкие клиенты. Для работы преподавателей установлено два класса по 15 тонких клиентов и соответственно по 15 терминальных сессий на виртуальные сервера (termsrvclass9 и termsrvclass11).

Аппаратные составляющие на один терминальный сервер класса: CPU – Intel Xeon 2 процессора по 2.2 ГГц 6 ядер, 16 Гб RAM, HDD – 230 Гб. Реальная загрузка памяти в режиме ожидания – 6 Гб из 16 Гб. Однако при проведении лабораторных работ (работа с процессором и памятью) сказывается высокая потребляемость ресурсов виртуальных серверов. Собрана статистика работы 30 виртуальных машин за недельный период (см. рис.1). На рисунке представлена загруженность центрального процессора в процентном отношении за весь период мониторинга.

По графику видно, что при проведении по расписанию занятий, связанных с программированием, во время которых компиляция проекта напрямую зависит от мощности центрального процессора, загруженность растет, но не достигает даже половины выделенной мощности. Максимальное наблюдаемое значение – 11 %.

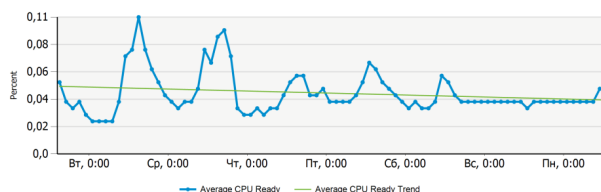
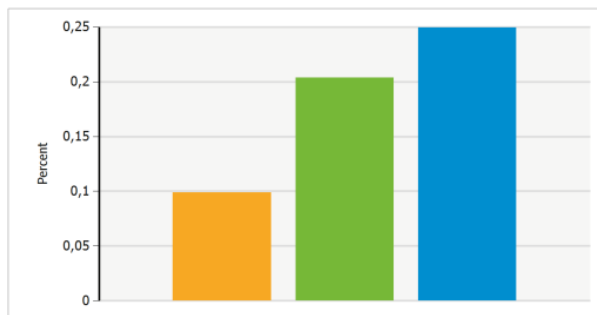


Рис. 1 – Показатель загруженности процессора (среднее значение)

По полученным через vCenter графикам можно построить рейтинг самых загруженных машин, и уже на его основе выделить топ 3 (см. рис.2). Анализ установленного на машинах программного обеспечения позволит найти проблемы в перегрузках хостов.

Проведя анализ показателей загрузки центрального процессора для данных виртуальных машин, можно сделать вывод, что производительности оборудования дата-центра, а также выделенных мощностей на виртуальные сервера вполне достаточно для решаемых задач.



Object Name	Avg	Trend
terminalsrv2	0,25	Increasing
termsrvclass9	0,20	Increasing
termsrvclass11	0,10	Increasing

Рис. 2 – Топ 3 загруженных виртуальных машин

Однако, стоит отметить, что хотя производительности процессора и достаточно для выполняемых задач, но существует другая проблема – нехватка оперативной памяти, на что указывают данные графиков использования опера-

тивной памяти как среднего значения, так и для сервера в целом (см. рис.3).

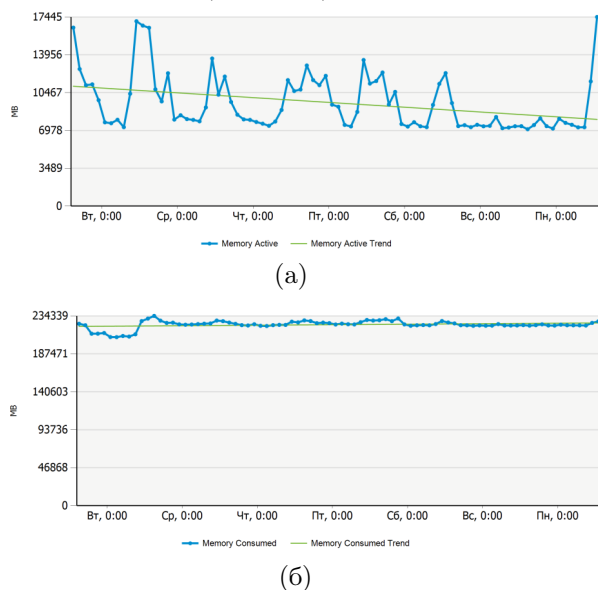
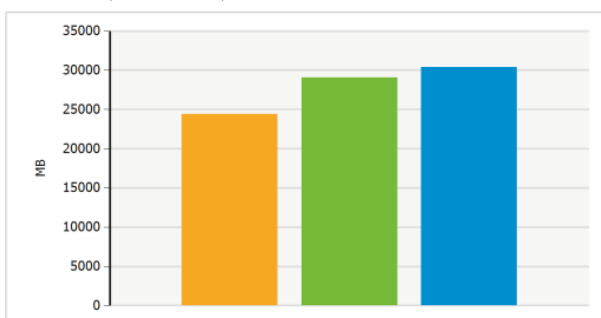


Рис. 3 – Использование памяти (уровень загрузки) в разрезе недели: (а) на виртуальных машинах; (б) на сервере

Из графика видно, что ресурсы оперативной памяти, которые выделены на Horizon приближаются к пиковым значениям. Даже при минимальной загруженности используется 25 Гб памяти (см. рис.4).



	Object Name	Avg	Trend
	termsrv-hor2	30361,86	Increasing
	termsrvclass2	29046,13	Decreasing
	termsrv-hor1	24390,25	Increasing

Рис. 4 – Загрузка памяти для терминальных классов

При этом в работе системы хранения данных (СХД) и локальной сети большие загрузки не наблюдаются (см. рис.5).

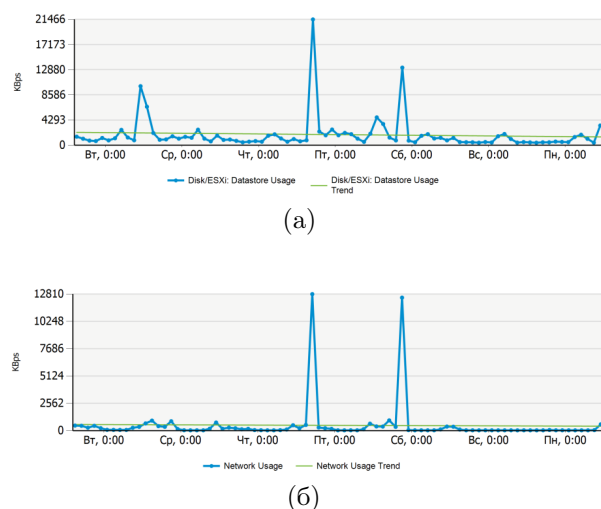


Рис. 5 – Уровень загрузки при работе ВМ: (а) СХД; (б) сетевого оборудования

Выбросы в показаниях графиков в виде пиковых значений загрузки оборудования объясняются временными курсами ЕРАМ, которые проводятся в указанное время (во вторую смену) на базе общеуниверситетских компьютерных классах.

Таким образом, можно сделать вывод о целесообразности перехода с персональных электронно-вычислительных машин на тонкие клиенты и организации работы их через внутреннее гибридное облако в учебных лабораториях, что подтверждается стабильной работой ВМ и распределение нагрузки на оборудование в целом. Дальнейшее исследование будет направлено на изучения вопросов динамического распределения нагрузки на оперативную память.

СПИСОК ЛИТЕРАТУРЫ

1. Peter, v. O. Mastering VMware Horizon 7 / v. O. Peter// N. Engl. – 2016. – В3. 2РВ
2. Lees, K. Operationalizing VMware vSAN / K. Lees// N. USA. – 2016. – VMware Press.

ПРИМЕНЕНИЯ ЦИФРОВЫХ ДВОЙНИКОВ И БЕССЕРВЕРНОЙ АРХИТЕКТУРЫ ДЛЯ УПРАВЛЕНИЯ ИОТ УСТРОЙСТВАМИ

Нестеренков С. Н., Кукареко А. В.

Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: nsn@bsuir.by, sasha.kukareko@gmail.com

Рассматриваются возможности применения цифровых двойников и бессерверной архитектуры для управления устройствами интернета вещей (IoT).

ВВЕДЕНИЕ

Благодаря развитию стандартов передачи информации (LTE 4G, 5G), технологий и удешевлению различных компонентов для сбора информации активно растет число устройств «подключенных» к интернету.

Интернет Вещей (Internet of Things – IoT) – это концепция, в которой устройствам (объектам) реального мира, оснащенным различными датчиками и доступом в интернет, предоставлена возможность автономного обмена полезной информацией между собой, сервером и другими потребителями. Согласно публикации IDC количество IoT устройств в 2019 году продолжает расти устойчивыми темпами и по прогнозам к 2025 году может достигнуть 41,6 млрд устройств, которые будут генерировать 79,4 зеттабайт (ZB) данных [1].

IoT технологии успешно применяются в различных сферах, что ведет к уменьшению затрат, повышению качества услуг, товаров и повышению безопасности. Сферы применения технологии варьируются от производства, сельского хозяйства, банковского дела, здравоохранения, логистики, транспорта, образования до умных лампочек, домов и городов [2].

I. Цифровой двойник

Цифровой двойник (ЦД) (Digital Twin) – это динамическое виртуальное представление физического объекта или системы в течение всего жизненного цикла с использованием данных в режиме реального времени для понимания, изучения и рассуждения [3].

Концепция цифрового двойника заключается в том, что он следует жизненному циклу, данным и состоянию связанным с устройством. Цифровые двойники обеспечивают такие функции, как моделирование устройства во время разработки, интеграция аналитики и машинного обучения, а так же контроль физического устройства.

Впервые понятие «цифровой двойник» появилось в 2003 году в статье «Цифровые двойники: превосходство в производстве на основе вир-

туального прототипа завода» профессора Технологического университета Флориды Майкла Гривза [4]. Благодаря росту применения IoT технологий и устройств во многих сферах, концепция «цифрового двойника» в 2018 году выбилась в тренды IoT технологий. Gartner включил ЦД в число лучших IoT технологий 2018 [5].

На сегодняшний день многие крупные провайдеры облачных решений, такие как Amazon Web Services и Microsoft Azure, предоставляют готовые сервисы для подключения IoT устройств и сервисы цифровых двойников. Так же они предоставляют удобные наборы средств разработки (SDK – software development kit) которые позволяют значительно ускорить как разработку прототипов, так и запуск целых IoT систем.

AWS IoT – использует термин «device shadow» для обозначения ЦД. Device shadow – это JSON файл, который содержит информацию о состоянии, метаданные, уникальный клиентский токен и версию устройства. Для взаимодействия с ЦД можно использовать протоколы HTTP и MQTT. Схему взаимодействия физического устройства и ЦД на платформе AWS IoT можно увидеть на Рис. 1.

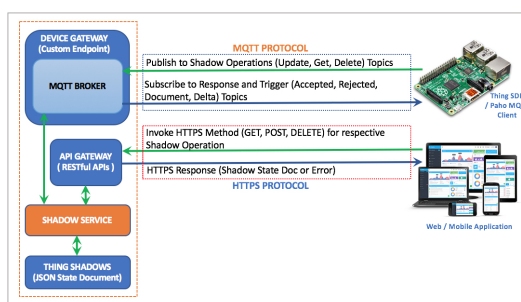


Рис. 1 – Взаимодействие устройств с цифровыми двойниками

Microsoft Azure IoT – использует термин «device twins». Device twins – это JSON файл, в котором хранится информация о состоянии устройства, включая метаданные, конфигурации и условия. Обязательными атрибутами device twin являются desired properties, reported properties и tags. Desired properties – это свойства устанавливаемые бек-энд приложением

и читаемые физическим устройством. Reported properties – это свойства устанавливаемые физическим устройством и читаемые бек-энд приложением. Для взаимодействия с ЦД можно использовать протоколы HTTP, MQTT и AMQP.

Несмотря на то, что и у AWS, и у Microsoft ЦД представлен в виде JSON файла, главным недостатком является отсутствие общепринятой спецификации касательно взаимодействия устройства с ЦД. Каждый провайдер реализует технологию по-своему, что усложняет смену изначально выбранной платформы и провайдера.

Преимущества использования ЦД заключаются в том, чтобы обеспечить уровень абстракции, который позволяет приложениям взаимодействовать с физическими устройствами, отслеживать их актуальное состояние, управлять ими и устанавливать желаемое «будущее» состояние.

II. ПРИМЕНЕНИЕ БЕССЕРВЕРНОЙ АРХИТЕКТУРЫ

Бессерверные вычисления (serverless computing) – модель облачных вычислений, в которой платформа динамично руководит выделением машинных ресурсов. Иногда бессерверные вычисления также называют «Функция как услуга» (Function as a Service – FaaS), потому что единицей кода является функция, которая выполняется платформой. Для выполнения каждого запроса создается отдельный изолированный контейнер, который уничтожается после завершения работы функции.

Основными преимуществами применения бессерверных технологий являются:

- поддержка нескольких языков программирования;
- гибкая масштабируемость – бессерверная архитектура имеет возможность масштабирования в соответствии с рабочей нагрузкой приложения;
- снижение затрат на поддержку инфраструктуры – вычислительные ресурсы оплачиваются только тогда, когда они фактически используются;
- снижение затрат на разработку – отсутствует необходимость в обслуживании инфраструктуры, так как эту работу делает провайдер услуг;
- высокая доступность – крупные провайдеры имеют ЦОДы во многих крупных регионах.

Недостатками бессерверных технологий являются:

- ограниченные ресурсы;
- холодный старт;
- высокая стоимость при большом количестве вызовов.

Большинство IoT устройств выполняют запросы либо через заданный интервал времени, либо по мере необходимости. Это указывает на две зада-

чи стоящие перед стандартной серверной инфраструктурой:

- в течение длительных периодов времени поступает мало запросов, что ведет к простоя вычислительных мощностей;
- при большом всплеске запросов, отсутствие возможности масштабироваться и параллельно обрабатывать запросы ведет к блокировкам.

Применение бессерверных технологий позволяют решить эти проблемы. На Рис 2. показано взаимодействие IoT устройств и бессерверной архитектуры.

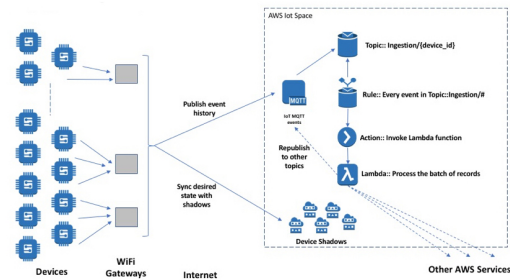


Рис. 2 – Взаимодействие IoT устройств и бессерверной архитектуры

ЗАКЛЮЧЕНИЕ

Ряд современных облачных провайдеров предоставляет набор готовых к применению сервисов и технологий таких как «цифровой двойник» и «бессервисная архитектура» для управления IoT устройствами. Использование таких технологий позволит сконцентрироваться на логике работы системы, избавит от необходимости поддержки инфраструктуры, а также снизит стоимость разработки и обслуживания.

СПИСОК ЛИТЕРАТУРЫ

1. The Growth in Connected IoT Devices Is Expected to Generate 79.4ZB of Data in 2025, According to a New IDC Forecast [Electronic resource] / IDC. – 2019. – Mode of access: <https://www.idc.com/getdoc.jsp?containerId=prUS45213219>. – Date of access: 30.09.2019.
2. Применение больших данных в электронном образовании / С. Н. Нестеренков [и др.] // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня : сборник материалов V Международной научно-практической конференции, Минск, 13–14 марта 2019 г. В 2 ч. Ч. 2 / Белорусский государственный университет информатики и радиоэлектроники; редкол. : В. А. Богуш [и др.]. – Минск, 2019. – С. 242 – 246.
3. Bolton R. N. Customer experience challenges: bringing together digital, physical and social realms / R. N. Bolton, J. R. McColl-Kennedy, L. Cheung, A. Gallan, C. Orsingher, L. Witell, M. Zaki // Journal of Service Management. – 2018. –P. 8
4. Grieves M. W. Digital Twin: Manufacturing Excellence through Virtual Factory Replication – LLC, 2014
5. Gartner Top 10 Strategic Technology Trends for 2019 [Electronic resource] / Gartner. – 2018. – Mode of access: <https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2019/>. – Date of access: 30.09.2019.

ОРКЕСТРАЦИЯ DOCKER КОНТЕЙНЕРОВ: КОНЦЕПЦИЯ ZERO DOWNTIME DEPLOYMENT

Новик И. П.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: nolik03@gmail.com

В данной статье обсуждаются проблемы микросервисной архитектуры и методы их решения путем оркестрации упакованных в Docker контейнеры микросервисов, рассматривается концепция бесшовной доставки новых версий микросервисов.

ВВЕДЕНИЕ

В настоящее время все большую популярность набирает подход разработкой приложений на основе микросервисной архитектурами. Вместе с преимуществами данного подхода возникают новые сложности в управлении инфраструктурой проектов. При развертывании новых версий приложения мы непременно сталкиваемся с проблемой простоя, образующегося во время переключения production-сервера. Данные простои очень дорого отражаются на бизнесе клиентов, зачастую мы можем видеть, как популярные сайты делают “заставку” на время технических работ по развертыванию новой версии приложения. За данными заставками могут скрываться как устаревшие техники обновления артефактов на серверах (такие как прямой передеплоймент), так и прямая доставка докер контейнеров на хост. В качестве решения проблемы предлагается применение концепции Zero Downtime Deployment которая позволяет обойти данную проблему.

I. МЕТОД СТАДАРТНОЙ ДОСТАВКИ НОВОЙ ВЕРСИИ ПРИЛОЖЕНИЯ

Рассмотрим стандартный метод доставки новой версии приложения с передеплойментом (см. рис. 1).



Рис. 1 – процесс доставки приложения с пролемой простоя

Трафик со старой версии приложения на новую не может переключаться мгновенно: предварительно мы должны убедиться, что новая версия не только успешно выкачена, но и «прогрета» (т.е. полностью готова к обслуживанию запросов).

Таким образом, некоторое время обе версии приложения (старая и новая) будут работать од-

новременно. Что автоматически приводит к конфликту общих ресурсов [1]: сети, файловой системы, IPC и т.д.



Рис. 2 – конфликты доставки приложения с проблемой простоя

II. МЕТОД ДОСТАВКИ НОВОЙ ВЕРСИИ ПРИЛОЖЕНИЯ ПО КОНЦЕПЦИИ ZERO DOWNTIME DEPLOYMENT

С использованием Docker проблема простоя решается запуском разных версий приложения в отдельных контейнерах, для которых гарантируется изоляция ресурсов в рамках одного хоста (сервера/виртуальной машины) (см. рис. 3).

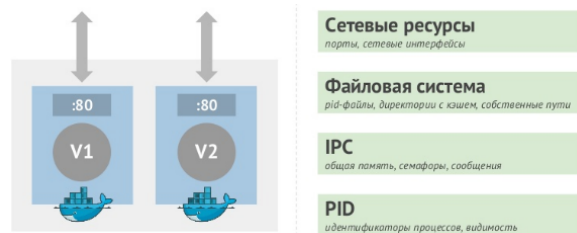


Рис. 3 – изоляция ресурсов путем запуска нескольких контейнеров на одном хосте

Контейнеризация даёт много других плюсов в процессе доставки кода приложения на конечный сервер (deploy). Любое приложение зависит от определенной версии (или диапазона версий) интерпретатора, наличия модулей/расширений и т.п., а также и их версий и относится это не только к исполняемой среде, но и ко всему окружению включая системное программное обеспечение и его версии (вплоть до используемого Linux-дистрибутива). То, что контейнеры содержат не только код приложений, но и предварительно установленное систем-

ное и прикладное программное обеспечение нужных версий, избавляет нас от проблем с зависимостями [2].



Рис. 4 – процесс доставки приложения по концепции Zero downtime deployment

Общую схему развертывания новых версий с учётом концепции Zero Downtime Deployment можно описать следующим алгоритмом:

1. старая версия приложения работает в первом контейнере;
2. новая версия разворачивается и «прогревается» во втором контейнере, (примечательно, что сама эта новая версия может нести не только обновлённый код приложения, но и любых его зависимостей, а также системных компонентов, например, новую версию OpenSSL или всего дистрибутива);
3. к тому моменту как новая версия полностью готова к обслуживанию запросов, трафик переключается с первого контейнера на второй, и старая версия может быть остановлена.

III. ОРКЕСТРАЦИЯ DOCKER КОНТЕЙНЕРОВ ПРИ РАЗВЕРТЫВАНИИ

Оркестрация - термин, который обозначает распределение контейнеров, управление кластером, и возможность добавления дополнительных хостов. При разделении приложения на отдельные изолированные сервисы, эти сервисы должны восприниматься, как единое целое. В микросервисной архитектуре развёртывание одного сервиса без развёртывания остальных не имеет

никакого смысла, поскольку они должны работать сообща.

От инструмента распределения задач, при развертывании микросервисной архитектуры, требуется группировка контейнеров. Управление группами позволяет администратору работать с набором контейнеров, как с единым целым. Запуск и выполнение тесно связанных компонентов, как единого целого, упрощает управление приложением, не жертвуя при этом преимуществами разделения функциональности между отдельными компонентами, что позволяет администраторам пользоваться преимуществами контейнеризации в случае использования микросервисной архитектуры, минимизирую при этом усилия по управлению системой [3], а также позволяет реализовывать концепцию Zero Downtime Deployment. В ходе проведения эксперимента предлагается использовать оркестратор контейнеров Kubernetes. В ходе эксперимента реализуется концепция развертывания на оркестраторе и применяется стратегия обновления RollingUpdate, при которой контейнеры будут обновляться плавно, по очереди.

IV. ЗАКЛЮЧЕНИЕ

В работе рассмотрены методы доставки новых версий приложения. Предложен алгоритм развертывания с учетом концепции Zero Downtime Deployment. Оркестратором для развертывания с учетом концепции выбрана платформа Kubernetes с применением стратегии обновления RollingUpdate.

СПИСОК ЛИТЕРАТУРЫ

1. Vikram, M. *Microservices Deployment Cookbook* / M. Vikram // PacktPublishing, 2017. – P. 129.
2. Fisher, L. M. *The Art of Scalability: Scalable Web Architecture, Processes, and Organizations for the Modern Enterprise (2nd Edition)* / M. Vikram // Addison-Wesley Professional, 2015. – P. 229.
3. Randall, S. *Docker Orchestration* / S. Randall // PacktPublishing, 2017. – P. 84.

ПОДХОДЫ К РЕШЕНИЮ ЗАДАЧИ АВТОМАТИЗИРОВАННОГО СОСТАВЛЕНИЯ ЭКСКУРСИОННЫХ ТУРОВ

Парамонов А. И., Труханович И. А.

Кафедра программного обеспечения информационных технологий, факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: anton_paramonov@tut.by, ilya.trukhanovich@gmail.com

В работе рассмотрена проблема автоматизированного построения туристических маршрутов. В контексте этого вопроса предложен подход к решению задачи о составлении экскурсионных туров. Задача компоновки туров сводится к задачам о рюкзаке и методам генетических алгоритмов. Показаны результаты эксперимента о применимости различных подходов.

ВВЕДЕНИЕ

Сегодня по многим социальным, экономическим, технологическим и геополитическим причинам туризм является одной из важнейших отраслей промышленности во всем мире. Он предлагает действительно высокую динамику и потенциал для количественного и качественного роста со значительным вкладом в мировую экономику и занятость, поскольку является крупнейшим в мире поставщиком рабочих мест.

В секторе туризма, как и в других отраслях, внедрение новых технологий формирует новые тенденции и взгляды на реализацию бизнес-процессов. Туристам сегодня важно не только иметь постоянный доступ к информации, но и получать сервисы для ее обработки при решении вопросов планирования туристических маршрутов. Основные запросы связаны с возможностью построения альтернатив туристических маршрутов. В настоящий момент такие сервисы уже представлены, но тем не менее в них все ещё сохраняется достаточно много аспектов, в рамках которых действия осуществляется пользователем (вручную) [1].

1. ПРОБЛЕМА АВТОМАТИЗАЦИИ СОСТАВЛЕНИЯ ТУРИСТИЧЕСКОГО МАРШРУТА

Прежде всего при построении конкретного туристического маршрута следует определиться с конечным набором туристических объектов для посещения. Для этого стоит учесть соответствие предпочтений пользователей с атрибутами этих объектов. При этом следует учитывать информацию о совокупном текущем рейтинге последних.

К тому же на выбор объектов для тура влияют дополнительные ограничения, которые устанавливаются клиентом. Прежде всего, у каждой туристической поездки есть свой бюджет, в который следует уложиться, а значит, суммарная стоимость посещения всех объектов не может его превысить. Как следствие, при выборе набора наиболее подходящих объектов для путешествия следует, прежде всего, обратить внимание

именно на этот ограничивающий фактор. В конечном итоге, когда необходимые объекты выбраны с учётом всех ограничений, нужно проинформировать построение самого маршрута, который должен быть оптимальным (кратчайшим). Для этого необходимо обладать информацией о взаимном расположении всех выбранных объектов.

Рассмотрим предварительный вариант алгоритма построения туристического маршрута. Он будет состоять из трёх основных этапов. Сначала необходим набор всех возможных подходящих туристических объектов. Для каждого из них должен быть известен текущий рейтинг и стоимость посещения. Кроме того, должны быть известны взаимные расстояния между объектами. Эта информация может быть задана в виде матрицы расстояний. На втором этапе осуществляется формирование групп объектов из множества, которое получено в результате выполнения первого этапа. Предлагается для решения данной задачи рассматривать методы решения задачи о рюкзаке [2]. В данном случае в качестве «стоимости» будет выступать рейтинг объекта, а в качестве «веса» – стоимость его посещения. Таким образом, мы максимизируем суммарный рейтинг при заданном ограничении суммарной стоимости (бюджета). Третий этап предполагает построение оптимального маршрута (или списка маршрутов) на основе сформированных на втором этапе групп туристических объектов. Для этого предлагается применить известные алгоритмы решения задачи коммивояжёра [3]. Благодаря чему оставшиеся объекты можно будет связать в цепь для посещения.

После всех этапов мы получим оптимальный маршрут для посещения туристических объектов, обладающий максимальным суммарным рейтингом при заданных ограничениях бюджета.

В конечном итоге процесс построения маршрута состоит из трёх этапов:

1. Выбор набора объектов (с заданными рейтингами и матрицей расстояний);

2. Решение задачи о рюкзаке для списка объектов;
3. Решение задачи коммивояжёра для набора объектов из предыдущего шага.

II. ПРИМЕНЕНИЕ МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ О РЮКЗАКЕ К ЗАДАЧЕ ФОРМИРОВАНИЯ ЭКСКУРСИОННЫХ ТУРОВ

Рассмотрим второй этап предложенного алгоритма, который заключается в решении задачи о рюкзаке при построении списка туристических объектов. Рассмотрим несколько методов решения данной задачи.

Задачу о рюкзаке можно решать с помощью динамического программирования с использованием соответствующих рекуррентных соотношений [4]. В соответствии со сказанным выше, в рамках решения максимизировать суммарный рейтинг объектов при условии превышения бюджета суммарной стоимостью.

Также задачу о рюкзаке можно решать с помощью генетического алгоритма [5]. В данном случае генотип будет представлять собой подмножество объектов списка, которые рассматриваются для включения в тур. Функцией приспособленности в данном случае будет являться суммарный рейтинг объектов при условии, что суммарная стоимость не превышает бюджет.

С помощью тестовых наборов произведём сравнение эффективности алгоритмов. Каждый тестовый набор представляет собой совокупность объектов, прошедших первый этап (выбор). Для каждого тестового набора определена максимальная стоимость (бюджет), а для каждого объекта в наборе заданы рейтинг и стоимость.

Запуск алгоритма для каждого тестового набора даёт результат – список туристических объектов (с суммарным рейтингом). Кроме того, имеет значения и время выполнения алгоритма. Эти данные (суммарный рейтинг и время выполнения) и будут отражены в результатах тестирования.

Рассмотрим результаты проверки для 5 тестовых наборов.

В табл. 1 приведены результаты решения задачи о рюкзаке методом динамического программирования.

Таблица 1 – Задача о рюкзаке (динамическое программирование)

Номер тестового набора	Количество объектов в наборе	Суммарный рейтинг	Время выполнения алгоритма (мс)
1	5	52	0.06
2	10	76	0.1
3	15	87	0.17
4	20	125	0.215
5	30	182	0.32

В табл. 2 приведены результаты решения задачи о рюкзаке генетическим алгоритмом.

Таблица 2 – Задача о рюкзаке (генетический алгоритм)

Номер тестового набора	Количество объектов в наборе	Суммарный рейтинг	Время выполнения алгоритма (мс)
1	5	52	0.007
2	10	76	0.011
3	15	87	0.016
4	20	110	0.0224
5	30	151	0.039

Для этих (и других тестовых наборов) наблюдается ярко выраженный выигрыш генетического алгоритма по времени при его проигрыше в точности.

ЗАКЛЮЧЕНИЕ

В ходе исследования были рассмотрены вопросы построения списка объектов в рамках задачи автоматизации построения туристического маршрута. В процессе решения задачи о рюкзаке при построении данного списка было выявлено преимущество генетического алгоритма по времени и метода динамического программирования в точности.

Поскольку построение данного списка ведётся в рамках автоматизации составления туристического маршрута, то предпочтение, как правило, стоит отдавать методу динамического программирования, поскольку точность в данном случае является более важным аспектом. Тем не менее, при предполагаемых соответствующих настройках сервиса, снижающих требования к точности результата, для построения списка туристических объектов может производиться обращение и к генетическому алгоритму.

СПИСОК ЛИТЕРАТУРЫ

1. Eno SigTur Mobile [Electronic resource] // PCT. – Mode of access: http://www.pct-turisme.cat/eng/projects_enosigturmob.html. – Date of access: 25.10.2019.
2. Knapsack Problem – an overview [Electronic resource] // ScienceDirect. – Mode of access: <https://www.sciencedirect.com/topics/computer-science/knapsack-problem>. – Date of access: 28.10.2019.
3. Романовский, И. В. Алгоритмы решения экстремальных задач / И. В. Романовский. – Москва : Наука, 2006. – 352 с.
4. Левитин, А. В. Алгоритмы: введение в разработку и анализ / А. В. Левитин. – Москва : Вильямс, 2006. – 576 с.
5. Скобцов, Ю. А. Основы эволюционных вычислений / Ю. А. Скобцов. – Донецк : ДонНТУ, 2008. – 326 с.

МЕТОДИКА ПРОГРАММНОЙ ИДЕНТИФИКАЦИИ ПАРАМЕТРОВ НАБЛЮДАЕМОГО ОБЪЕКТА

Пархоменко И. Н., Господ А. В.

Кафедра автоматизации технологических процессов и производств

Могилевский государственный университет продовольствия

Могилев, Республика Беларусь

E-mail: gospod_andrei@mail.ru

Предложена новая методика программного определения площади исследуемого объекта, с помощью одной камеры, используя трансформацию цвета (функция I), пороговое преобразование (функция II) и определение контуров (функция III).

Необходимость отказа от ручного труда формируется современной необходимостью и одним из компонентов автоматизации является машинное зрение, позволяющее идентифицировать объекты [1]. Применение технического зрения в различных областях: в медицине - для диагностики заболеваний, в промышленности - автоматизация дает уменьшение себестоимости, в автомобилестроении - навигация беспилотников [2]. И одним из параметров наблюдаемых объектов, который необходимо идентифицировать является его площадь [3], для чего разработана методика её определения с помощью веб камеры и трех функций.

Для преобразования входного изображения из одного цветового пространства в другое используется функция I. Которая на входе получает 8-битное изображение без знака, 16-битное без знака или с плавающей точкой одинарной точности, код преобразования цветового пространства и количество каналов в целевом изображении, в результате имеем изображение в серых тонах, которое передается в функцию II.

Функция II применяет пороговое преобразование, для каждого пикселя применяется одинаковое пороговое значение. Если значение пикселя меньше порогового значения, оно устанавливается равным 0, в противном случае устанавливается максимальное значение. Первым аргументом является исходное изображение, которое должно быть изображением в градациях серого. Вторым аргумент - это пороговое значение, которое используется для классификации значений пикселей. Третьим аргумент - это максимальное значение, которое присваивается значениям пикселей, превышающим пороговое значение. На выходе имеем два параметра, первый - это порог, который был использован, а второй - пороговое изображение (см. рис. 1). Таким образом каждый пиксель, обнаруженный на изображении, превышающий пороговое значение содержания цвета соответствует единице в массиве под тем же порядковым номером, что и пиксель на изображении. Это необходимо для того, чтобы создать «очищенное» от лишних цветов 8-битное изображение, на котором будет находиться черный

фон и сам объект, который нам нужен и будет представлен собственным контуром в белом заполнении.

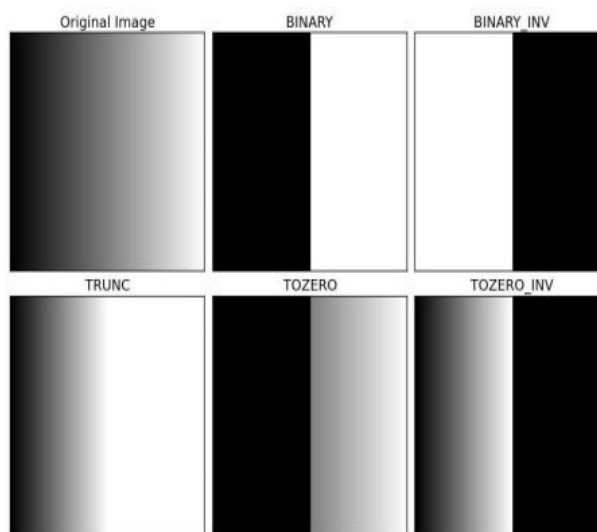


Рис. 1 – Пороговое преобразование

Но в глобальном пороговом значении использованы произвольно выбранное значение в качестве порога. Поэтому чтобы избежать необходимости выбирать значение, а определять его автоматически, рассмотрим изображение только с двумя различными значениями изображения (бимодальное изображение), где гистограмма будет состоять только из двух пиков. Хороший порог будет в середине этих двух значений. Точно так же определяется оптимальное глобальное пороговое значение из гистограммы изображения. Пороговое значение может быть выбрано произвольно. Затем алгоритм находит оптимальное пороговое значение, которое возвращается в качестве первого выхода. Входное изображение имеет шумы (см. рис.2). В первом случае применяется глобальное пороговое значение со значением 127. Во втором случае, пороговое значение применяется напрямую. В третьем случае изображение сначала фильтруется гауссовым ядром 5x5 для удаления шума, затем применяется пороговое значение. Поскольку идет работа с бимодальными изображениями, алгоритм пытается найти пороговое значение (t), которое мини-

мизирует взвешенную дисперсию внутри класса, определяемую соотношением:

$$\sigma_{\omega}^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

тогда

$$q_1(t) = \sum_{i=1}^t P(i) \ \& \ q_2(t) = \sum_{i=t+1}^I P(i)$$

$$\mu_1(t) = \sum_{i=1}^t \frac{iP(i)}{q_1(t)} \ \& \ \mu_2(t) = \sum_{i=t+1}^I \frac{iP(i)}{q_2(t)}$$

$$\sigma_1^2(t) = \sum_{i=1}^t [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)} \ \& \ \sigma_2^2(t) = \sum_{i=t+1}^I [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)}$$

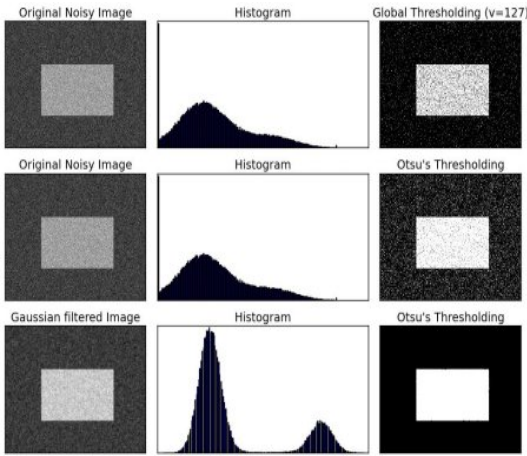


Рис. 2 – Фильтрация шума

Для определения контура не черного объекта используется функция III, которая возвращает массив векторов точек, составляющий собой объект с изображения. На входе функция III получает 8-битное одноканальное изображение, где ненулевые пиксели рассматриваются как 1, нулевые пиксели остаются 0, поэтому изображение рассматривается как двоичное. На выходе - обнаруженные контуры, которые сохраняются как векторы точек. Также получим выходной вектор, содержащий информацию о топологии изображения. Он имеет столько же элементов, сколько число контуров. Для каждого контура i -го контура $[i]$ иерархия элементов $[i] [0]$, $hiearchy [i] [1]$, $hiearchy [i] [2]$ и $hiearchy [i] [3]$ установлены в 0 - основные индексы в контурах следующего и предыдущего контуров на том же иерархическом уровне, первый дочерний контур и родительский контур, соответственно. Если для контура i нет следующих, предыдущих, родительских или вложенных контуров, соответствующие элементы иерархии $[i]$ будут отрицательными. Изображение рассматривается как массив

$$z(z_1, \dots, z_n, \dots, z_N)$$

где z – значения интенсивности пикселей, n – общее число пикселей. Для отделения объекта от

фона алгоритм определяет значения элементов массива прозрачности

$$a(a_1, \dots, a_n, \dots, a_N)$$

причем a_n может принимать два значения, если $a_n = 0$, значит пиксель принадлежит фону, если $a_n = 1$ то объекту. Внутренний параметр Θ содержит гистограмму распределения интенсивности переднего плана и гистограмму фона:

$$\Theta = \{h(z; a), a = 0, 1\}$$

Задача сегментации - нахождение неизвестных a_n . Рассматривается функция энергии:

$$E(a, \Theta, z) = U(a, \Theta, z) + V(a, z)$$

Причем минимум энергии соответствует наилучшей сегментации:

$$U(a, \Theta, z) = - \sum_n \log h(z_n, a_n)$$

$$V(a, z) = \sum_{(m,n)} \frac{1}{dis(m,n)} [a_n \neq a_m] \exp(-\beta(z_m - z_n)^2)$$

где $V(a, z)$ – слагаемое отвечает за связь между пикселями, сумма идет по всем парам пикселей, которые являются соседями, $dis(m, n)$ – евклидово расстояние, $a_n \neq a_m$ отвечает за участие пар пикселей в сумме, если $a_n = a_m$, то эта пара не будет учитываться, $U(a, \Theta, z)$ - отвечает за качество сегментации, т.е. разделение объекта от фона. Найдя глобальный минимум функции энергии E , получим массив прозрачности:

$$\hat{a} = argmin_a E(a, \Theta)$$

Для минимизации функции энергии, изображение описывается как граф и ищется минимальный разрез графа.

Таким образом разработано программное обеспечение определяющее площадь наблюдаемого объекта, что в перспективе позволит также находить расстояние до наблюдаемого объекта с использованием всего лишь одной камеры, т.е. без использования стерео пары.

СПИСОК ЛИТЕРАТУРЫ

1. Escriva, D. M., Joshi, P., Mendonca, V., Shilkrot, R. Building Computer Vision Projects with OpenCV 4 and C++ / D. M. Escriva, P. Joshi; V. Mendonca; R. Shilkrot // Packt Publishing. – Birmingham. – 2019. – p. 538.
2. Kaehler, A., Bradski, G. Learning opencv / A. Kaehler, G. Bradsk // O'Reilly Media. – Gravenstein Highway North. – 2017. – p. 1024.
3. Datta, S. Learning OpenCV 3 Application Development / S. Datta // Packt Publishing. – Birmingham. – 2016. – p. 310.

МОДИФИКАЦИЯ МЕТОДА РАЗРЕШЕНИЯ ЛЕКСИЧЕСКОЙ МНОГОЗНАЧНОСТИ В ОБЛАСТИ БИМЕДИЦИНЫ

Пашук А. В., Гуринович А. Б., Кузнецов А. П., Смирнов В. Л.

Кафедра информатики, кафедра вычислительных методов и программирования, кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: pashuk@bsuir.by, gurinovich@bsuir.by, kuznap@bsuir.by

Разрешение лексической многозначности (Word Sense Disambiguation) является промежуточной задачей в процессе поиска и извлечения информации, представляющей собой проблему выбора правильного смысла неоднозначных слов. В данном исследовании предлагается модификация предложенного ранее алгоритма [3] с целью увеличения точности определения верного значения целевого термина.

ВВЕДЕНИЕ

Количество научных статей в области биомедицины стремительно увеличивается (рисунок 1), а значит, что все большую актуальность приобретают вопросы качества поиска и структурирования информации. Увеличение объемов информации также замедляет получение новых знаний.

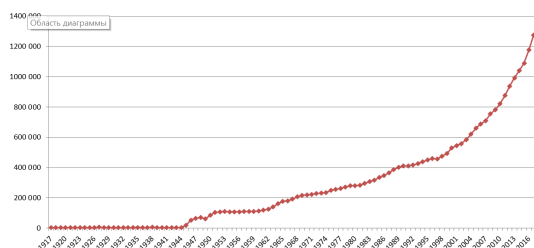


Рис. 1 – Количество публикаций, опубликованных на MEDLINE/PubMed по состоянию на 2018 год [1]

Качество поиска напрямую зависит от степени понимания машиной обрабатываемых научных текстов. Одной из проблем понимания неструктурированной текстовой информации, является проблема разрешения лексической многозначности (Word Sense Disambiguation, WSD). В общем смысле WSD является проблемой классификации – это процесс отнесения слова или фразы к определенному значению, которое отличается от других значений, которые потенциально может принимать это же слово (фраза). Обычно используется одно из двух предположений: слово может принимать только одно значение в рамках рассматриваемого документа, либо слово принимает определенное значение в рамках контекста – соседних слов, предложений.

Проблему разрешение лексической многозначности можно сформулировать следующим образом: дан документ A , содержащий термин t_k из словаря T . Любой термин t_k может быть отнесен с минимум одним из значений S_{ki} из словаря S_k . Задача состоит в том, чтобы определить наиболее вероятное значение s_{k*} для термина t_k , используемого в документе A .

Подробный обзор подходов к разрешению лексической многозначности приведен в обзорной статье [2], в которой автор предлагает использовать три основные категории алгоритмов: основанные на обучении с учителем, основанные на знаниях и основанные на обучении без учителя. Первая категория подходов использует размеченный набор данных с дополнительной информацией, полученной из контекста вокруг термина, для построения моделей машинного обучения, которые предсказывают правильный смысл для заданного контекста.

Подходы, основанные на знаниях [5], [6], не используют какой-либо размеченный корпус данных, а опираются исключительно на словари, такие как Unified Medical Language System (UMLS), которые содержат краткие определения различных смыслов термина и соответствующих синонимов. В подходах без учителя используются методы, основанные на тематическом моделировании [7], для устранения неоднозначности. Системы, использующие такие методы иногда называют Word Sense Discrimination, т.к. происходит группировка нескольких употреблений заданного термина в кластеры, где каждому кластеру соответствует определенное значение целевого слова. Т.е. в этом случае не происходит определение значения и списка предопределенных значений, а только группировка в кластеры.

На рисунке 2 приведена схема модуля, реализующего разработанный алгоритм.

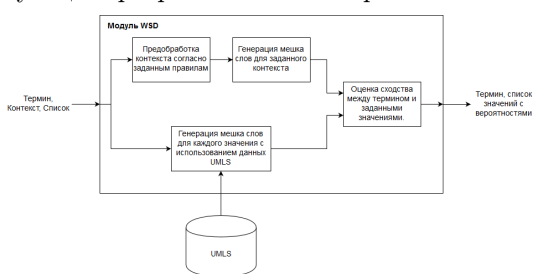


Рис. 2 – Архитектура модуля разрешения лексической многозначности [1]

В [3] были приведены основные сведения о используемом алгоритме разрешения лексиче-

ской многозначности. В данном исследовании будут предложены некоторые улучшения данного алгоритма, затрагивающие блок "Предобработка контекста согласно заданным правилам".

I. ДОПОЛНИТЕЛЬНАЯ ОБРАБОТКА КОНТЕКСТА

Под дополнительной обработкой текста понимается лемматизация и стемминг слов, входящих в контекст термина, для которого необходимо определить смысл. В текстах используются разные грамматические формы одного и того же слова, а также могут встречаться однокоренные слова. Процедуры лемматизации и стемминга преследуют цель привести все встречающиеся словоформы к одной словарной форме (например, lungs к lung - легкие к легкое), что позволяет увеличить точность сравнения bags-of-words для определения и заданного контекста.

Дополнительной операцией, позволяющей увеличить качество работы алгоритмов сравнения текстов, является удаление стоп-слов (например, предлоги и местоимения).

Стоит отметить, что применительно к обработке биомедицинских текстов следует соблюдать осторожность при применении описанных методик, т.к. существует вероятность потери полезной информации. Так, существующие лемматизаторы могут некорректно преобразовывать специфичные биомедицинские термины, что может привести к ошибкам в работе основного алгоритма. Частично эта проблема решается дополнительными фильтрами, учитывающими регистр написания слов и другие параметры.

В рамках исследования была использована библиотека Natural Language Toolkit (NLTK) [9].

II. ИСПОЛЬЗОВАНИЕ ГРАФА ЗНАНИЙ

Основным источником данных о биомедицинских терминах с используемым алгоритме является словарь UMLS, имеющий графовую структуру. Для улучшения качества работы разрабатываемого алгоритма можно использовать не только информацию о терминах из контекста (например, их определения для построения bags-of-words), а также информацию о взаимосвязях между ними. Таким образом, есть возможность построить граф для заданного контекста и всех входящих в него терминов и оценить вероятность правильности каждого смысла (значения) для разрешения лексической многозначности.

На рисунке 3 приведен пример связей из UMLS для двух значений термина Cold.

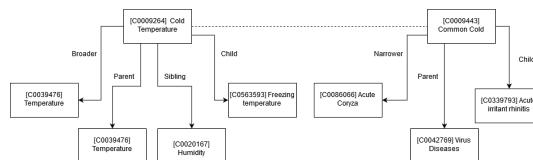


Рис. 3 – Пример графа, сформированного для двух значений термина Cold

В рамках исследования был использован алгоритм PageRank, позволяющий ранжировать все возможные значения многозначных слов на основе их структурной значимости в графе и их связи со словами в заданном контексте.

ЗАКЛЮЧЕНИЕ

Экспериментальная проверка разработанного алгоритма разрешение лексической многозначности с использованием описанных модификаций показала, что метод дает точность до 87.33

Из возможных улучшений алгоритма можно выделить возможность использования алгоритмов Word Embeddings перед составлением словаря bag-of-words. Также дополнительная информация о возможных улучшениях алгоритма может быть получена после анализа ошибок первого и второго рода (false positives и false negatives соответственно).

СПИСОК ЛИТЕРАТУРЫ

1. Statistical Reports on MEDLINE®/PubMed® Baseline Data / Mode of access: <https://www.nlm.nih.gov/bsd/licensee/baselinestats.html>. – Date of access: 11.09.2019.
2. Navigli, R. Word sense disambiguation: a survey / R. Navigli // ACM Comput. Surv. (CSUR). – 2009. – Vol. 41 (2). – P. 10.
3. Пашук А. В. Анализ методов разрешения лексической многозначности в области биомедицины / А. В. Пашук, А. Б. Гуринович, Н. А. Волорова, А. П. Кузнецов // Доклады БГУИР. – 2019. – №5. – С. 60-65.
4. Disambiguation of Biomedical Text / Mode of access: <https://www.slideserve.com/ownah/disambiguation-of-biomedical-text>. – Date of access: 10.09.2019.
5. Knowledge-based biomedical word sense disambiguation: comparison of approaches / Mode of access: <http://paperity.org/p/56856103/knowledge-based-biomedical-word-sense-disambiguation-comparison-of-approaches>. – Date of access: 10.09.2019.
6. DALE: A Word Sense Disambiguation System for Biomedical Documents Trained using Automatically Labeled Examples / Mode of access: <https://www.aclweb.org/anthology/N13-3001>. – Date of access: 08.09.2019.
7. The effect of word sense disambiguation accuracy on literature based discovery. - BMC Medical Informatics and Decision Making. Mode of access: <https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/s12911-016-0296-1>. – Date of access: 08.09.2019.
8. Word Sense Disambiguation (WSD). Test Collection. Collaborations & Outside Resources / Mode of access: <https://wsd.nlm.nih.gov/collaboration.shtml>. – Date of access: 12.09.2019.
9. Natural Language Toolkit / Mode of access: <https://www.nltk.org/>. – Date of access: 10.09.2019.

ПОВЫШЕНИЕ НАДЕЖНОСТИ ЭЛЕКТРОСНАБЖЕНИЯ ПОТРЕБИТЕЛЕЙ РЕСПУБЛИКИ КАЗАХСТАН ПУТЁМ ВНЕДРЕНИЯ ЦИФРОВЫХ ТЕХНОЛОГИЙ

Рахимов Р. В.

Кафедра системного анализа и управления, Евразийский национальный университет имени Л.Н. Гумилева
Нур-Султан, Республика Казахстан
E-mail: peresilochki@mail.ru

В данной статье автором рассматривается вопрос повышения надежности электроснабжения потребителей за счет внедрения цифровых технологий, на примере опыта реализации цифровой подстанции в энергетике. Изучены комплексные системы энергосбережения мировых производителей Omron, Siemens, Schneider Electric, ABB и др.

Потребности населения и промышленности в электроэнергии растут с каждым годом, за последние десять лет нагрузка на электрические сети возросла в 1,7 раза, однако существующие мощности уже не способны сравниться с растущим спросом на электроэнергию.

Надежное энергоснабжение является важнейшей составляющей для объектов инфраструктуры городов, промышленных предприятий и зданий. Надежность, эффективность, безопасность – таковы основные требования, которые предъявляются к системам распределения электроэнергии.

В числе главных инноваций, которые способны обеспечить качественное функционирование электроэнергетической инфраструктуры, все чаще обсуждаются вопросы использования цифровых технологий при добыче, в генерации и передаче электрической энергии. Ключевым звеном энергетической системы является электрическая подстанция, предназначенная для приема, преобразования и распределения электрической энергии до потребителя, в устройства и элементы которой возможно и необходимо применение цифровых решений.

Эксплуатация традиционных распределительных подстанций выявила ряд их недостатков:

- Низкий уровень безопасности для персонала;
- Сложная конфигурация сетей;
- Минимальные показатели надежности и доступности;
- Высокие метрологические потери во вторичных цепях;
- Отсутствие возможности работы и прослеживания за объектами в реальном времени;
- Высокие эксплуатационные затраты [1];
- Отсутствие интеллектуальных систем связи и интеграции с сетями Smart Grid.

В последние годы, все чаще используется термин «цифровая подстанция», который означает: высоко автоматизированную подстанцию, в которой практически все процессы обмена ин-

формацией и управления осуществляются в цифровом виде на всех уровнях управления. Цифровая подстанция обеспечивает эффективное использование информации о процессах на подстанции, повышение согласованности действия различных видов оборудования. Идея цифровой подстанции заключается в создании единого комплекса управления, состоящего из трех автономных частей в основе каждого из которых есть своя отдельная модель электроэнергетической системы, а именно:

1. Оперативно-диспетчерское управление. Управляющие воздействия реализуются, в основном, оперативно-диспетчерским персоналом с использованием вспомогательных устройств автоматики;
2. Противоаварийное управление. Управление при сильных возмущениях в условиях электромеханических переходных процессов (например, внезапное отключение линии, генератора и т.д.);
3. Релейная защита. Локальное управление электроэнергетической системой путём быстрого выявления и отделения поврежденных элементов от исправной части электроэнергетической системы.
4. Системы интеллектуального учёта электроэнергии. Организация учета, обеспечивающая экономически эффективную, устойчивую систему электроснабжения с низкими потерями, высоким уровнем качества и безопасности и эффективно интегрирующая действия всех пользователей [2].

Основные преимущества цифровых подстанций:

- Унификация информационных протоколов обмена данными;
- Сокращение кабельного хозяйства;
- Обеспечение наблюдаемости каналов сбора, передачи информации и управления;
- Снижение метрологических потерь во вторичных цепях;
- Упрощение способов тиражирования первичной информации;
- Упрощение механизмов поверки устройств;

- Унификация механизмов конфигурирования подстанции;
- Формирование единой системы диагностики устройств вторичной коммутации;
- Переход к выполнению удаленной функциональной диагностики;
- Обеспечение информационной безопасности энергообъекта;
- Переход к необслуживаемым подстанциям [1].

Эти три части управляющего комплекса построены на основе принципиально разных моделей электроэнергетических систем, имеют разные динамические характеристики и, поэтому, реализуются в виде отдельных управляющих систем. Однако, все передающие устройства сети должны соответствовать электротехническим стандартам IEC 61850 и IEEE 1613, быть устойчивыми к неблагоприятным условиям внешней среды.

Комплексными решениями по эффективному обслуживанию и управлению распределительными подстанциями существуют у компании Omron, Siemens, Schneider Electric, АВВ и др.

Внедрение компонентов автоматизации позволяет повысить надежность и эффективность работы оборудования, а также уменьшить возможность ошибочных действий персонала, способом разработки взаимосвязанных подсистем, обеспечивающих централизованное автоматизированное управление технологическим процессом подстанции. В структуре построения автоматизированной системы управления технологическим процессом закладывается принцип многоуровневой иерархической распределенной системы. Ключевые технологии для применения в энергетике представлены широким спектром аппаратно-программного комплекса, имеющего высокую конкурентоспособность на мировом рынке.

Комплексная система электроснабжения (TIP – Totally Integrated Power) содержит обширный ассортимент продукции, систем и решений для сетей высокого, низкого и среднего напряжения. В промышленных приложениях компании оказывают поддержку в форме комплексной автоматизации (TIA), а также предлагают комплексные решения для зданий (TBS) и решения по реализации концепции построения интеллектуальных сетей.

Цифровые подстанции позволяют осуществлять контроль силового оборудования в реальном времени с применением стандарта IEC 61850, имеют современный интерфейс диспетчера для мониторинга первичного оборудования, автоматизации и управления данными, в безопасных режимах работы, снижая уровень опе-

рационных затрат и занимая меньше пространства на размещение.

Исследования технико-экономических показателей позволяют сделать вывод, что стоимость нового решения при переходе на серийный выпуск продукции не будет превышать стоимости традиционных решений построения систем автоматизации, а по некоторым показателям, таким как проектирование цифровых подстанций, монтаж, поставка программно-технического комплекса и пусконаладочные работы, будет значительно дешевле [2].

Таким образом, применение цифровых технологий, а именно, строительство цифровых подстанций позволит обеспечить требуемый уровень надежности системы электроснабжения потребителей при минимальных затратах.

В электроэнергетическом комплексе страны в настоящее время наблюдается полномасштабное развитие цифровых устройств, которые обладают эффективностью, надежностью, безопасностью и позволяют создать цифровую подстанцию. Применение технологии цифровой подстанции должно позволить в будущем существенно сократить расходы на проектирование, пусконаладочные работы, эксплуатацию и обслуживание энергетических объектов, в первую очередь за счет возможности посредством «цифры» значительно повысить уровень автоматизации.

Переход от традиционных распределительных подстанций к цифровым позволит сократить капитальные и эксплуатационные затраты, что в рамках целого государства даст значительный экономический эффект. Примером успешно реализованных проектов можно считать отдельные регионы России: Калининград, Уфа, Белгород, тем самым перенимая опыт российских коллег необходимо продолжить работу в данном направлении.

СПИСОК ЛИТЕРАТУРЫ

1. Рахимов Р. В. К вопросу применения компенсирующих устройств с целью повышения энергоэффективности линий 10-0,4 кВ // Сборник материалов XIV -Международной конференции студентов и молодых ученых. – 2019. – С. 5827–5830.
2. Рахимов Р. В. Перспективы применения интеллектуальных систем учета с целью повышения энергоэффективности в Республике Казахстан // Вестник Казахстанско-Британского университета. 2019. – Том 16. – Выпуск №3. – С. 103–108.
3. Фардиев И. Ш., Сафиуллин Д. Х., Забелкин Б. А., Васильев Ю. А. Меер В. М. Об инновационном проекте "Умная сеть" // Энергетика Татарстана – 2010. – № 3. – С. 14–15.
4. Ануфриев А. О. Реализация концепции Цифровая подстанция на ПС-110 «Приречная» // Новости электротехники – 2019. – № 1 (115). – 21 с.
5. Курьянов В. Н., Куш Л. Р., Горбунова Н. Р., Бондарев И. В., Цыпик В. В. Цифровые подстанции. Опыт реализации // Наука, образование и культура – 2018. – № 3 (27) – С. 9–22.

АВТОМАТИЗАЦИЯ ПРОЦЕССА ЗАПУСКА АВТОТЕСТОВ С ПОМОЩЬЮ ГРИД СИСТЕМЫ

Рубель П. А.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: polina-rubel@mail.com

В работе рассматривается одно из наиболее распространенных и эффективных средств исследования сложных систем и процессов - грид система. Также принципы работы grid системы путем внедрения ее в процесс автоматизированного тестирования на примере Selenium Grid.

ВВЕДЕНИЕ

В различных областях деятельности для обработки информации существует множество вычислительных систем различного масштаба. Одной из наиболее активно развиваемых технологий создания распределенных систем является Grid-технология которая оказалась востребована для решения как научных, так и коммерческих задач. Grid-система представляет собой набор взаимодействующих между собой компонентов (вычислительных узлов и хранилищ данных), которые могут функционировать параллельно и в работе обеспечена синхронизация. Одной из областей, в которой применяют принцип работы грид - является тестирование веб-страниц.

Таким образом, целью данной работы является исследование грид системы в сфере автоматизированного тестирования.

I. СТРУКТУРА GRID-СИСТЕМЫ

Grid создается на базе вычислительных кластеров. Кластер представляет собой совокупность компьютеров, объединенных локальной сетью и предназначенных для решения ресурсоемких (по процессорному времени, оперативной памяти и памяти на жестких дисках) вычислительных задач, и относится к классу многопроцессорных вычислительных систем. Схематично кластерная система изображена на Рис 1.



Рис. 1 – Схема кластерной системы

На вход системы поступают прикладные задачи обработки, которые могут иметь существенно различающиеся ресурсные требования. Непосредственной обработкой данных занимаются распределённые узлы кластера. Планировщик задач с некоторой периодичностью отправ-

ляет запросы на получение информации, которая требуется для выполнения итерации планирования, внутренний компонент системы управления обрабатывает принятые запросы и формирует ответные сообщения, передаваемые планировщику. Планировщик на основании полученной информации определяет узлы для запуска заданий и оповещает систему управления о необходимости активации процесса задачи, отправляя соответствующее сообщение. Таким образом, разработчики системы управления должны предоставлять следующую функциональность: формирование списка пользовательских задач, формирование списка разрешенных вычислительных ресурсов, а также предоставление информации о загрузке узлов, объеме оперативной памяти, размере свободного дискового пространства и проч.

II. СХЕМА SELENIUM GRID

Selenium Grid – это кластер, состоящий из нескольких Selenium-серверов. Он предназначен для организации распределённой сети, позволяющей параллельно запускать много браузеров на большом количестве машин.

Selenium Grid имеет топологию «звезда», то есть в его составе имеется выделенный сервер, который носит название «хаб» или «коммутатор», а остальные сервера называются «ноды» или «узлы». «Хаб» является центральным компонентом архитектуры Selenium Grid. Он загружает тесты, которые должны быть выполнены. Как только «хаб» получает входной / тестовый пример, который необходимо выполнить, он ищет узел (клиент), который соответствует требуемым возможностям, и перенаправляет «запрос выполнения теста» на соответствующий узел. Узел - это машина, на которой выполняются тесты. Узел может иметь конфигурацию, отличную от «хаб». Нет ограничений на количество узлов, которые могут быть подключены к концентратору. Рекомендуется выбирать машины с наилучшей возможной конфигурацией, чтобы вы могли получить максимальную производительность от вашей установки Selenium Grid. Схема selenium grid представлена на рисунке 1.

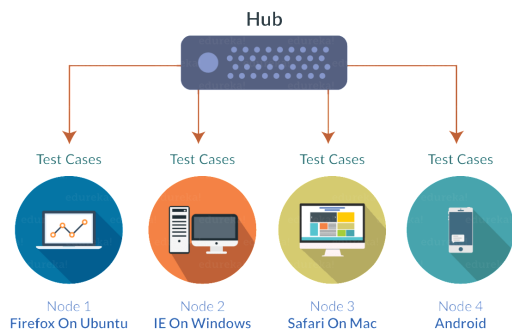


Рис. 2 – Схема Selenium grid

Сеть может быть гетерогенной, то есть коммутатор и узлы могут работать под управлением разных операционных систем, на них могут быть установлены разные браузеры. Одна из задач Selenium Grid заключается в том, чтобы «подбирать» подходящий узел, когда во время старта браузера указываются требования к нему – тип браузера, версия, операционная система, архитектура процессора и ряд других атрибутов.

Ранее Selenium Grid был самостоятельным продуктом. Сейчас физически продукт один – Selenium Server, но у него есть несколько режимов запуска: он может работать как самостоятельный сервер, как коммутатор кластера, либо как узел кластера, это определяется параметрами запуска.

III. ПРИНЦИП РАБОТЫ SELENIUM GRID

Настройка Selenium Grid может иметь только один «хаб» и число узлов n . Основная задача «хаба» заключается в распространении теста кейса, предоставленного «узлу», который соответствует возможностям / требованиям, необходимым для выполнения кросс-браузерного тестирования. В данном исследовании оптимизируем процесс запуска автотестов. Имеется рабочая станция, в которой можно запускать одновременно только две виртуальные машины. При запуске автотестов в один поток (запуская каждую ВМ вручную), общее время составит 18,54 минут. Для оптимизации используем Selenium grid. «Хаб» - центральный компонент архитектуры Selenium Grid. Основная задача «хаба» заключается в распространении теста кейса, предоставленного «узлу», который соответствует возможностям / требованиям, необходимым для выполнения кросс-браузерного тестирования. Как только «хаб» получает вход-

ной / тестовый файл, который необходимо выполнить, он ищет узел, который соответствует требуемым возможностям, и перенаправляет «запрос выполнения теста» на соответствующий узел. Таким образом с помощью Selenium grid время работы тестов сократилось, так как тесты запускаются одновременно в 2 потока. Результат представлен в таблице 1.

Таблица 1 – Результат выполнения тест кейсов

Узел (node)	Кол-во тест кейсов	Время вып. тест-я (один поток)(мин)	Время при исп. Selenium grid(мин)
Windows 10 Chrome Firefox	6	9.18	15.27
Linux Chrome Firefox	6	9.36	

IV. ЗАКЛЮЧЕНИЕ

В ходе проведенной работы был проведен обзор имитационного моделирования, рассмотрена структура грид-системы, проведен анализ грид системы, показаны преимущества использования. Рассмотренная грид система на примере Selenium grid в сфере автоматизированного тестирования показала преимущество ее использования для автоматизации этого процесса. Исходя из проведенной работы, очевидно, что грид система - технология активно развивающаяся и идеально подходит для решения многих вычислительных задач.

СПИСОК ЛИТЕРАТУРЫ

1. Интернет портал по Selenium Grid технологиям [Электронный ресурс]. – Режим доступа: <http://automation-remarks.com/nastraivaiem-selenium-grid-za-5-minut/> – Дата доступа: 8.10.2019.
2. Васенин В. А., Эволюция технологии Grid/ Васенин В. А., Шундеев А. С. / Информационные технологии., - 2012.- С.2-9.
3. Интернет портал по Selenium [Электронный ресурс]. – Режим доступа: <https://selenium2.ru/docs/selenium-grid.html> - Дата доступа: 8.10.2019.
4. Интернет портал по Grid технологиям [Электронный ресурс]. – Режим доступа: <http://grid.jinr.ru/> – Дата доступа: 6.10.2019.
5. 2. The Grid 2: Blueprint for a new computing infrastructure: Databook/ Elsevier - Foster Ian, Carl Kesselman, 2003.

АНАЛИЗ МЕТОДА СКАНИРОВАНИЯ ЛВС

Савик К. В.

Факультет информационных технологий и управления, Белорусский государственный университет
информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: zahariev@bsuir.by, hootkich@gmail.com

Сетевая безопасность состоит из аппаратных и программных компонентов, предназначенных для защиты данных и информации, обрабатываемой в сети. Кроме того, эти компоненты обеспечивают установку профилактических мер для защиты сетевой инфраструктуры и ее данных от несанкционированного доступа, изменения данных, повреждения и несанкционированного раскрытия. В конечном счете, сетевая безопасность предназначена для создания безопасной среды, в которой пользователи компьютеров, программ и мобильных приложений могут выполнять компьютерные или цифровые действия без сетевых уязвимостей.

ВВЕДЕНИЕ

Надежность и безопасность сети крайне важны в мире, где компьютерные сети являются ключевым элементом в коммуникациях и транзакциях между объектами. Сетевые администраторы, правительство, консультанты по безопасности и хакеры использовали различные инструменты для проверки уязвимостей целевых сетей, таких как, например, возможность удаленного доступа к компьютерам в сети и управления ими без авторизации. Благодаря этому интенсивному тестированию целевая сеть может быть «защищена» от распространенных уязвимостей и эзотерических атак. Однако существующие системы тестирования дают противоречивые результаты, используют недоказанные методы или наносят ущерб целевой сети.

Таким образом, целью данной работы является исследование метода сканирования узлов в сети, частности на открытые порты, на примере утилиты "nmap"(Network Mapper).

I. NMAP

NMAP (Network Mapper) — это утилита с открытым исходным кодом для исследования сети и проверки безопасности. Она была разработана для быстрого сканирования больших сетей, хотя прекрасно справляется и с единичными целями. Nmap использует "сырые" IP пакеты оригинальным способом, чтобы определить какие хосты доступны в сети, какие службы (название приложения и версию) они предлагают, какие операционные системы (и версии ОС) они используют, какие типы пакетных фильтров/брандмауэров используются и еще множество других характеристик. В то время, как Nmap обычно используется для проверки безопасности, многие системные администраторы находят ее полезной для обычных задач, таких как контролирование структуры сети, управление расписаниями запуска служб и учет времени работы хоста или службы.

Выходные данные Nmap это список просканированных целей с дополнительной информацией по каждой из них в зависимости от заданных опций. Ключевой информацией является «таблица важных портов». Эта таблица содержит номер порта, протокол, имя службы и состояние. Состояние может иметь значение open (открыт), filtered (фильтруется), closed (закрыт) или unfiltered (не фильтруется). Открыт означает, что приложение на целевой машине готово для установки соединения/принятия пакетов на этот порт. Фильтруется означает, что брандмауэр, сетевой фильтр, или какая-то другая помеха в сети блокирует порт, и Nmap не может установить открыт этот порт или закрыт. Закрытые порты не связаны ни с каким приложением, но могут быть открыты в любой момент. Порты расцениваются как не фильтрованные, когда они отвечают на запросы Nmap, но Nmap не может определить открыты они или закрыты. Nmap выдает комбинации открыт|фильтруется и закрыт|фильтруется, когда не может определить, какое из этих двух состояний описывает порт. Эта таблица также может предоставлять детали о версии программного обеспечения, если это было запрошено. Когда осуществляется сканирование по IP протоколу (-sO), Nmap предоставляет информацию о поддерживаемых протоколах, а не об открытых портах.

II. Методы сканирования

Синтаксис запуска программы следующий: Nmap [Scan type(s)] [options] target specification, где вместо Scan type указывается тип сканирования (по умолчанию, если это место оставить пустым, Nmap будет открыто сканировать доступные порты). В качестве options вводятся всевозможные ключи и параметры сканирования, а вместо target specification — либо IP-адрес компьютера, либо диапазон IP-адресов (который определяется маской подсети), либо название хоста.

Когда запускается сканер Nmap, и начинается сканирование портов, сначала идет запрос

пингом, а затем поэтапно сканируются и порты. Следовательно, не ответив на пинг, хост сканироваться не будет. Этот подход можно считать одним из методов защиты от сканирования.

Пример работы nmap:

nmap -v scanme.nmap.org - сканер проверит хост на наличие открытых портов и служб, которые слушают эти порты.

Nmap scan report for scanme.nmap.org (45.33.32.156) Not shown: 994 closed ports PORT STATE SERVICE 22/tcp open ssh 80/tcp open http 139/tcp filtered netbios-ssn 445/tcp filtered microsoft-ds 9929/tcp open nping-echo 31337/tcp open Elite

Nmap done: 1 IP address (1 host up) scanned in 7.00 seconds

Из показанного вывода можно увидеть, что открыты 22, 80, 9929, 31337 порты и т.к. они имеют состояние open, то это означает, что приложение на целевой машине готово к принятию пакетов на указанный порт. Порты 139 и 445 с состоянием filtered показывают, что брандмауэр, фильтр, или что-то другое в сети блокирует порт, так что Nmap не может определить, является ли порт открытым или закрытым.

III. ПРИМЕР ИСПОЛЬЗОВАНИЯ NMAP

В эффективной атаке необходимо проанализировать цель и используемые технологии, чтобы знать, какой тип атаки следует запустить. Запускать атаки, относящиеся к уязвимостям UNIX, если цель работает только на серверах Microsoft, не имеет смысла. Небольшое время, затрачиваемое на исследования, экономит много времени при атаке проникновения.

Цель сканирования узлов - найти следующую информацию:

- IP-адреса хостов в целевой сети;
- операционные системы на целевых системах;
- доступные порты протокола пользовательских дейтаграмм (UDP) и протокола управления передачей (TCP) в целевых системах.

В качестве примера, возьмем последнюю цель сканирования узлов для получения несанкционированного доступа. В исследовательский целях, сканируется локальная машина.

nmap -p "*"localhost - сканирует localhost на все порты. Nmap scan report for localhost (127.0.0.1) Host is up (0.000065s latency). Not shown: 8287 closed ports PORT STATE SERVICE 22/tcp open ssh 139/tcp open netbios-ssn 445/tcp open microsoft-ds 631/tcp open ipp 3000/tcp open ppp 5432/tcp open postgresql

Из результата особо интересует 22 порт, доступный по ssh. Для получения доступа по ssh необходимо знать пользователя и пароль. В качестве примера будет использован другой пользователь. Пароль будет получен с помощью утилиты hydra.

```
hydra -l k.savik -P ./pass.txt localhost -t 4 ssh [DATA] max 4 tasks per 1 server, overall 4 tasks, 7 login tries, 2 tries per task [DATA] attacking ssh://localhost:22/ [22][ssh] host: localhost, login: k.savik password: 1111111 1 of 1 target successfully completed, 1 valid password found
```

Как видно из вывода, пароль - 1111111. И с помощью его можно получить удаленный доступ по ssh.

IV. ВЫВОД

В ходе проведенной работы был проведен обзор сканирования узлов при помощи утилиты nmap, а так же ее практическое применение в связке с утилитой hydra для получения удаленного доступа по ssh протоколу. Исходя из проведенной работы, стало видно, насколько важен этап сбора информации о узлах в сети и насколько уязвимыми они могут быть.

1. Weir, Matt, et al. "Password cracking using probabilistic context-free grammars." Security and Privacy, 2009 30th IEEE Symposium on. IEEE, 2009.
2. <http://xgu.ru/downloads/mac2port>
3. <https://www.securitylab.ru/analytics/216229.php>
4. Wang, K. C. SYSTEMS PROGRAMMING IN UNIX/LINUX. Springer, 2018.

АВТОМАТИЗАЦИЯ УЧЁТА АКЦИЗНОЙ ПРОДУКЦИИ ДЛЯ ЕГАИС

Сманцер М. С., Андрухович А. К.

Кафедра информатики, Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: drowmar@gmail.com, anastasia.andruhovich@gmail.com

В данной статье рассматриваются вопросы автоматизации процесса учёта акцизной алкогольной продукции на предприятиях Республики Беларусь. В Беларуси имеется множество производителей алкогольной продукции, и многие из них импортируют алкоголь в Российскую Федерацию, где введён закон о том, что вся производимая алкогольная продукция, прежде чем пойти на прилавки магазинов должна быть зарегистрирована в ЕГАИС – единой государственной автоматизированной информационной системе. Предлагается решение автоматизированного учёта акцизной продукции на отечественных предприятиях для ЕГАИС, реализованное с использованием современных технологий: WCF, языка программирования C#, WPF, средств NiceLabel.

ВВЕДЕНИЕ

Республика Беларусь активно импортирует производимую продукцию в различные страны, в том числе и в Российскую Федерацию. В соответствии с действующим федеральным законом Российской Федерации, производство и все ступени последующей реализации алкоголя, вплоть до продажи конечному потребителю, должны проверяться и учитываться в единой информационной системе. ЕГАИС – это единая государственная автоматизированная информационная система, предназначенная для государственного контроля над объёмом производства, импортом и оборотом этилового спирта, алкогольной, спиртосодержащей продукции. Каждое предприятие выбирает свой способ учёта продукции. Одни предприятия выделяют данную работу определённым людям, которые поштучно считают и маркируют производимый продукт. Однако такой подход достаточно непродуктивен. Другие же выбирают более современное решение данной проблемы: разрабатывают системы для маркировки и учёта продукции. А с учётом того, что алкогольная продукция контролируется на государственном уровне, автоматизация учёта является единственным возможным решением.

Автоматизация учета готовой продукции дает возможность оперативно получать информацию о наполнении склада и его остатках, обеспечивает точное соответствие идентификационного кода во всех системах, а также позволяет в режиме реального времени просмотреть статус выполнения процессов. Базы данных позволяют структурировать, хранить и извлекать информацию оптимальным для пользователя способом.

Одной из важных характеристик внедрения таких систем является то, что у каждого предприятия уже существует свой налаженный производственный процесс, детали и особенности которого необходимо тщательно изучить прежде, чем начать внедрение.

I. ОПИСАНИЕ СИСТЕМЫ

Автоматизированная система учёта акцизной продукции для ЕГАИС включает в себя сопровождение всего цикла производства: от маркирования бутылок, до отгрузки паллет. Она позволяет контролировать объем производства и сбыт алкогольной продукции. Данный программный продукт позволяет изучать состояние производительности и оборотоспособности алкоголя, а также вести строгий надзор за акцизами. Все участники рынка алкоголя обязаны направлять данные о каждой марке, принятой на склад, а также при списании бутылок с акцизами. Автоматизированные системы учёта позволяют отслеживать передвижение каждой единицы товара непосредственно от производственного комбината до конечного покупателя.

Автоматизированная система состоит из следующих подсистем:

- центральной базы данных;
- сервиса доступа к центральной базе данных;
- клиентского приложения администратора для просмотра центральной базы и настройки параметров системы;
- клиентского приложения технолога/мастера для настройки параметров выпуска и упаковки продукции ;
- клиентского приложения оператора участка контроля и увязки ШК акцизной марки и колпачка бутылки автоматической линии;
- клиентского приложения оператора участка упаковки коробов автоматической линии;
- клиентского приложения оператора участка формирования паллет автоматической линии;
- клиентского приложения оператора участка упаковки коробов ручной линии;

- клиентского приложения оператора участка формирования паллет ручной линии;
- клиентского приложения кладовщика для отгрузки продукции и формирования файлов партнерам;
- средств экспорта/импорта;
- программно-аппаратного комплекса на мобильном терминале сбора данных для отгрузки продукции.

Каждое рабочее место разработанной системы является независимым. Центром всей системы является сервис. Остальные рабочие места подключаются к сервису для доступа к необходимой информации по производству, а также отправке данных, получаемых в процессе производства. Система работает в режиме полного онлайн. Данный режим является более оптимальным, по сравнению с режимом псевдо-онлайн, так как вся работа с базой данных ведётся на сервере, что освобождает рабочие места от лишней нагрузки.

II. ВНЕДРЕНИЕ

Для внедрения системы необходимо установить всё необходимое оборудование в цеху производства. После этого на каждое рабочее место устанавливается необходимое программное обеспечение. Администратором системы настраиваются все необходимые параметры системы, создаются этикетки средствами NiceLabel.[1] А также заполняются необходимые для производства справочники данных с помощью автоматизированного рабочего места управления комплексом.

III. ЭТАПЫ ПРОИЗВОДСТВА

1. До производства.

На рабочем месте технолога или мастера отображается текущая информация о состоянии выполнения заказа. Пользователь может выполнить подготовительную работу, продолжить выполнение ранее начатого заказа, ввести новый заказ или закончить заказ. Также мастеру доступен просмотр данных о произведённой продукции.

2. Производство.

В зависимости от необходимости предприятия, производство может быть как ручным, так и автоматически. В связи с этим меняется необходимое оборудование и настройки системы. Также система может быть настроена и таким образом, что некоторые рабочие места являются автоматическими, другие же ручными.

Последовательность производственного процесса следующая. Первым этапом является регистрации бутылок. После этого из зарегистрированных бутылок необходимо собрать короба. И далее из коробов необходимо собрать паллеты.

Отличие автоматической линии от ручной следующее. В первом варианте для производства достаточно запустить необходимые программы на рабочих местах, нажать кнопку старта на каждом соответственно. Остальные действия происходят по ходу производства практически без вмешательства человека. Регистрация бутылок выполняется сканером, расположенным над конвейером. Перед сбором коробов выполняется сканирование бутылок, и после этого происходит печать этикетки на короб. И далее после успешной верификации наклеенной этикетки короб по конвейеру отправляется на паллету.

В ручной же линии отсутствует конвейер. Аналогом сканеров, используемых на автоматической линии, являются ручные сканеры, а принтеры-апликаторы заменены обычными производственными принтерами. Различные внештатные ситуации, возникающие на автоматических линиях, также решаются с использованием ручных операций.

3. После производства.

Работа мастера по отгрузке заключается в фиксации факта производства сканированием паллет. Данный пользователь выполняет создание заказа на отгрузку с указанием удостоверения качества и количества, загружает данный заказ на ТСД. После сканирования необходимых паллет выполняет выгрузку заказа с ТСД. Конечным этапом производства является формирование файлов для передачи партнерам.

IV. ИСПОЛЬЗУЕМЫЕ ТЕХНОЛОГИИ

Все части системы написаны на языке программирования C#. Программный продукт состоит из различных частей и имеет архитектуру сервер-клиент. Протокол общения между сервисом и клиентами реализован с помощью технологии WCF. Одним из элементов разрабатываемой системы является программа терминала сбора данных. На данном устройстве установлена Windows CE, что позволило также реализовать программу на C#. Для реализации десктопных клиентов была выбрана технология WPF. Печать производится средствами NiceLabel.[2–4]

СПИСОК ЛИТЕРАТУРЫ

1. Egais [Электронный ресурс] / Требования ЕГАИС к маркировке паллет и коробов. – Режим доступа: <http://egais.ru/news/view?id=1543>. – Дата доступа: 10.09.2019.
2. CLR via C#. Программирование на платформе Microsoft.NET Framework 4.5 на языке C#. / Д. Рихтер. – СПб.: Мастер-класс, 2017. – 896 с.
3. Pro SQL Server Internals. / D. Korotkevitch. – СПб.: Apress, 2016. – 804 с.
4. NiceLabel [Электронный ресурс] / ПО для печати. – Режим доступа: <https://www.nicelabel.com/ru/downloads/user-guides>. – Дата доступа: 10.09.2019.

ОБЗОР И АНАЛИЗ АЛГОРИТМОВ ДИСПЕТЧЕРИЗАЦИИ ГРУППЫ ЛИФТОВ

Струц Д. А.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: dmitry.a.struts@gmail.com

Проведен обзор и описание существующих алгоритмов диспетчеризации группы лифтов, отмечены их достоинства и недостатки, описаны перспективные направления для развития.

ВВЕДЕНИЕ

В городских районах строится всё больше высотных зданий. Как следствие, существует потребность в новых методах автоматизации зданий. В каждом высотном здании система управления группой лифтов необходима для достижения оптимальной производительности. Проблема, которую необходимо решить, это назначить для каждого вызова на этаже лифт, который минимизирует предварительно выбранную целевую функцию. Традиционно главной целью была минимизация времени ожидания пассажиров, чтобы подняться на лифте, которое определяется как фактическое время ожидания предполагаемого пассажира после регистрации вызова, пока не откроются двери прибывшего лифта. В последнее время растет интерес к другим критериям, например, необходимое ограничение максимального времени ожидания и размеров очереди людей в ожидании лифта, потребления энергии, из соображений стоимости и устойчивости [1-4]. Так как проблема диспетчеризации является NP-трудной, большинство эффективных подходов используют наработки в сфере искусственного интеллекта (ИИ).

ОПИСАНИЕ СТРАТЕГИЙ ДИСПЕТЧЕРИЗАЦИИ ГРУППЫ ЛИФТОВ

Широкое распространение получил круговой алгоритм управления, основной целью которого является достижение равной загрузки каждого лифта. Вызовы распределяются по мере их поступления последовательным образом по отдельным лифтам. Его преимуществами является простота реализации, равномерное распределение нагрузки между лифтами, а также обеспечение выполнения требований пассажиров на приемлемом уровне при неинтенсивном пассажиропотоке.

Алгоритм трех переходов, используется для определения последовательности обслуживания вызовов с этажей. Алгоритм разбивает все вызовы на 3 категории, по количеству изменения направлений движения для обработки вызова [5].

Также используют идею зонирования высотных зданий по вертикали, которая заключается в разделении здания на несколько приле-

гающих друг к другу зон и каждый из лифтов обрабатывает вызовы с этажей только зоны, назначенной для обслуживания данным лифтом. Существует две группы по признаку фиксации зон: статичные или динамические. Оптимизация проводится при помощи правильного определения размеров зон в каждом конкретном случае. Считается, что динамическое разбиение на зоны работает лучше, так как способно адаптироваться к изменениям в потоке движения, а производительность может быть улучшена с помощью других алгоритмов с применением ИИ [6,7].

В процессе развития алгоритмов управления лифтов в составе группы появились динамические стратегии автоматизированного управления лифтами, способные подстраиваться к изменениям условий в процессе функционирования [8]. Оптимизация может быть достигнута в режиме реального времени, параллельно или путем последовательного моделирования в течение определенного периода (например, ночью), используя информацию, предварительно сохраненную в базу данных [9]. Стратегия на основе поиска, сущность которой заключается в нахождении оптимального решения при каждом новом возникающем событии, при этом формируется определенная реакция на данное событие. Алгоритм поиска производит решение задачи назначения, которая заключается в распределении зафиксированных вызовов между группой лифтов с учётом команд из кабины лифта. Если количество свободных лифтов больше числа произведённых вызовов, то алгоритм распределит их в соответствии с определёнными критериями. Если количество вызовов превышает количество свободных лифтов, то из множества заявок будут выделены те, обслуживание которых обеспечит минимум целевой функции. Недостатком данной стратегии является статичность в интервале формирования решений, что означает тот факт, что новые вызовы, которые поступили после назначения не имеют возможности изменить решение системы управления. Для нивелирования данного недостатка применяется переопределение управляющего воздействия, которое подразумевает корректировку решения согласно с новой текущей ситуации. Стоит отме-

титель, что данное решение существенно повышает вычислительную сложность алгоритма. По этой причине данный метод целесообразно применять при достаточно высокой плотности пассажиропотока и когда другие подходы не дают существенного повышения эффективности.

Большое количество алгоритмов управления, которые реализуются на базе контроллеров, относятся к стратегии, описываемой правилами в терминах «IF-THEN» логики. Нечеткая логика представляет собой комбинацию численных и символических методов для получения точных результатов от неточных данных. Модель предложенная в [10,11] хорошо совпадает с образом человеческим правил нечеткого вывода в тех случаях, когда система имеет много входов и много выходов. Использование нечёткой логики может существенно расширить возможности методов, основанных на определении правил, и повысить уровень вариативности системы управления.

Наряду с этим имеется возможность решать задачи управления с применением генетических алгоритмов, которые подразумевают имитацию процесса эволюции – гены родителя наследуются потомками. Генетические алгоритмы рассматриваются как совокупность отдельных шагов. Для имитации процесса эволюции требуется создать множество правил, на основе которых формируются промежуточные решения. Сам же процесс эволюции представляет собой «слияние» генов получение в итоге гена-потомка – определённого решения. Генетические алгоритмы получили широкое применение и могут применяться для получения хороших результатов [12-15]. Недостаток заключается в их высокой трудоёмкости, обусловленная итерационным характером реализации алгоритмов, а эффективность алгоритма определяется исходным набором правил.

ЗАКЛЮЧЕНИЕ

В рассмотренных стратегиях диспетчеризации группы лифтов существует компромисс между качеством и скоростью получения результата. Динамические стратегии имеют более высокую вычислительную сложность и сильно зависят от качества выбранной целевой функции, в то время как классические стратегии предоставляют быстрое решение, но не способны эффективно обслужить плотный пассажиропоток. В последнее время существует большой интерес в проектировании адаптивных комбинационных алгоритмов, а также увеличения количества входных данных с помощью использования дополнительных датчиков для принятия решения.

1. Fernández, J.R., Cortés, P., Guadix, J., Muñozuri, J. (2013). Dynamic fuzzy logic elevator group control

system for energy optimization. *International Journal of Information Technology Decision Making* 12 (3), 591-617.

2. Hasan, M.Z., Fink, R., Suyambu, M.R., Baskaran, M.K. (2012). Assessment and improvement of elevator controllers for energy efficiency. In *IEEE International Conference on Consumer Electronics* art. no. 6241747.
3. Liu, Y., Hu, Z., Su, Q., Huo, J. (2010). Energy saving of elevator group control based on optimal zoning strategy with interfloor traffic. In *Proceedings – 3rd International Conference on Information Management, Innovation Management and Industrial Engineering, ICII 3*, 328-331.
4. Tyni, T., Ylinen, J. (2006). Evolutionary bi-objective optimization in the elevator car routing problem, *European Journal of Operational Research* 169(3), 960-977.
5. А.П. Кузнецов, В.Н. Никонов, А.С. Шмаловский, М.В. Силивонец / Алгоритмы управления лифтами с использованием семантической информации. // *Известия ТулГУ. Технические науки*. 2013. вып. (№) 7.С 531-536[
6. Xu, Y., Luo, F., Wang, J. (2004). A new modeling method for elevator group control system with cellular automata. In *Proceedings of 5th World Congress on the Intelligent Control and Automation, WCICA4*, 3596-3599.
7. Li, Z., Tan, H.-Z., Zhang, Y.-N., Mao, Z.-Y. (2007b). Dynamic optimization of elevator group control based on artificial immune algorithm for inter-floor peak traffic during lunch-time. *Control Theory and Applications* 24 (2), 177-182.
8. Utgoff, P.E, Connell, M.E. (2012). Real-time combinatorial optimization for elevator group dispatching. *IEEE Transactions on Systems Man and Cybernetics Part A: Systems and Humans* 42(1), 130-146.
9. Liu, J., Liu, Y. (2007). Ant colony algorithm and fuzzy neural network based intelligent dispatching algorithm of an elevator group control system. In *Proceedings of the IEEE International Conference on Control and Automation*, 2306-2310.
10. Y. Naitoh, T. Furuhashi and Y. Uchikawa, “A Variable Ordinal Structure Model for Fuzzy Reasoning and Its Application to Decision Problem of Working Order”, *International Conference On Industrial Electronic, Control and Instrumentation*, Vol. 2, pp 1539-1543, 1991.
11. T.Ohnishi, “Fuzzy Reasoning by Ordinal Structure Model of Control Rule”, *Journal of Japan Society for Fuzzy Theory and Systems* , Vol.2, No. 4, pp 125-132, 1990.
12. Cortés, P., Larrañeta, J., Onieva, L. (2004). Genetic algorithm for controllers in elevator groups: analysis and simulation during lunchpeak traffic. *Applied Soft Computing* 4 (2), 159-174.
13. Hirasawa, K., Eguchi, T., Zhou, J., Yu, L., Hu, J., Markon, S. (2008). A double-deck elevator group supervisory control system using genetic network programming. *IEEE Transactions on Systems, Man, and Cybernetics Part C, Applications and Reviews* 38 (4), 535-550.
14. Bolat, B., Cortés, P., Yalcin, E., Alisverisci, M. (2010). Optimal car dispatching for elevator groups using genetic algorithms. *Autosoft - Intelligent Automation and Soft Computing* 16 (1), 89-99.
15. Chen, T., Hsu, Y., Huang, Y. (2012). Optimizing the intelligent elevator group control system by using genetic algorithm. *Advanced Science Letters* 9, 957-962

ОТОБРАЖЕНИЯ ТЕПЛОВЫХ ИСТОЧНИКОВ НА КАРТЕ МЕСТНОСТИ, ОСНОВАННОЕ НА МЕТОДЕ ВЫДЕЛЕНИЯ ОБЪЕКТОВ С ТЕПЛОВЫМ ИЗЛУЧЕНИЕМ НА СТАТИЧЕСКИХ RGB ИЗОБРАЖЕНИЯХ

Тарасюк Е. В.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: tarasiuk-ev@bsuir.by

Многие направления науки, техники и производства в значительной степени ориентированы на развитие систем, в которых информация носит характер поля (изображения). При обработке такой информации возникает ряд сложных научных и технических проблем. Одной из самых сложных задач является обработка изображений. О важности этой проблемы говорит тот факт, что исследования по распознаванию образов, анализу изображений и речи включены в перечень приоритетных направлений развития науки и техники.

ВВЕДЕНИЕ

Распознавание изображений находит применение в различных приложениях: контроль топологии печатных плат, текстуры ткани, контроль доступа к информации по идентификации личности (биометрическая идентификация), доступ к объектам ограниченного доступа, оперативный поиск в картотеке изображений, дактилоскопия и др. [1]. Для того чтобы научить компьютер распознавать объекты и реагировать на их движения, требуются устойчивые алгоритмы анализа и распознавания изображений [3].

I. ОБНАРУЖЕНИЯ ОБЪЕКТОВ ПРИ ПОМОЩИ ЦВЕТОВОЙ СЕГМЕНТАЦИИ

Существующие алгоритмы обнаружения объектов можно разделить на две категории: методы эмпирического распознавания и методы моделирования изображения [4].

Среди эмпирических методов обнаружения широкое распространение получили методы, использующие цветовую сегментацию областей. Это обусловлено тем, что использование выбранного цвета как признака для обнаружения объекта имеет ряд важных преимуществ: малая вычислительная сложность и высокая скорость обработки, устойчивость к изменению освещения, устойчивость к изменению ориентации и масштаба объекта, устойчивость к изменению формы объекта и частичного перекрытия.

К числу недостатков данного метода можно отнести потенциальную возможность ложного обнаружения предметов с цветом, близким к заданному и зависимость от цветового баланса камеры и освещения [4].

Несмотря на имеющиеся в методе недостатки, перечисленные преимущества метода обнаружения объектов по цветовому параметру позволяют использовать выбранный диапазон цветов в качестве признака для распознавания объекта. Основным этапом обнаружения объекта по цве-

товому параметру является выделение пикселей, близких по цвету к заданному.

Для выделения пикселей используется модель распределения оттенка в цветовом пространстве. Из существующих методов цветовой сегментации оптимальным является метод моделирования распределения с помощью байесовской карты вероятностей [5, 6]. Суть метода заключается в вычислении отношения $\frac{P(\text{skin}/c)}{P(-\text{skin}/c)}$ вероятностей принадлежности цвета к оттенку объекта для каждой точки дискретизированного цветового пространства по формуле и сравнении значения $P(\text{skin}/c)/P(-\text{skin}/c)$ с некоторым выбранным порогом θ .

$$P_{out} = 2 \sum_{r \in \Omega_2} \frac{P_{skin}(x)}{S}$$

На рисунке 1 отображен пример выделения области из неидеальных исходных данных.

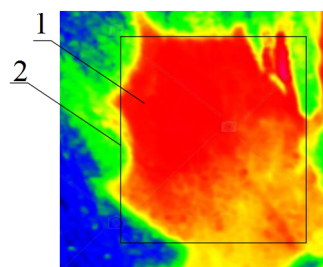


Рис. 1 – Пример выделения области из неидеальных исходных данных. 1 – область пикселей цвета объекта; 2 – выделенная область объекта

Значения $P(c/\text{skin})$ и $P(c/-\text{skin})$ непосредственно вычисляются из набора тренировочных изображений, сегментирование которых производится вручную. Априорные вероятности $P(\text{skin})$ и $P(-\text{skin})$ также можно вычислить исходя из тренировочного набора. Однако эти вероятности можно принять равными 0.5, поскольку изменение априорных вероятностей влияет лишь

на величину порога, используемого для обнаружения объекта, и не приводит к изменению соотношения ложных обнаружений и ложных промахов. [7]

Данная модель обладает высокой вероятностью распознавания, малой вероятностью ложного обнаружения и инвариантностью по отношению к выбору цветового пространства, в котором проводится сегментация.

Результатом цветовой сегментации является информация о степени близости цвета каждого пикселя к заданному цвету. Результатом полученной информации является полутоновое изображение, где интенсивность пикселей установлена равными характеристике близости пикселя к заданному цвету.

II. ОБОБЩЕННЫЙ АЛГОРИТМ ОБРАБОТКИ И ПЕРЕДАЧИ ИЗОБРАЖЕНИЯ

Первостепенной задачей в системах определения объектов с передачей данных о его местонахождении и размерах является определение наличия объекта на исследуемой поверхности. Способов решения данной задачи множество, однако самым простым и надежным способом является использование лазерного дальномера и инфракрасной камеры.

Исходя из длины волны, инфракрасное излучение делят на несколько спектральных частей. При этом нет единого определения границ каждой отдельной части. Ближний ИК-диапазон отмечен длинами волн, приближенных до конечной части спектра видимого света. Примерный расчётный отрезок волны обозначен длиной: 730 — 1300 нм. Частота излучения составляет примерно 215-400 Гц. Короткий ИК-диапазон излучает минимум тепла.

Средний ИК-диапазон, охватывает длины волн 1300-3000 нм. Частоты здесь измеряются диапазоном 20-215 ТГц. Уровень излучаемого тепла относительно невысок.

Дальний ИК-диапазон наиболее близок к диапазону микроволн. Расклад: 3-1000 мкм. Частотный диапазон 0,3-20 ТГц. Эту группу составляют короткие длины волн на максимальном частотном отрезке где излучается максимум тепла.

После формирования изображения в инфракрасном диапазоне производится обнаружение объекта при помощи цветовой сегментации. После определения границ объекта производится удаление пикселей, не принадлежащих объекту. При этом масштаб и разрешение изображения изменяются, а предыдущие значения этих пара-

метров сохраняются для наложения полученного изображения на карту местности, которая должна соответствовать масштабу изображения.

Сформированное изображение передается по цифровому, или аналоговому каналу связи по приема-передающему оборудованию на стационарный пункт обработки информации и контроля.

Полученное изображение обрабатывается и восстанавливает исходный масштаб. Для ликвидации ошибочного наложения изображения на карту местности производится проверка. После чего происходит наложение изображения на карту местности.

Наиболее успешно задача данных и ее восстановление производится, если функция передаваемого сигнала является статической: $\lambda(x, y)$. При работе с динамической функцией $\lambda(x, y, t)$ вероятность ошибки, а так же время передачи данных и наложение на карту местности значительно возрастают. В связи с чем рекомендуется использовать получение статического изображения, а не видеофайла.

СПИСОК ЛИТЕРАТУРЫ

1. Ваиник В. Н., Червоненкис А. Я. Теория распознавания образов // Стохастические проблемы обучения. М.: Наука, 1974. С. 144-157.
2. Самаль Д. И., Старовойтов В. В. Обнаружение антропометрических точек лица с помощью фильтров Габора // Цифровая обработка изображений. М.: Наука, 2001. С. 141-150.
3. Визильтер Ю. В., Желтов С. Ю., Ососков М. В. Система распознавания и визуализации характерных черт человеческого лица в реальном времени на персональной ЭВМ с использованием web-камеры // Тр. Междунар. конф. по компьютерной графике и машинному зрению
4. Rabiner L. R. A tutorial on hidden Markov models and selected applications in speech recognition // IEEE Sci. Works Compilat. 1989. V. 77, N 2. P. 257-285.
5. Jordao L., Perrone M., Costeira L., Santos L. Active face and feature tracking // Proc. of the Intern. conf. on lineage analysis and processing. Lisbon: S. n., 1999. V. 9. P. 572-576.
6. Albiol A., Torres L., Delp E. J. Optimum color spaces for skin detection // Proc. of the IEEE Intern. conf. on image processing, Thessaloniki, Oct. 8-10 2001. IEEE Conf. Publ., 2001. V. 1. P. 122-124.
7. Друки А.А. Применение деформируемых эллиптических моделей для выделения лиц на изображениях и в видеопотоке. [Электронный ресурс]/Режим работы: <https://cyberleninka.ru/article/v/algorithmy-vydeleniya-lits-na-staticheskikh-rgb-izobrazheniyah-i-v-videopotoke/>. - Дата доступа: 01.10.2019
8. Baskan S., Bulut M. A., Atalay V. Projection based method for segmentation of human face and evaluation // Pattern Recognition Lett. 2002. V. 20. P. 1623-1629

ВЛИЯНИЕ НАДЕЖНОСТИ ЛЕГИТИМНОГО ПРИЕМНОГО ОБОРУДОВАНИЯ НА ВЕРОЯТНОСТЬ ОШИБОЧНОЙ РЕГИСТРАЦИИ ДАННЫХ В КВАНТОВО-КРИПТОГРАФИЧЕСКИХ КАНАЛАХ СВЯЗИ

Тимофеев А. М.

Кафедра защиты информации, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: tamvks@mail.ru

Применительно к квантово-криптографическому каналу связи получено выражение для оценки вероятности ошибочной регистрации символов «1», учитывающее мертвое время приемного модуля, в качестве которого использован счетчик фотонов. Установлено, что при прочих равных параметрах рост средней длительности мертвого времени продлевающегося типа приводит к увеличению средних скоростей счета сигнальных импульсов, при которых достигаются наименьшие значения вероятностей ошибочной регистрации символов «1».

ВВЕДЕНИЕ

В настоящее время достаточно интенсивно развиваются системы связи на базе квантово-криптографических каналов. Это обусловлено тем, что такие системы имеют весьма высокий уровень информационной безопасности и, в частности, позволяют обеспечивать абсолютную скрытность и конфиденциальность передаваемых данных [1]. Однако существуют трудности их практической реализации, которые в основном связаны с тем, что данные передаются посредством предельно слабого оптического излучения, содержащего в среднем не более десяти фотонов на каждый передаваемый бит (символ). Для регистрации такого излучения применяют высокочувствительные приемные модули – счетчики фотонов [1, 2]. При этом особенно важно обеспечивать высокую надежность функционирования используемого оборудования [3, 4].

Под надежностью будем понимать свойство оборудования выполнять возложенные на него функции информационной безопасности с сохранением своих характеристик (параметров) в определенных пределах при данных условиях эксплуатации.

Одним из критериев надежности является вероятность ошибочной регистрации данных, методики оценки которой описаны в [5]. Однако эти методики не применимы для квантово-криптографических каналов связи, поскольку не учитывают такой важный параметр приемного модуля, как мертвое время [1, 2]. В течение этого времени приемный модуль не чувствителен к падающему на него оптическому излучению, в результате чего возникают так назы-

ваемые просчеты, которые влияют на количество ошибок при регистрации данных. В связи с этим целью данной работы являлось установить влияние мертвого времени приемного модуля квантово-криптографического канала связи на вероятность ошибочной регистрации данных.

Объектом исследования являлся квантово-криптографический канал связи, в котором в качестве приемного модуля использован счетчик фотонов.

Предметом исследования являлось установить влияние продлевающегося мертвого времени счетчика фотонов на вероятность ошибочной регистрации двоичных символов «1». Данным типом мертвого времени характеризуются счетчики фотонов, построенные на базе лавинных фотоприемников, включенные по схеме пассивного гашения лавины.

I. ВЫРАЖЕНИЕ ДЛЯ ОЦЕНКИ ВЕРОЯТНОСТИ ОШИБОЧНОЙ РЕГИСТРАЦИИ СИМВОЛОВ «1»

Дальнейшие рассуждения будут основаны на том, что передача информации осуществляется с использованием дискретного двоичного асинхронного однородного квантово-криптографического канала связи без памяти и со стиранием. Всеми потерями информации, за исключением потерь в счетчике фотонов, пренебрегаем.

Учитывая статистические распределения, полученные в работах [6, 7], применительно к счетчикам фотонов с рассматриваемым типом мертвого времени запишем выражение для вероятности ошибочной регистрации символов «1»:

$$P_{osh1} = \sum_{N=0}^{N_2} \frac{[(n_t + n_{s1}) (\Delta t - \tau_d)]^N \exp[-(n_t + n_{s1}) (\Delta t - \tau_d)]}{N!}, \quad (1)$$

где N_1 и N_2 – нижний и верхний пороговые уровни регистрации соответственно, n_t – средняя скорость счета темновых импульсов на выходе счетчика фотонов, n_{s1} – средняя скорость счета сигнальных импульсов на выходе счетчика фотонов при передаче символов «1», Δt – среднее время однофотонной передачи, τ_d – средняя длительность мертвого времени продлевающегося типа.

II. РЕЗУЛЬТАТЫ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ И ИХ ОБСУЖДЕНИЕ

На рис. 1 представлена зависимость вероятности ошибочной регистрации символов «1» от средней скорости счета сигнальных импульсов для различной средней длительности мертвого времени продлевающегося типа.

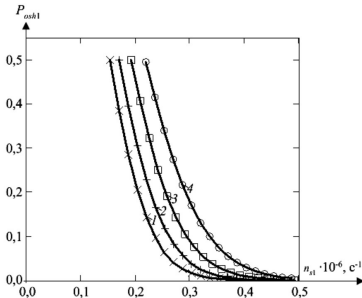


Рис. 1 – Зависимость вероятности ошибочной регистрации символа «1» от средней скорости счета сигнальных импульсов n_{s1} при средней длительности мертвого времени:
1 – $\tau_d = 0$, 2 – $\tau_d = 5$ мкс, 3 – $\tau_d = 10$ мкс,
4 – $\tau_d = 15$ мкс

Расчет проводился для одинаковых значений $N_1 = 1$, $N_2 = 7$, $n_t = 10^3 \text{ с}^{-1}$ и $\tau_b = 100$ мкс по методикам [3, 4]. Отметим, что при других значениях N_1 и N_2 , и отношениях n_t/n_{s1} проявление эффекта мертвого времени продлевающегося типа для рассматриваемого канала связи аналогично представленному на рис. 1.

Диапазон значений n_{s1} определялся по методикам [6, 7] с учетом того, что вероятность ошибочной регистрации символов «1» P_{osh1} должна быть менее 0,5.

Из представленных результатов видно, что при прочих равных параметрах рост средней длительности мертвого времени продлевающегося типа приводит к увеличению средних скоростей счета сигнальных импульсов n_{s1} , при которых достигаются наименьшие значения $P_{osh1} = 0,00$: при $n_{s1} = 35,0 \times 10^4 \text{ с}^{-1}$ для $\tau_d = 0$; при $n_{s1} = 38,9 \times 10^4 \text{ с}^{-1}$ для $\tau_d = 5$ мкс; при $n_{s1} = 43,7 \times 10^4 \text{ с}^{-1}$ для $\tau_d = 10$ мкс; при $n_{s1} = 50,0 \times 10^4 \text{ с}^{-1}$ для $\tau_d = 15$ мкс. Причем с ростом τ_d вероятность ошибочной регистрации символов «1» P_{osh1} увеличивается. Так, например, при $n_{s1} = 31,0 \times 10^4 \text{ с}^{-1}$ вероятность ошибочной регистрации символов «1» равна $1,31 \times 10^{-2}$ для $\tau_d = 0$; $3,17 \times 10^{-2}$ для $\tau_d = 5$ мкс; $7,19 \times 10^{-2}$ для $\tau_d = 10$ мкс; $15,08 \times 10^{-2}$ для $\tau_d = 15$ мкс.

Такое влияние τ_d и скорости счета n_{s1} на P_{osh1} объясняется смещением максимумов статистических распределений смеси числа темновых и сигнальных импульсов при передаче символов «1» $P_{st1}(N)$, полученных для рассматриваемого канала связи в работах [6, 7]. При наименьших величинах n_{s1} для исследуемых диапазонов значений скоростей счета сигнальных импульсов этот максимум близок к 0, поэтому вероятность P_{osh1} весьма высока, что следует из формулы (1) и иллюстрируется рис. 1. Однако с увеличением n_{s1} вероятность P_{osh1} уменьшается за счет сдвига максимумов $P_{st1}(N)$ в сторону больших значений N . Это приводит к спаду зависимостей $P_{osh1}(n_{s1})$ вплоть до наименьших значений (см. рис. 1). Поскольку, согласно [6, 7], в исследуемых диапазонах n_{s1} увеличение τ_d при прочих равных параметрах смещает максимумы $P_{st1}(N)$ в сторону 0, P_{osh1} растет с увеличением τ_d (см. рис. 1).

ЗАКЛЮЧЕНИЕ

Таким образом, в данной работе получено выражение для оценки вероятности ошибочной регистрации символов «1», передаваемых по квантово-криптографическому каналу связи, в котором в качестве приемного модуля использован счетчик фотонов с мертвым временем продлевающегося типа.

Выполненные исследования показали, что при прочих равных параметрах рост средней длительности мертвого времени продлевающегося типа приводит к увеличению средних скоростей счета сигнальных импульсов, при которых достигаются наименьшие значения вероятностей ошибочной регистрации символов «1».

СПИСОК ЛИТЕРАТУРЫ

1. Килин, С.Я. Квантовая криптография: идеи и практика / С.Я. Килин. – Мн.: Бел. наука, 2007 – 391 с.
2. Гулаков, И.Р. Фотоприемники квантовых систем: монография / И.Р. Гулаков, А.О. Зеневич. – Минск: УО ВГКС, 2012. – 276 с.
3. Тимофеев, А.М. Методика повышения достоверности принятых данных счетчика фотонов на основе анализа скорости счета импульсов при передаче двоичных символов «0» / А.М. Тимофеев // Приборы и методы измерений. – 2019. – т. 10. – № 1. – С. 80–89.
4. Тимофеев, А.М. Достоверность принятой информации при ее регистрации в однофотонном канале связи при помощи счетчика фотонов / А.М. Тимофеев // Информатика. – 2019. – т. 16. – № 2. – С. 90–98.
5. Дмитриев, С.А. Волоконно-оптическая техника: современное состояние и перспективы / С.А. Дмитриев, Н.Н. Слепов. – М.: «Волоконно-оптическая техника», 2005. – 576 с.
6. Тимофеев, А.М. Оценка влияния продлевающегося мертвого времени счетчика фотонов на вероятность ошибочной регистрации данных квантово-криптографических каналов связи / А.М. Тимофеев // Вестник связи. – 2018. – № 1(147). – С. 56–62.
7. Тимофеев, А.М. Влияние времени однофотонной передачи информации на вероятность ошибочной регистрации данных асинхронных квантово-криптографических каналов связи / А.М. Тимофеев // Вестник ТГТУ. – 2019. – т. 25. – № 1. – С. 36–46.

СОЗДАНИЕ ПРОГРАММНЫХ СРЕДСТВ ДЛЯ ПРОЕКТИРОВАНИЯ И УПРАВЛЕНИЯ РАЗРАБОТКОЙ ПРОГРАММНЫХ ПРОДУКТОВ

Ульянко В. Г., Сечко Г. В.

Кафедра электронных вычислительных машин, кафедра защиты информации, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: viktor.uljanko@gmail.com, georg.sechko@gmail.com

Решается задача создания программных средств для проектирования и управления разработкой программных продуктов. Проводится обзор и анализ существующих аналогов программных средств для проектирования и управления разработкой программных продуктов и их недостатков. Обосновывается необходимость создания нового программного средства, свободного от названных недостатков.

ВВЕДЕНИЕ

Каждый человек постоянно осуществляет проекты в своей повседневной жизни, это может быть ремонт в квартире, написание книги, в конце концов написание данного дипломного проекта. Все эти виды деятельности имеют между собой целый ряд общих признаков, делающих их проектами: они направлены на достижение конкретных целей; они включают в себя координированное выполнение взаимосвязанных действий; они имеют ограниченную протяженность во времени, с определенным началом и концом; все они в определенной степени неповторимы и уникальны.

УПРАВЛЕНИЕ РАЗРАБОТКОЙ ПРОЕКТОВ

Проектом называется совокупность распределенных во времени мероприятий или работ, направленных на достижение поставленной цели [1]. Стоит упомянуть, что время один из ключевых ресурсов на проектах и, согласно исследования, погрешность в оценке сроков выполнения проектов составляет от 20% до 120% [2]. Примерами проектов являются строительство зданий, комплексов, предприятий, освоение выпуска нового вида продукции, проведение модернизации производства, разработка программного продукта и т. д.

Результат проекта – это некоторая продукция или полезный эффект, создаваемые в ходе реализации проекта. В качестве результата, в зависимости от цели проекта, могут выступать: научная разработка, новый технологический процесс, программное средство, строительный объект, реализованная учебная программа, реструктурированная компания, сертифицированная система качества и т. д. Об успешности проекта судят по тому, насколько его результат соответствует по своим затратным, доходным, инновационным, качественным, временным, социальным, экологическим и другим характеристикам запланированному уровню.

Управление проектами – это приложение знаний, опыта, методов и средств к работам про-

екта для удовлетворения требований, предъявляемых к проекту, и ожиданий участников проекта. Чтобы удовлетворить эти требования и ожидания необходимо найти оптимальное сочетание между целями, сроками, затратами, качеством и другими характеристиками проекта.

Прежде всего, у проекта обязательно имеются одна или несколько целей [1]. Под целями понимается не только конечные итоги проекта, но и выбранные пути достижения этих результатов. Достижение целей проекта может быть реализовано различными способами. Для сравнения этих способов необходимы критерии успешности достижения поставленных целей. Обычно, в число основных критериев оценки различных вариантов исполнения проекта входят сроки и стоимость достижения результатов. При этом запланированные цели и качество обычно служат основными ограничениями при рассмотрении и оценке различных вариантов. Конечно, возможно использование и других критериев и ограничений – в частности, ресурсных. Для управления проектами необходимы рычаги. Влиять на пути достижения результатов проекта, цели, качество, сроки и стоимость исполнения работ можно, выбирая применяемые технологии, состав, характеристики и назначения ресурсов на выполнение тех или иных работ. Таким образом, применяемые технологии и ресурсы проекта можно отнести к основным рычагам управления проектами. Кроме этих основных существуют и вспомогательные средства, предназначенные для управления основными. К таким вспомогательным рычагам управления можно отнести, например, контракты, которые позволяют привлечь нужные ресурсы в нужные сроки. Кроме того, для управления ресурсами необходимо обеспечить эффективную организацию работ. Это касается структуры управления проектом, организации информационного взаимодействия участников проекта, управления персоналом.

Для целенаправленного выполнения проекта работниками должен быть выполнен ряд работ, различных как по своему назначению, так

и по квалификационным требованиям, предъявляемым к разработчикам [1]. Иными словами, в ходе развития проекта командой разработчиков выполняются те или иные функции. Распределение функций между исполнителями является ничем иным как распределением ролей в команде, выполняющей проект.

Понятно, что значимость ролей разработчиков и тех, кто с ними связан, различается в зависимости от того, какой проект выполняет программистская команда. Тем не менее, можно указать на ряд наиболее типичных ролей, игнорирование которых приводит в лучшем случае к потерям производительности труда команды в целом, а в худшем – к неудаче развития проекта. В качестве достаточно полного перечня ролей можно указать на следующий список, предлагаемый в рамках подхода Центра объектно-ориентированной технологии фирмы IBM:

- Заказчик (Customer) – реально существующий (в организации, которой подчинена команда, или вне ее) инициатор разработки или кто-либо иной, уполномоченный принимать результаты (как текущие, так и окончательные) разработки;
- Менеджер проекта (Project Manager) – отвечает за развитие проекта в целом, гарантирует, что распределение заданий и ресурсов позволяет выполнить проект, что работы и предъявление результатов идут по графику, что результаты соответствуют требованиям. В рамках этих функций менеджер проекта взаимодействует с заказчиком и планировщиком ресурсов;
- Руководитель команды (Team Leader) – производит техническое руководство командой в процессе выполнения проекта. Для больших проектов возможно привлечение нескольких руководителей подкоманд, отвечающих за решение частных задач;
- Архитектор (Architect) – отвечает за проектирование архитектуры системы, согласовывает развитие работ, связанных с проектом;
- Проектировщик подсистемы (Designer) – отвечает за проектирование подсистемы или категории классов, определяет реализацию и интерфейсы с другими подсистемами;
- Разработчик (Developer) – реализует проектируемые компоненты, владеет и создает специфичные классы и методы, осуществляет кодирование и автономное тестирование, строит продукт. Это широкий термин, который может подразделяться на более узкие роли (например, разработчик классов). В зависимости от сложности проекта

команда может включать различное число разработчиков;

- Разработчик информационной поддержки (Information Developer) – создает документацию, сопровождающую продукт, когда выпускается версия. Включаемые в нее инсталляционные материалы, равно как ссылочные и учебные, а также материалы помощи предоставляются на бумажных и машинных носителях. Для сложных проектов возможно распределение этих задач между несколькими разработчиками информационной поддержки;
- Специалист по пользовательскому интерфейсу (Human Factors Engineer) – отвечает за удобство применения системы. Работает с заказчиком, чтобы удостовериться, что пользовательский интерфейс удовлетворяет требованиям;
- Тестировщик (Tester) – проверяет функциональность, качество и эффективность продукта. Строит и исполняет тесты для каждой фазы развития проекта;

Любой проект в процессе своей реализации проходит различные стадии, называемые в совокупности жизненным циклом проекта. Для реализации различных функций управления проектом необходимы действия, которые называются процессами управления проектами.

Процессы управления проектами могут быть разбиты на шесть основных групп, реализующих различные функции управления: процессы инициации – принятие решения о начале выполнения проекта; процессы планирования – определение целей и критериев успеха проекта и разработка рабочих схем их достижения; процессы исполнения – координация людей и других ресурсов для выполнения плана; процессы анализа – определение соответствия плана и исполнения проекта поставленным целям и критериям успеха и принятие решений о необходимости применения корректирующих воздействий; процессы управления – определение необходимых корректирующих воздействий, их согласование, утверждение и применение; процессы завершения – формализация выполнения проекта и подведение его к упорядоченному финалу.

Под эффективным управлением проектами понимается интеграция информационных систем планирования с управленческими процедурами и организационной структурой.

СПИСОК ЛИТЕРАТУРЫ

1. Что такое проект? - презентация онлайн - ppt Онлайн [Электронный ресурс]. Режим доступа: Что такое проект? - презентация онлайн - ppt Онлайн. Дата доступа: 23.09.2019.
2. Оценка временных затрат на разработку с использованием метрик программного обеспечения / Ю. В. Лях // Доклады БГУИР. – 2017. – № 4 (106). – Сс. 83-87с.

СЕРВОПРИВОД С КОМБИНИРОВАННЫМ УПРАВЛЕНИЕМ

Хаджинов М. К., Доманов А. Т., Павлова А. В.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: km_h@tut.by, pavlova@bsuir.by

Рассмотрен сервопривод подчинённого регулирования с тремя входами для комбинированного управления координатой, скоростью и ускорением. Осуществлен синтез задатчика комбинированного управления, в который введено упреждение в сигналы управления для частичной компенсации инерционности сервопривода. Приведены варианты настроек задатчика и результаты моделирования работы сервопривода с задатчиком

ВВЕДЕНИЕ

Современные автоматические системы высокой динамической точности обычно строят на основе принципа комбинированного управления, сочетающего в себе принципы управления по отклонению и по возмущению. Точность работы комбинированных систем выше точности систем, использующих только один из принципов управления, причем недостатки обоих принципов при их объединении устраняются. Высокоточный сервопривод, в том числе с контуром оптимизации по быстродействию [1], обычно имеет структуру подчинённого регулирования с отдельными регуляторами тока, скорости и координаты. Очевидно, что координированное управление сразу тремя сигналами на вход всех регуляторов эффективнее управления только одной - координатой. Для реализации комбинированного управления скоростью, ускорением и координатой необходимо создать устройство или алгоритм для вычисления скорости и ускорения из управляющего сигнала координатой – задатчик комбинированного управления. Ставится задача разработки алгоритма задатчика, пригодного для практического использования. Естественное на первый взгляд решение задачи с использованием в алгоритме дифференцирования и двойного дифференцирования неприемлемо из-за наличия шумов в сигнале управления координатой. Дополнительной задачей, возлагаемой на задатчик, является введение некоторого упреждения в сигналы управления как ускорением и скоростью, так и координатой для частичной компенсации инерционности сервопривода.

I. СТРУКТУРА СЕРВОПРИВОДА С ЗАДАТЧИКОМ

Предлагается формировать структуру задатчика в виде следящего оценивающего фильтра с модальным регулятором. Простейший следящий фильтр второго порядка можно составить из последовательно включённых двух интеграторов для оценивания скорости и координаты и модального регулятора, обеспечивающе-

го отслеживание управляющего сигнала координаты и желаемую динамику оценивания. Выходы интеграторов дадут фильтрованные оценки скорости и координаты, а входной сигнал первого интегратора даст нефильТРованную оценку ускорения. Таким образом будет сформировано три сигнала для координированного управления тремя регуляторами сервопривода. Настройкой модальным регулятором динамики следящего фильтра можно достичь компромисса между улучшением динамики сервопривода за счёт комбинированного управления, дополнительной инерционностью задатчика и его фильтрующими свойствами. Можно улучшить динамические характеристики задатчика, если заставить вырабатывать выходные управляющие сигналы ускорения, скорости и координаты с упреждением входного сигнала управления координатой. Для достижения этого в структуру следящего фильтра добавляем инерционное звено после второго интегратора. Таким образом ошибка следящего фильтра будет формироваться как разница между сигналом управляющей координаты и задержанным инерционным звеном сигналом оценки координаты самого фильтра. Порядок фильтра увеличивается до трёх. Понятно, что модальный регулятор для задатчика с упреждением можно настроить на динамику, близкую к динамике задатчика второго порядка без упреждения.

II. НАСТРОЙКИ ЗАДАТЧИКА СЕРВОПРИВОДА

За базу для настройки модального регулятора задатчика принята частота среза контура тока сервопривода, которая выбрана 1000 1/с. Три регулятора сервопривода были настроены как единый модальный регулятор на полином Латропа и обеспечивают переходную характеристику с перерегулированием 1.55 %. Использовалась также настройка регуляторов сервопривода на «асимптотический» эмпирический полином с перерегулированием 0.016 %. Частота масштабирования полиномов настройки модальных регуляторов задатчиков варьировалась от равной частоте среза контура тока сервопривода до

утроенной и удевятирѐнной. Частота упреждения (излома ЛАХ аperiodического звена в задатчике) также варьировалась от равной частоте среза контура тока сервопривода до меньшей в два и три раза. Задатчики управления задавались уравнениями в пространстве состояний. Трансформация модели задатчика с утроенной частотой масштабирования и тройным упреждением в передаточные функции имеет вид (по очереди: задатчик угла, задатчик скорости, задатчик ускорения):

$$W(S) = \frac{62100(s + 333.3)(s + 1304)}{(s + 2040)(s^2 + 3660s + 1.323e07)}$$

$$W(S) = \frac{8.1e07s(s + 333.3)}{(s + 2040)(s^2 + 3660s + 1.323e07)}$$

$$W(S) = \frac{8.1e07s^2(s + 333.3)}{(s + 2040)(s^2 + 3660s + 1.323e07)}$$

Комбинированное управление сервоприводом от задатчика управления эффективно работает в режиме слежения. В переходных режимах ему свойственно большое перерегулирование. Была проделана работа по снижению перерегулирования переходных процессов сервопривода с задатчиком дозированием (уменьшением коэффициентов) комбинированного управления. Результаты моделирования сервопривода с задатчиком показали меньшую эффективность медленного задатчика с частотой масштабирования, равной частоте среза контура тока сервопривода. Перерегулирование составляло 55 % без упреждения и 80 % с упреждением. С ускорением задатчика в 3 и 9 раз перерегулирование уменьшалось до 43 и 37 % без упреждения и до 60 и 52 % с упреждением. Представляются интересными результаты дозированного комбинированного управления с коэффициентами $k_s = 0.4$ по скорости и $k_u = 0.4$ по ускорению с задатчиком без упреждения с утроенной частотой масштабирования. Длительность переходной характеристики с нулевым перерегулированием уменьшилась до 4.3 мс по сравнению с 7.4 мс у сервопривода без задатчика. Для быстрого задатчика, при увеличении частоты масштабирования в 9 раз и частотой упреждения, равной частоте среза контура тока сервопривода, дозированное комбинированное управление с коэффициентами $k_s = 0.5$ по скорости и $k_u = 0.7$ по ускорению получилась

переходная характеристика с перерегулированием 0.75 % и длительностью 1.8 мс. Т.е. задатчик ускорил переходный процесс более чем в 4 раза. Переходная характеристика сервопривода с задатчиком и без него изображена на Рис.1. Для среднего задатчика, но с утроенным упреждением дозированное комбинированное управление с коэффициентами $k_s = 0.1$ по скорости и $k_u = 0.4$ по ускорению длительность переходной характеристики сокращает до 1 мс.

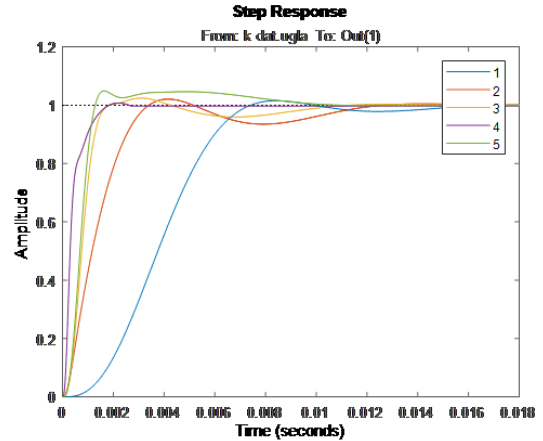


Рис. 1 – Переходные характеристики сервопривода: 1-без задатчика, 2-с задатчиком с девятикратной частотой масштабирования без упреждения, 3 – задатчик с утроенной частотой масштабирования и удвоенным упреждением, 4-задатчик с девятикратной частотой масштабирования и одиночным упреждением, 5- задатчик с утроенной частотой масштабирования и утроенным упреждением

III. ЗАКЛЮЧЕНИЕ

Предложенная структура задатчика в виде следящего фильтра эффективно работает при настройке его динамики быстрее контура тока электропривода. Рекомендуется величину упреждения задатчика устанавливать больше эквивалентной инерционности контура тока. Дозированное комбинированное управление позволяет устранить перерегулирование и ускорить переходную характеристику в 4 и более раз.

СПИСОК ЛИТЕРАТУРЫ

1. Хаджинов, М.К. Сервоприводы с квадратичной обратной связью по скорости / М.К. Хаджинов, А.Т.Доманов, А.В.Павлова // Информационные технологии и системы 2017 (ИТС 2017): материалы международной научной конференции, БГУ-ИР, Минск, Беларусь, 25 октября 2017 - С. 72-73. <https://libeldoc.bsuir.by/handle/123456789/20570/>

ПОСТРОЕНИЕ СОКРАЩЕННОГО ГРАФА ДОСТИЖИМЫХ СОСТОЯНИЙ ПАРАЛЛЕЛЬНОГО АЛГОРИТМА

Черемисинова Л. Д., Черемисинов Д. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: cldr@newman.bas-net.by, cher@newman.bas-net.by

Рассматривается задача верификации систем управления с параллелизмом поведения на основе графа достижимых состояний. Показано, как можно существенно сократить сложность такого графа для систем рассматриваемого класса, и предложен метод построения сокращенного графа достижимых состояний.

ВВЕДЕНИЕ

Рост сложности современных микроэлектронных систем, повышение требований к надежности их проектирования и реализации обусловили тот факт, что неотъемлемой частью процесса проектирования стало тестирование, в частности проверка соответствия поведения системы требованиям, предъявляемым спецификацией на ее проектирование. Одной из широко используемых технологий для решения этой проблемы является тестирование на основе моделей [1], которое предполагает моделирование реальной системы на тестовой последовательности, генерируемой на основе модели (спецификации), описывающей желаемое поведение системы.

В настоящее время наиболее разработанным направлением в области тестирования является верификация на основе моделей, представляемых конечными автоматами. Проблема тестирования систем, которым присущ параллелизм происходящих в них процессов, менее изучена. Обычно рассматриваются системы, состоящие из параллельно работающих компонентов, которые моделируются сетью конечных автоматов. Известны также подходы к тестированию систем с использованием моделей «истинного параллелизма», таких как сети Петри, основанные на использовании графов достижимости.

В настоящей работе рассматриваются системы управления с параллелизмом поведения, спецификация на проектирование которых задается на языке параллельных автоматов [2] (подкласс раскрашенных сетей Петри). Основным инструментом, лежащим в основе методов анализа поведенческих свойств систем рассматриваемого типа, является граф достижимых состояний параллельного автомата, в процессе обхода которого генерируется тестовая последовательность. Основным недостатком такого подхода является то, что размер графа достижимости растет экспоненциально с ростом числа состояний и переходов параллельных автоматов.

В работе показано, как можно существенно сократить сложность такого графа и предложен

метод построения сокращенного графа достижимых состояний.

I. ПАРАЛЛЕЛЬНЫЙ АВТОМАТ И ГРАФ ДОСТИЖИМЫХ СОСТОЯНИЙ

В отличие от классического конечного автомата в параллельном автомате рассматриваются так называемые частичные состояния. В любой момент времени параллельный автомат может одновременно находиться в нескольких таких состояниях. Параллельный автомат можно рассматривать как динамическую дискретную систему, которая допускает параллельные процессы, соответствующие взаимно параллельным частичным состояниям. Множество таких состояний в момент времени определяет полное (глобальное) состояние S_t , которое можно интерпретировать как маркировку сети Петри.

Алгоритм поведения систем управления на языке параллельных автоматов представляется совокупностью переходов вида: $\tau_i = (\gamma_i^1, k_i^1) \rightarrow (\gamma_i^2, k_i^2)$, где γ_i^1 и γ_i^2 – подмножества частичных состояний, в которых автомат находится перед и после срабатывания i -го перехода; k_i^1 и k_i^2 – конъюнкции входных и выходных булевых переменных, трактуемые как условие перехода и выходные сигналы, сопровождающие переход. Переход срабатывает, когда текущее полное состояние S_t включает все состояния из γ_i^1 , а переменные автомата принимают значения, обращающие k_i^1 в единицу. После срабатывания перехода переменным из k_i^2 присваиваются значения, обращающие k_i^2 в единицу, а состояние S_t заменяется на $S_{t+1} = (S_t \setminus \gamma_i^1) \cup \gamma_i^2$.

Таким образом, динамика параллельных автоматов описывается в пространстве достижимых полных состояний и значений логических сигналов. Чтобы вычислить следующее полное состояние S_{t+1} автомата, необходимо определить множество T переходов τ_i , которые запускаются в состоянии S_t и состояние K_{t+1} на множестве переменных автомата. Следующие состояния задаются таким образом:

1) $S_{t+1} = (S_t \setminus \gamma^1) \cup \gamma^2$, где $\gamma^1 = \cup_i \gamma_i^1$ и $\gamma^2 = \cup_i \gamma_i^2$ – объединения множеств частичных

состояний всех параллельно выполняемых переходов τ_i из T ;

2) состояние K_{t+1} на множестве переменных получается из K_t таким изменением значений переменных, входящих в конъюнкцию k_i^2 всех переходов τ_i , чтобы все $k_i^2 = 1$.

Основной подход к тестированию системы управления состоит в исследовании пространства всех возможных состояний его модели и всех переходов, которые система может совершать между этими состояниями. Граф достижимых состояний задает динамику системы управления при всевозможных изменениях сигналов на его входных полюсах.

Граф достижимости является ориентированным мультиграфом, вершинам которого соответствуют все возможные полные состояния S_t автомата, а дугам – переходы. Дуга графа помечается символом перехода τ_i и связывает вершины S_t и S_p графа, если результат срабатывания τ_i меняет состояние S_t автомата на S_p . Граф достижимости легко получается путем вычисления всех полных состояний S_t , начиная с начальной.

II. СОКРАЩЕНИЕ ЧИСЛА ВЕРШИН В ГРАФЕ ДОСТИЖИМОСТИ

Граф достижимости содержит все достижимые состояния параллельного автомата, получаемые путем всевозможных частичных упорядочений параллельных переходов. Его размер растет экспоненциально с увеличением числа частичных состояний и степени параллелизма переходов. Большое пространство состояний негативно влияет на время тестирования. Иногда система слишком велика, чтобы построить график достижимости и получить даже какую-нибудь тестовую последовательность путем обхода его вершин.

Эффективным подходом к сокращению графа достижимости является доопределение частичного порядка на множестве параллельных переходов, в основу которого положена коммутативность асинхронно протекающих процессов. Различные методы доопределения частичного порядка используются [3] при формальной верификации сетей Петри. Эти методы, однако, не учитывают информационное взаимодействие переходов, что недопустимо при решении задачи генерации тестовых наборов.

Доопределение частичного порядка на множестве параллельных переходов основано на том факте, что некоторые параллельно выполняемые переходы не зависят друг от друга, в том смысле, что порядок выполнения одних переходов не влияет на условия срабатывания других. Соответственно, можно рассматривать не все возможные порядки выполнения таких переходов, а упорядочить их произвольным образом, в том числе и совместив их выполнение. Получаемый в результате граф достижимых состояний эквивалентен

исходному по степени соответствия спецификации системы, но содержит значительно меньше вершин и дуг.

III. ПОДХОД К СОКРАЩЕНИЮ ГРАФА ДОСТИЖИМОСТИ

Переход $\tau_i = (\gamma_i^1, k_i^1) \rightarrow (\gamma_i^2, k_i^2)$ срабатывает в состоянии S_t параллельного автомата, если $\gamma_i^1 \subseteq S_t$ и $k_i^1 \wedge K_t \neq 0$.

Пара переходов $\tau_i = (\gamma_i^1, k_i^1) \rightarrow (\gamma_i^2, k_i^2)$ и $\tau_j = (\gamma_j^1, k_j^1) \rightarrow (\gamma_j^2, k_j^2)$, срабатывающих в состоянии S_t параллельного автомата, являются совместимыми (коммутативными) в том смысле, что они могут быть выполнены в любом порядке, приводя к одному и тому же полному состоянию, если при любых значениях входных сигналов автомата выполняются следующие условия:

1) условия срабатывания переходов совместимы, т.е. $k_i^1 \wedge k_j^1 \neq 0$;

2) изменение значений переменных в результате срабатывания одного перехода не приводит к нарушению условия срабатывания другого, т.е. $k_j^1 \wedge k_i^2 \neq 0$ и $k_i^1 \wedge k_j^2 \neq 0$;

3) изменения значений выходных сигналов, вызываемые переходами, совместимы, т.е. $k_i^2 \wedge k_j^2 \neq 0$.

При любом порядке выполнения совместимых переходов полное состояние автомата заменяется на состояние $(S_t \setminus (\gamma_i^1 \cup \gamma_j^1)) \cup (\gamma_i^2 \cup \gamma_j^2)$. Определение совместимости переходов может быть обобщено и на случай более двух переходов: переходы, возможные в некотором состоянии, могут быть выполнены одновременно, если они попарно совместимы.

Предлагаемый метод сокращения графа достижимости основан на просмотре его вершин (начиная с начальной) со степенью исхода, большей 1. Среди переходов из такой вершины выделяются подмножества попарно совместимых, соответствующие переходам дуги, а также достижимые из них вершины графа, объединяются.

IV. ЗАКЛЮЧЕНИЕ

Предлагаемый подход к сокращению графа достижимости лег в основу метода построения сокращенного графа достижимости непосредственно по описанию алгоритма на языке параллельных автоматов.

V. СПИСОК ЛИТЕРАТУРЫ

1. Tretmans, J. Model based testing with labelled transition systems / J. Tretmans // Formal Methods and Testing: Lecture Notes in Computer Science. – 2008. – Vol. 4949. – P. 1–38.
2. Закревский, А.Д. Параллельные алгоритмы логического управления / А.Д. Закревский // Минск: Ин-т техн. кибернетики НАН Беларуси, 1999. – 202 с.
3. Lluch-Lafuente, A. Partial Order Reduction in Directed Model Checking / A. Lluch-Lafuente // Proc. of the 9th Intern. SPIN Workshop on Model Checking of Software, Heidelberg, April 11–13, 2002. – P. 112–127.

Секция «Интеллектуальные
информационные технологии»

Section «Intellectual information
technologies»

СИСТЕМА АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ ЭМОЦИЙ ПО РЕЧЕВОМУ СИГНАЛУ

Брановицкий А. А., Борисов Д. В.

Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: {art7yom, divlboris}@gmail.com

В статье описаны основные области применения и особенности создания систем распознавания эмоций по речевому сигналу. Приведён аналитический обзор актуальных методов для выделения характерных особенностей речевого сигнала. Проиллюстрировано влияние выбора модели нейронной сети на точность распознавания эмоционального состояния.

ВВЕДЕНИЕ

На современном этапе развития общества происходит интенсивное развитие информационных технологий, которые вносят качественные улучшения во все сферы человеческой жизни. Одной из перспективных задач в данной области является повышение качества взаимодействия между субъектами в системах взаимодействия человек-компьютер, человек-человек.

Поскольку идентификация эмоционального состояния человека играет важную роль для эффективной коммуникации с ним, задача автоматического распознавания эмоций человека на основе характеристик речевого сигнала представляет интерес не только в теоретическом плане, но и для решения различных прикладных задач.

Распознавание эмоций человека по речи может найти применение в системах взаимодействия человек-компьютер (эмоциональное окрашивание речи операторов автоматизированных колл-центров; улучшение коммуникации людей с голосовыми помощниками, системами виртуальной реальности) или человек-человек (автоматические переводчики, передающие эмоции, заложенные в переводимую речь; детекторы лжи; диагностика психических расстройств на основе изменения эмоционального фона за период времени; мониторинг настроения толпы).

I. СТРУКТУРА СИСТЕМЫ РАСПОЗНАВАНИЯ ЭМОЦИЙ ПО РЕЧЕВОМУ СИГНАЛУ

Как и любая типичная система распознавания, система распознавания эмоций на основе речи соотносит исходные данные на входе – речевой сигнал, к определённому классу на выходе – виду эмоции, с помощью выделения существенных признаков – речевых особенностей, характеризующих эти данные.

На рисунке 1 представлена схема работы стандартной системы распознавания эмоций по речевому сигналу. В работе системы можно выделить четыре основных этапа:

1. предварительная обработка сигнала;
2. выделение характерных особенностей речевого сигнала;

3. предобработка особенностей речевого сигнала перед классификацией;
4. классификация.

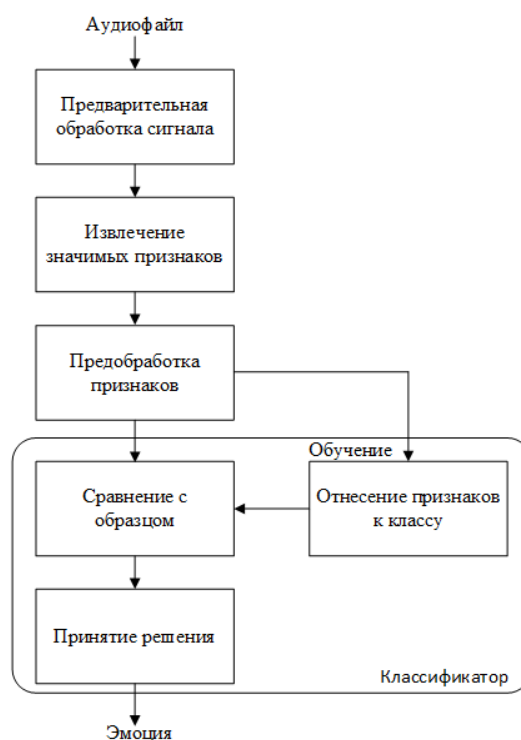


Рис. 1 – Схема работы системы распознавания эмоций

Вход системы – аудиофайлы, которые подвергаются предварительной обработке. Следующим шагом является извлечение значимых признаков из речевого сигнала. Затем применяются алгоритмы фильтрации, отсеивающие малозначимые признаки. Далее вектор признаков поступает на вход классификатору. Выход системы – вид эмоции.

II. МЕТОДЫ, ПРИМЕНЯЕМЫЕ ДЛЯ ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКИ СИГНАЛА

Предварительная обработка сигнала является важным этапом системы, так как оказывает огромное влияние на производительность классификатора. Очень важно подавать на вход нейронной сети данные, отражающие ключевые

особенности образца для классификации. Предварительная обработка охватывает аналогово-цифровое преобразование (АЦП) входного сигнала для получения его спектральных составляющих; цифровую фильтрацию сигнала, нормализацию, сегментацию сигнала, удаление неречевых компонентов сигнала [1].

Оцифрованные образцы речи сначала нормализуются. Далее образцы сегментируются на кадры продолжительностью 30 мс с перекрытием 10 мс с использованием окна Хэмминга. Наконец, удаляются кадры с тишиной и невокализованные кадры.

III. ХАРАКТЕРНЫЕ ОСОБЕННОСТИ РЕЧЕВОГО СИГНАЛА И МЕТОДЫ ИХ РАСЧЁТА

Для проведения успешной классификации чрезвычайно важно извлечь значимые признаки из речевого сигнала. Наиболее эффективные особенности речевого сигнала для классификации на данный момент времени сводятся к следующему списку:

- Частота основного тона (англ. pitch) – это частота колебания голосовых связок при произнесении тоновых звуков.
- Форманты (англ. formants) – группа усиленных обертонов, формирующих специфический тембр голоса. Рассчитываются при помощи метода линейного прогнозирующего кодирования (англ. Linear Predictive Coding, LPC).
- Мел-кепстральные коэффициенты (англ. Mel-frequency Cepstral Coefficients, MFCCs). Мел – единица высоты звука, основанная на восприятии этого звука нашими органами слуха [2].
- Скорость пересечения нуля (англ. zero crossing rates, ZCR) – это скорость изменения знака вдоль сигнала, то есть скорость, с которой значение сигнала изменяется с положительного на отрицательное или с отрицательного на положительное.
- Джиттер и шиммер (англ. jitter and shimmer). Джиттер – мера пертурбации (возмущений) частоты основного тона, показывающая произвольные изменения в частоте смежных вибрационных циклов голосовых складок. Шиммер – мера аналогичная джиттеру, только характеризующая пертурбации амплитуд сигнала на смежных циклах колебаний основного тона.
- Энергия сигнала (англ. energy). Рассчитывать необходимо краткосрочный энергетический контур, так как он напрямую связан с уровнем эмоционального возбуждения.
- Длительность и вокализация (англ. duration and voicing). Вокализация – отношение количества вокализованных к количеству невокализованных кадров.

IV. ПРЕДОБРАБОТКА ПРИЗНАКОВ РЕЧЕВОГО СИГНАЛА

Векторы признаков, полученные на предыдущем этапе, перед подачей их на вход выбранному классификатору нормализуются; содержащие более 2–10% нулевых значений, отбрасываются. Так как на большинство классификаторов негативно влияет избыточность, то для уменьшения размерности итогового вектора признаков используется алгоритм выбора признаков (прямой выбор признаков, генетический алгоритм, последовательный прямой плавающий поиск).

V. КЛАССИФИКАЦИЯ

Классификация – заключительный этап работы системы распознавания эмоций. Точность классификации в значительной мере зависит от выбранного типа классификатора. Ранее в качестве классификатора для систем распознавания эмоций применялись Скрытые Марковские модели (англ. НММ), метод k-ближайших соседей (англ. k-NN), Модель гауссовых смесей (англ. GMM), метод опорных векторов (англ. SVM), но в данный момент отдаётся предпочтение нейронным сетям различных архитектур [3].

VI. ОЦЕНКА ЭФФЕКТИВНОСТИ СИСТЕМЫ

При практической реализации системы для сокращения размерности вектора признаков применялся алгоритм прямого выбора признаков (англ. Forward Feature Selection, FFS), в качестве классификаторов выступали нейронные сети. Таблица 1 демонстрирует влияние типа и архитектуры нейронной сети на итоговую эффективность разрабатываемой системы.

Таблица 1 – Успешность классификации эмоций нейронными сетями

ANN	Ang	Bore	Fear	Happy	Sad	Avg
FFBP	74%	67%	77%	84%	69%	74%
FIT	71%	75%	62%	68%	70%	69%
CF	69%	72%	70%	70%	64%	69%
LVQ	58%	78%	70%	63%	80%	70%
PNN	96%	83%	76%	88%	79%	84%

В целом, выбор подходящей нейронной сети позволяет создать систему распознавания эмоций по речевому сигналу, обладающую довольно высокой точностью.

СПИСОК ЛИТЕРАТУРЫ

1. Rong, J. Acoustic Features Extraction for Emotion Recognition / J. Rong // 6th IEEE/ACIS International Conference. – 2007. – P. 419–424.
2. Patil, K. J. Emotion Detection From Speech Using Mfcc & Gmm / K. J. Patil, P. H. Zope, S. R. Suralkar // IJERT. – 2012. – Vol. 1, № 9. – P. 101–103.
3. Hamidi, M. Emotion Recognition From Persian Speech With Neural Network / M. Hamidi, M. Mansoorizade // IJAIA. – 2012. – Vol. 3, № 5. – P. 107.

РАЗРАБОТКА УНИВЕРСАЛЬНОГО ВЕБ-СКРАПЕРА НА БАЗЕ ФРЕЙМВОРКА NESTJS И БРАУЗЕРА CHROMIUM HEADLESS

Давлатов Ш. Р.

Кафедра защиты информации,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: shohrukh.92@gmail.com

В работе рассматриваются наиболее популярные методы и средства веб-скрапинга, которые широко применяются для автоматизации процесса извлечения данных из различных ресурсов сети Интернет. Предлагается новая архитектура универсального веб-скрапера на основе браузера Chromium Headless и фреймворка NestJS. Решается задача автоматического парсинга сайтов из заданного списка доменов для последующих процессов анализа и получения статистики используемых технологий, уязвимостей, географического распределения серверов и многое другое.

ВВЕДЕНИЕ

Веб-скрапинг это автоматическое считывание различных данных, выложенных на определенных страницах в Интернете и их последующая систематизация для того, чтобы разработать прикладные приложения на основе полученной информации. Таким образом, можно программно извлекать, обрабатывать, систематизировать и сохранять данные тысяч веб-страниц за считанные минуты. Например, инструменты веб-скрапинга позволяют автоматически получать информацию о товарах и ценах из онлайн-магазинов или собирать данные из новостных сайтов. Как правило, постановка задачи веб-скрапинга выглядит следующим образом: есть исходные данные, доступные только на определенных веб-ресурсах в сети, требуется программно получить информацию из заданного списка ресурсов, преобразовать в нужный формат и сохранить в локальном хранилище для последующего процесса обработки данных [1]. Рассмотрим основные области применения веб-скрапинга:

- Сбор и анализ данных для маркетинговых исследований рынка
- Извлечение контактных данных (электронные почты, номера телефонов и т.д.)
- Обнаружение потенциальных уязвимостей веб-ресурсов
- Определение технологий, на базе которых был разработан веб-ресурс.

I. ВЕБ-СКРАПИНГ

Рассмотрим простейший подход веб-скрапинга, который основан на статическом анализе HTML контента веб-страницы. Как правило, для подобного рода задач требуется незначительная компьютерная мощность и минимум времени. Данный способ работает только в том случае, если исходный HTML код веб-страницы содержит все необходимые данные. Примером таких приложений являются клиент-серверные решения на базе шаблона проекти-

рования MVC, где серверная часть отправляет браузеру HTML файл с готовыми данными. Первым шагом скрапинга является написание CSS селектора для получения необходимых блоков информации (это могут быть ячейки таблицы, заголовки меню и т.п.). Для реализации данного подхода, необходимо отправить HTTP GET запрос на URL-адрес страницы и получить обратно исходный HTML код. В экосистеме платформы NodeJS можно использовать инструмент CheerioJS для парсинга необработанного HTML кода и извлечение данных с помощью CSS селектора. Данная библиотека позволяет работать с данными, используя синтаксис, аналогичный jQuery. Отметим, что во многих случаях получение доступа к информации из необработанного HTML-кода является сложной задачей, ввиду того, что DOM дерево манипулируется сценариями JavaScript, выполняемые в фоновом режиме. Типичным примером является одностраничное веб-приложение, где HTML-документ содержит минимальный объем информации, а код JavaScript заполняет его во время выполнения. В этом случае, одним из возможных решений является создание DOM дерева и выполнение JavaScript сценариев, как это делают обычные браузеры. После данного процесса необходимая информация может быть извлечена из объекта DOM с помощью заданных CSS селекторов. Одним из возможных решений задачи автоматического создания DOM дерева является запуск сайта на Headless браузере, который обладает таким же набором функций что и обычный браузер, только без графического пользовательского интерфейса. Он работает в фоновом режиме, и управляется программно с помощью веб-драйвера и определенных программных сценариев. Для работы обычного браузера требуется гораздо больше вычислительной мощности, чем для отправки простого запроса GET и анализа ответа с помощью Headless браузера. Следовательно, использование подобных браузеров позволяет намного повысить быстродействие про-

грамм веб-скраперов. С другой стороны, этот метод является очень гибким и можно использовать его для навигации по страницам, имитации щелчков мыши, использования клавиатуры, заполнения форм, создания скриншотов страниц и выбора элементов для извлечения текстового содержимого. Headless Chromium является одним из самых популярных Headless браузеров, которая разрабатывается компанией Google [2]. Данная библиотека предоставляет API высокого уровня для программного управления Chromium в автономном режиме.

II. АРХИТЕКТУРА ПРИЛОЖЕНИЯ

Рассмотрим детально архитектуру разработанного приложения (рис. 1), которая основана на шаблоне проектирования «Цепочка обязанностей». Данный паттерн позволяет структурировать модули системы таким образом, чтобы запросы передавались последовательно по цепочке обработчиков. Каждый последующий модуль решает, может ли он обработать запрос сам и стоит ли передавать запрос дальше по цепи. Первое звено цепочки состоит из модуля для получения и первичной обработки списка исходных доменов. Для каждого элемента списка параллельно запускается процесс для отправки GET запроса и получения HTML ответа от соответствующих серверов. В качестве библиотеки для отправки асинхронных запросов была выбрана node-fetch, которая идентична по функциональности и совместимости методу fetch в обычных браузерах.



Рис. 1 – Архитектура приложения

После получения ответов от сервера, вся информация хранится в локальной базе данных MySQL для последующих процессов обработки и анализа [3]. Структура таблиц базы данных выглядит следующим образом: таблица доменов, категорий, заголовки HTTP ответов сервера, мета-теги страницы и HTML-контент. Для по-

вышения производительности результат запроса из базы данных разбивается на отдельные блоки фиксированного размера и для каждого блока параллельно запускаются последующие процессы. Далее, с помощью утилиты Wappalizer определяются версии технологий, на базе которых были разработаны веб-ресурсы [1]. На основе полученных версий ПО и общедоступных баз уязвимостей можно найти потенциально зараженные ресурсы в сети Интернет. С помощью whois сервисов определяется географическое расположение серверов по ip-адресу. Также, на базе публичных API для проверки безопасности веб-ресурсов проверяется каждый домен в исходном списке. Подобные сервисы есть у Google и Yandex (<https://developers.google.com/safe-browsing/v4> и <https://tech.yandex.com/safebrowsing> соответственно). Данные API позволяют проверять безопасность ресурса в сети по его URL. Разработчики браузеров, почтовых программ и клиентов обмена сообщениями могут использовать API для предупреждения пользователей, когда они пытаются перейти по опасной ссылке. Владельцы форумов, социальных сетей и сервисов сокращения ссылок могут воспользоваться автоматической проверкой каждого опубликованного URL-адреса. А системные администраторы корпоративных сетей могут применять централизованный подход к защите сотрудников от нежелательных веб-сайтов.

III. ЗАКЛЮЧЕНИЕ

В работе изучаются популярные инструменты веб-скрапинга, которые широко применяются для автоматического поиска и обработки необходимой информации из различных веб-ресурсов. Предложенное решение на базе технологий NestJS и Chromium Headless дает возможность достаточно гибко настроить систему под любые нужды пользователей, что позволяет проводить детальный анализ веб-ресурсов из определенной предметной области. Это достигается путем использования декларативного стиля создания модулей программы на основе декораторов фреймворка NestJS. Следующим шагом разработки станет создание онлайн SaaS платформы, основанная на разработках данной статьи.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Learning Web Scraping with JavaScript / M. Mekhatria. – Packt Publishing, 2018. – 320 с.
2. The Chromium Projects website [Electronic resource] / Google Inc. – Mode of access: <https://www.chromium.org/Home/>. – Date of access: 09.09.2019.
3. NestJS framework documentation [Electronic resource] / MIT by Kamil Mysliwiec, 2019. – Mode of access: <https://docs.nestjs.com>. – Date of access: 05.09.2019.

ПОДХОДЫ К ИНТЕЛЛЕКТУАЛИЗАЦИИ СРЕДСТВ АВТОМАТИЗАЦИИ ПРОЦЕССА РАЗРАБОТКИ КОМПЬЮТЕРНЫХ СИСТЕМ

Давыденко И. Т., Шункевич Д. В., Синельников П. М., Трунц В. В.

Кафедра интеллектуальных информационных технологий,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: davydenko@bsuir.by, shunkevichdv@gmail.com, pavelsinelnikovbe@gmail.com, trunts.vitalij@gmail.com

В работе рассмотрены современные средства автоматизации проектирования компьютерных систем, а также направления интеллектуализации таких средств.

ВВЕДЕНИЕ

На современном этапе развития технологий практически во всех производственных сферах широко используются различного рода системы автоматизации проектирования (САПР), использование которых актуально как при производстве материальных объектов, так и при разработке компьютерных систем. Современные САПРы позволяют автоматизировать многие процессы, связанные как с непосредственно проектированием объекта, так и с его разработкой и внедрением, и, как следствие, позволяют существенно сократить сроки производства и себестоимость продукта, а также минимизировать число ошибок, связанных с человеческим фактором. Важным направлением развития САПРов является их интеллектуализация, предъявляющая принципиально новые требования к технологиям разработки САПРов.

I. ПОДХОДЫ К ИНТЕЛЛЕКТУАЛИЗАЦИИ САПР

Большое внимание в современной литературе уделяется различным подходам к построению интеллектуальных САПРов. Анализ источников позволил выявить ключевые направления в области интеллектуализации САПРов, в рамках которых имеются значительные успехи.

Одним из важнейших и наиболее перспективных направлений в области построения интеллектуальных САПРов является **порождающее проектирование** (generative design [1]), которое предполагает, что компьютерная система сама выступает в роли активного участника процесса проектирования. Согласно концепции порождающего проектирования проектировщик задает требуемое минимальное описание параметров проектируемого объекта, после чего система самостоятельно порождает первоначальный вариант модели проектируемого объекта, которая далее уточняется и дорабатывается в диалоге с проектировщиком-человеком.

Другим важным направлением интеллектуализации САПРов, существенно менее развитым по сравнению с рассмотренным ранее, является направление, предполагающее, что САПР дол-

жен выполнять также функции **обучающей системы** [2]. При этом важно отметить, что рассматривается как обучение проектировщика, так и обучение самой системы в процессе работы. В свою очередь обучение проектировщика также может рассматриваться в двух аспектах: обучение работе с САПР, то есть изучение функциональных возможностей и принципов работы с конкретной системой, а также обучение в рамках собственно предметной области, в которой ведется проектирование, то есть изучение детальных аспектов работы проектируемых объектов, их назначения, особенностей проектирования и т.д. Развитие данного направления предъявляет дополнительные требования к технологиям, лежащим в основе таких систем, в частности, требует согласованного использования разнородной информации и различных моделей ее обработки.

Еще одним важным направлением интеллектуализации является **имитационное моделирование** (simulation modeling). Имитационное моделирование может широко использоваться на разных этапах, в частности: 1) моделирование поведения объекта разработки под воздействием различных факторов, в различной внешней среде и т.д., что особенно актуально при разработке сложных дорогостоящих систем, предназначенных для работы в непредсказуемых условиях; моделирование управления проектом, которое может применяться для тренировки или тестирования персонала, оценки потенциальных рисков при выборе определенной стратегии управления, отработки определенных практик и умений в условиях, аналогичных реальному проекту [3]. Для реализации возможности полноценного имитационного моделирования необходимо наличие средств, позволяющих, с одной стороны детально описывать сложные разнородные объекты с различных точек зрения, т.е. фактически интегрировать в рамках единой системы различные виды знаний, а также различные подходы к интерпретации таких описаний. Кроме того, необходимо наличие возможности легко модифицировать имеющиеся модели, в частности, легко менять число и виды воздействующих факторов, легко менять принципы по-

ведения объектов среды и самих моделируемых объектов и т.д.

Следующим этапом развития систем в области производства вообще является переход от САПРов к более общим PDM-системам (Product Data Management — система управления данными об изделии), и далее к комплексным PLM-системам (Product Lifecycle Management — система управления жизненным циклом изделия) [4], а также CALS-системам и технологиям (Continuous Acquisition and Lifecycle Support — непрерывная информационная поддержка поставок и жизненного цикла изделий).

Построение таких комплексных информационных систем требует унификации (стандартизации) разнородной информации. Для решения данной проблемы в настоящее время широко используется онтологический подход, как в области разработки программных систем [5], так и в других областях [6-7].

Таким образом, разного рода интеллектуализация средств автоматизации проектирования требует решения проблемы совместимости различных моделей представления и обработки информации, конкретный перечень которых для разных систем может существенно отличаться, поскольку зависит от объекта проектирования, требований к функциональным возможностям средств и т.д. Кроме того, развитие таких средств, в том числе связанное с изменениями объекта проектирования, требует снижения трудоемкости модификации указанных средств, что также в настоящее время является серьезной проблемой.

В данной работе предлагается решать указанные проблемы за счет использования унифицированных моделей представления и обработки информации, предложенных в рамках *Технологии OSTIS* [8], в частности, модели гибридных баз знаний и гибридных решателей задач. Системы, разрабатываемые по указанной технологии, называются *ostis-системами* [8]. Подробнее достоинства и особенности их разработки рассмотрены в работах [8-10]. Далее рассмотрим подробнее принципы построения средств автоматизации разработки *ostis-систем*.

II. ПРЕДЛАГАЕМЫЙ ПОДХОД

Средства автоматизации проектирования *ostis-систем* должны учитывать специфику таких систем. Реализация таких средств на единой с основной системой технологической основе — *Технологии OSTIS* — позволит значительно упростить интеграцию подсистем, решающих разные задачи, что, в свою очередь, позволит расширить функциональные возможности таких средств, за счет возможности совместного использования в рамках *ostis-систем* различных видов знаний и различных моделей решения задач с возможностью неограниченного расширения перечня используемых в *ostis-системе* видов знаний и мо-

делей решения задач без существенных трудозатрат.

В работах авторов были предложены *модель гибридных баз знаний ostis-систем* и модели представления различных видов знаний в рамках такой базы знаний [9], а также *модель гибридного решателя задач*, позволяющая интегрировать в рамках такого решателя различные модели решения задач [10]. Для удовлетворения указанных требований были также разработаны комплексная методика и соответствующие инструментальные средства автоматизации построения и модификации *ostis-систем* [9-10].

Таким образом, переход от традиционных компьютерных систем к *ostis-системам* позволит не только получить ряд преимуществ, связанных с такими ключевыми свойствами *ostis-систем*, как гибридность, модифицируемость и обучаемость, которые подробно рассмотрены в указанных выше работах, но и позволит вывести на принципиально новый уровень процессы проектирования и сопровождения компьютерных систем и степень автоматизации такого рода процессов.

СПИСОК ЛИТЕРАТУРЫ

1. Autodesk. [Online]. Available: <https://www.autodesk.com/solutions/generative-design>
2. S. M. Duffy and A. H. B. Duffy, "Sharing the learning activity using intelligent cad," *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 10, no. 2, p. 83–100, 1996.
3. L. S. Cardona-Meza and G. Olivar-Tost, "Modeling and simulation of project management through the PMBOK standard using complex networks," *Complexity*, vol. 2017, pp. 1–12, 2017.
4. J. Stark, *Product Lifecycle Management (Volume 1): 21st Century Paradigm for Product Realisation*, 3rd ed. Switzerland: Springer, Cham, 2015.
5. A. Emdad, "Use of ontologies in software engineering," 2008, pp. 145–150.
6. А. В. Федотова, И.Т. Давыденко, "Применение семантических технологий для проектирования интеллектуальных систем управления жизненным циклом продукции," *Известия высших учебных заведений. Машиностроение*, №3 (672), с. 74–81, 2016.
7. А. Fedotova, I. Davydenko, and A. Pfortner, "Design intelligent lifecycle management systems based on applying of semantic technologies," vol. 1. Springer International Publishing Switzerland, 2016, pp. 251–260.
8. В. В. Голенков, Н. А. Гулякина, "Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Часть 1: Принципы создания," *Онтология проектирования*, №1, с. 42–64, 2014.
9. I. Davydenko, "Semantic models, method and tools of knowledge bases coordinated development based on reusable components," *Open semantic technologies for intelligent systems*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 99–118.
10. D. Shunkevich, "Agent-oriented models, method and tools of compatible problem solvers development for intelligent systems," *Open semantic technologies for intelligent systems*, V. Golenkov, Ed., BSUIR. Minsk : BSUIR, 2018, pp. 119–132.

РАЗРАБОТКА МОДЕЛИ СИСТЕМООНТОЛОГИЧЕСКОГО РЕЕСТРА МУЛЬТИ-АГЕНТСКОЙ КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Галиевская Е. П.

Кафедра телекоммуникаций и информационных технологий, факультет радиофизики и компьютерных технологий, Белорусский государственный университет
Минск, Республика Беларусь
E-mail: catmycat3@gmail.com

Ключевые слова: онтология, мультиагентская система, корпоративная среда, реестр. Цель работы: исследование возможностей практической организации в среде коммуникаций локализованных сообществ типа «Корпоративные системы», замкнутых на основе предопределенного круга выполняемых производственных функций. Методы исследования: модель Остервальдера, IDEF0 методология, системноонтологический подход.

ВВЕДЕНИЕ

В наше время технические средства достигли такого уровня, что они мобильны и компактны, и мы на их основе можем сформировать принципиально распределенную среду автономно существующих сущностей. Так сформировалась сама идея о «автономных агентах».

Также возникает необходимость обратить пристальное внимание не просто на передачу данных, или телекоммуникацию, а на то как можно создать уровень общения, который обеспечивает взаимные контакты таким образом, что в среде, где «все слышат друг друга», создается замкнутое сообщество, в котором объекты взаимодействуют друг с другом, не обращая внимания на других, и не допуская других в свой круг общения. Вот здесь задача и состоит в том, как это все организовать, чтобы обеспечить нужную целостность такого сообщества вне зависимости от того, в какой среде передачи данных функционируют ее агенты.

Дело в том, что только обеспечение коммуникации не является достаточным, так как может быть достаточно много центров управления, что повышает риски возникновения различных проблем, как нарушение целостности. Эти риски надо нейтрализовать, а именно, ввести такой элемент, центр, который эту целостность мог бы контролировать.

Здесь мы исходим из того, что имеются теоретико-системные и онтологические механизмы, которые могут быть надстройкой к телекоммуникационной среде, которые: позволяют выделять предметную область как некоторую целостность в пространстве объектов различной природы и форм взаимодействия между ними; дают возможность разработать нужные механизмы, которые будут обеспечивать функционирование таких образований, что выделены в качестве предметных областей.

Из общих соображений, мы можем сказать, что это обязательно будет реестр и модель дан-

ных, которая принципиально носит распределенный характер, поскольку мы должны распределить это все между агентами.

I. ТРЕБОВАНИЯ К СОЗДАНИЮ ПРЕДМЕТНОЙ ОБЛАСТИ

В качестве платформы построения мультиагентских сообществ была избрана корпоративная среда. Это с функциональной точки зрения достаточно хорошо исследованная предметная область. В то же время ее автоматизация на современном этапе реально должна строиться на основе расширенного применения технологий формирования <<автономных агентов>>.

Поэтому в рамках целей данной работы на эту основу так как она есть с точки зрения ее понимания в виде сложившейся реальности мы далее и накладываем технологии построения распределенных систем автоматизации их информационного обслуживания.

При этом в отношении потенциальных агентов используется не весь спектр желаемых требований по их интеллектуальности [1], а только то, что минимально необходимо для решения задач их функционирования в распределенной форме управления данными в пределах собственных полномочий и способностью обмениваться ими с другими агентами. К ним, как это представляется, можно отнести:

- автономность - способность функционировать и осуществлять контроль событий в рамках заданных полномочий;
- реактивность - восприятие состояния среды и реакция на его изменение;
- общительность - взаимодействие и коммуникация с другими агентами;
- знание - наличие данных о себе, о других агентах в рамках распределенной модели их функционирования.

II. СИСТЕМНО-ОНТОЛОГИЧЕСКАЯ ИНТЕРПРЕТАЦИЯ КОРПОРАТИВНОЙ СРЕДЫ

Абстрактные конструкции теории систем и схем онтологий не ориентированы на подробности их тематической интерпретации.[2] Однако в разработках на общие положения обязаны накладываться ограничения, которые задаются условиями проектирования.

Эту ситуацию продемонстрировала на примере зоомагазина, где по технологиям стандарта IDRF0 получена замкнута функциональная структура данной предметной области.

Здесь же была заложена основа для определения состава ролевых сущностей, которые могут выступать в качестве агентов, образующих составы фактов в процессе взаимодействий между собой и с объектами окружающей действительности.

Данная предметная область будет выступать и в дальнейшем в качестве предметной платформы отработки вопросов построения распределенной модели данных, управляемой агентами в рамках их полномочий.

III. РАЗРАБОТКА

На примере предметной области <<Зоомагазин>> была построена модель в виде ее функциональной декомпозиции как корпоративной среды, в которой установлен состав объектов, так или иначе связанных с протекающими в ней бизнес процессами.[2–4]

В отношении этого состава объектов и формируется реестр, который обеспечивает поддержание целостности как самой системы <<Зоомагазин>>, так и тех моделей данных, которые отражают свойства и состояния этих объектов.

Конструктивно реестр представляется двумя видами таблиц:

- таблица нулевого слоя 3D модели;
- таблицы справочники T_{i*} конфигурации

[3].

Таким образом, реестр корпоративной среды – это база данных с двойной системой ссылок, которая с одной стороны поддерживает ее внутреннюю целостность, а с другой стороны предоставляет возможность <<видеть>> те же объекты во внешнем окружении.

Такая структура достаточно просто декомпозируется при необходимости образования распределенной структуры, части которой могут быть закреплены за объектами ролевых типов, на которые будут возложены обязанности агентов.

ЗАКЛЮЧЕНИЕ

Так как представления бизнес-процессов предметными специалистами в не отражают роли агентов, то трансформация их видения должна проводиться через IDEF методологии, где уже присутствуют нужные механизмы.

А именно, инфологический анализ IDEF модели позволяет установить состав всех типов объектов, ответственных за реализацию функций корпоративной среды. И с онтологической точки зрения они и образуют сообщество в виде замкнутого мира.

Далее может использоваться теоретико-системная концепция конфигурационного пространства, которая позволяет получить выражение реестра в виде табличной формы с двойной системой внутренних и внешних ссылок, совмещенной с моделью данных, хранящей сведения обо всех конкретных объектах корпоративной среды и их взаимодействиях в ходе протекания бизнес-процессов.

При необходимости образования распределенной среды такая структура достаточно просто декомпозируется так, что ее части могут быть закреплены за объектами ролевых типов, на которые будут возложены обязанности агентов.

При этом ссылочная целостность поддерживается путем взаимодействия локальных идентификаторов со внешними ссылками.

Показано, что на этой основе может быть создана распределенная вычислительная среда программных агентов, которая должна обеспечивать реализацию всего комплекса вопросов автоматизации в соответствии с созданной онтологической моделью. А реализация агентов может базироваться на типовой архитектуре программных приложений.

СПИСОК ЛИТЕРАТУРЫ

1. Городецкий В. И. Многоагентные системы / Городецкий В. И., Грушинский М. С., Хабалов А. В. // труды 7-й международной конференции по информационным сетям и системам – 2000.
2. Емельяненко В. И. Модели данных многообъектных систем //X Международная научно-техническая конференция «Информационные технологии в промышленности, логистике и социальной сфере». – 2019. – С. 68–70.
3. Емельяненко В. И. Трёхмерные модели распределённых баз данных //Международный конгресс по информатике: информационные системы и технологии. CSIST. – 2013. – С. 178–182.
4. Остервальдер А. Построение бизнес-моделей: Настольная книга стратега и новатора / Остервальдер А., Пинье И.// – 2010.

РАСПРЕДЕЛЕНИЕ ПАМЯТИ В НЕОГРАНИЧЕННОМ ЛИНЕЙНОМ АДРЕСНОМ ПРОСТРАНСТВЕ

Ивашенко В. П.

Кафедра интеллектуальных информационных технологий, Факультет информационных технологий и управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ivashenko@bsuir.by

В статье рассматривается подход к построению и реализации алгоритмов распределения неограниченно линейно адресуемой памяти для универсальных моделей решения задач

ВВЕДЕНИЕ

В технических системах, ориентированных на решение задач [1], особый интерес представляют универсальные модели [2], способные обеспечить интеграцию различных моделей решения задач [3]. В соответствии со схемой уровней управления [4] для систем управляемых знаниями, относящихся к системам ориентированным на решение задач, одной из ключевых задач управления устройствами является задача распределения памяти [5]. Ранее были рассмотрены программная модель и алгоритмы, обеспечивающие близкое к оптимальному решение этой задачи для ограниченного, конечного линейного адресного пространства (ЛАП) [6]. Однако, решение задач в рамках универсальных моделей требует эффективных механизмов работы с неограниченной памятью [7], при этом аналогичные задачи работы с объектами неограниченного размера актуальны для уровней управления данными и знаниями. Для реализации этих механизмов необходимо дать ответ на вопросы: «Какую платформу выбрать?», «Как кодировать участки памяти?», «Какие структуры следует использовать для доступа к участкам памяти, чтобы обеспечить их поиск и поддержать их изменение?», «Какова стратегия распределения памяти?».

1. ПРЕДСТАВЛЕНИЕ СВОБОДНЫХ УЧАСТКОВ НЕОГРАНИЧЕННОЙ ПАМЯТИ

Рассмотрим ответ на поставленные вопросы, исходя из известных свойств реализованных моделей распределения памяти для ограниченного ЛАП. Основными из этих свойств являются следующие:

1. стратегия «первый подходящий» является одной из близких к оптимальным стратегиям с точки зрения минимизации требований на размер памяти, включающей используемую для хранения данных и подверженную внешней фрагментации;

2. для стратегии «первый подходящий» временная сложность реализованных алгоритмов моделей распределения памяти оценивается как полилоглинейная;

3. для хранения структур, описывающих свободные участки памяти, используются сами

эти участки и дополнительный участок памяти ограниченного (неизменного в соответствии с размером адресуемой памяти) размера.

Предлагаемое представление свободных участков памяти ориентировано на сохранение этих свойств. С целью минимизации затрат памяти на дополнительные структуры данных, предлагаемое представление так же отталкивается от моделей наиболее экономичных в этом плане, а именно – от «систем близнецов» [5]. Недостатком «систем близнецов» является увеличение затрат (до двух раз) на размер памяти подверженной внутренней фрагментации.

Так как память имеет неограниченный размер, то и адреса ячеек памяти требуют неограниченного количества разрядов. С целью минимизации затрат на адаптацию используемых структур данных к возрастанию значащих разрядов адреса, для их хранения используются отдельные ячейки-регистры. Именно поэтому в качестве платформы реализации рассмотрена ранее предложенная модель обработки информации, которая как раз ориентирована на обеспечение решения подобных задач [7].

Структура свободных участков памяти, как это характерно для «систем близнецов» организуется в виде дерева, каждый полностью или частично свободный блок имеет размер равный суммарному размеру $2^k - 1$ ячеек памяти (рис. 2). Центральная ячейка любого полностью или частично свободного блока хранит служебную информацию, описывающего состояние блока и состояние ячеек блоков следующего уровня в дереве. В центральной ячейке для хранения этой информации используется до семи бит, соответствующих признакам: 1) занятости (отсутствия) младшего блока-потомка блока в дереве; 2) занятости (отсутствия) старшего блока-потомка блока в дереве; 3) блока как старшего блока-потомка у блока-родителя; 4) блока как текущего корневого блока в дереве; 5) младшего блока-потомка как блока со свободным блоком максимального размера; 6) блока, как не являющегося полностью свободным блоком, блоком-листом в дереве; 7) старшего блока-потомка как «блока-фаворита», поиск и выделение свободного блока в котором осуществляются в первую очередь,

при прочих равных. Установка значения признака «блока-фаворита», соответствует реализуемой стратегии распределения памяти. Для распределения памяти «первый подходящий» с выделением блоков в направлении только одной стороны, достаточно установить значение признака одинаковым для всех блоков в дереве. Для распределения памяти с выделением блоков с двух сторон (см. рис. 1) можно чередовать значение этого признака при подъёме от предыдущего корневого блока дерева к новому текущему корневному блоку дерева, при этом значение этого признака в новых блоках поддерева нового блока-потомка корневого блока дерева совпадает со значением этого признака в корневом блоке дерева.

II. ВЫДЕЛЕНИЕ И ВЫСВОБОЖДЕНИЕ УЧАСТКОВ НЕОГРАНИЧЕННОЙ ПАМЯТИ

Задачи выделения свободного блока и высвобождения блока памяти по указанному адресу заданных размеров в неограниченном ЛАП сведены к следующим подзадачам: 1) вычисление размера блока-родителя и блока-потомка в дереве; 2) вычисление адреса ячейки блока-родителя; 3) вычисление адреса младшего или старшего блока-потомка; 4) вычисление адреса и размера корневого блока; 5) вычисление «блока-фаворита» у указанного блока; 6) вычисление адреса подходящего по размеру блока в поддереве; 7) сравнение размеров максимальных блоков в поддеревах младшего и старшего потомков; 8) создание нового корневого блока и блока-потомка. Каждая из этих задач решается соответствующим алгоритмом, описанном в виде процедуры на языке выбранной платформы реализации. Кроме структур, описывающих состояние участков памяти, используются дополнительные регистры (в количестве не менее 22-х), которые хранят следующую информацию: адрес выполняемой команды, адреса вызовов и возвратов из процедур, параметры процедур распределения памяти – размер распределяемого блока, адрес выделенного блока, размер блока в дереве участков памяти, адрес ячейки блока в дереве участков памяти, результаты процедур, их параметры и управляющие флаги, включающие значения признаков свободного блока, старшего блока-потомка родителя, «блока-фаворита», блока максимального размера, корневого блока и т.д.

Временная сложность реализованных алгоритмов выделения и высвобождения блоков па-

мяти может быть выражена как $O(\ln^3(2^n) * f(n))$, где 2^n – количество бит распределённой памяти (количество используемых значащих разрядов адресных регистров не превышает n), $f(n)$ – время доступа к ячейке памяти по адресу. Пространственная сложность реализованных алгоритмов – $O(\ln(2^n))$. Перерасход памяти из-за внешней фрагментации, выраженный размером фактически используемой памяти: $O(m * \ln(m))$, где m – максимальный размер памяти, занятой данными.

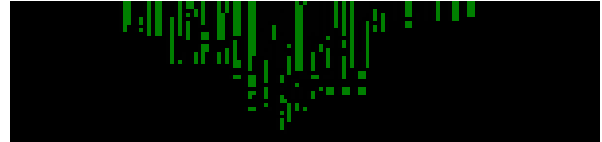


Рис. 1 – История изменений состояний памяти в неограниченном ЛАП в процессе её распределения

ЗАКЛЮЧЕНИЕ

Алгоритмы и демонстрационный пример (рис. 1) модели распределения неограниченной линейно адресуемой памяти реализованы средствами JavaScript (<https://bitbucket.org/version/openjsvmm/>).

СПИСОК ЛИТЕРАТУРЫ

1. Wolfram, S. Jeopardy, IBM, and Wolfram|Alpha. Mode of access: <http://blog.stephenwolfram.com/2011/01/jeopardy-ibm-and-wolframalpha/> Date of access: 17.09.2017.
2. Wolfram, S. A New Kind of Science. Champaign, IL: Wolfram Media, Inc. 2002. p. 1197.
3. Luger, G. Stubblefield, William (2004), Artificial Intelligence: Structures and Strategies for Complex Problem Solving (5th ed.), The Benjamin/Cummings Publishing Company, Inc., p. 720.
4. Ивашенко, В. П. Операции управления массивами данных в линейно адресуемой памяти / В. П. Ивашенко, С. В. Синцов // Доклады БГУИР, №10, – Минск, 2016. – С. 86–93.
5. Кнут, Д. Искусство программирования. Том 3. Сортировка и поиск. – М.: Вильямс, 2014. – 832 с.
6. Ивашенко, В. П. Алгоритмы полилогарифмической временной и логарифмической пространственной сложности для системы динамического распределения линейно адресуемой памяти с однородным доступом к данным / В. П. Ивашенко // Карповские научные чтения : сб. науч. ст. – Минск, 2012. – Вып. 6, ч. 1. – С. 196–201.
7. Ивашенко, В. П. Модели обработки информации и программные средства для универсальных моделей решения задач / В. П. Ивашенко // Информационные технологии и системы 2017 (ИТС 2017) : материалы междунар. науч. конф. (Республика Беларусь, Минск, 25 октября 2017 года) / редкол. : Л. Ю. Шилин [и др.]. – Минск: БГУИР, 2017. – С. 106–107.

...	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	...
	1	3	1	7	1	3	1	15	1	3	1	7	1	3	1		
		3		7		3		15		3		7		3			
				7				15				7					
								15									

Рис. 2 – Иерархия кодирования блоков соответствующих размеров в неограниченной памяти

ОНТОЛОГИЯ ПРЕДМЕТНОЙ ОБЛАСТИ ПРОСТРАНСТВЕННЫХ СУЩНОСТЕЙ ДЛЯ СИСТЕМЫ СЕМАНТИЧЕСКОГО АНАЛИЗА ИЗОБРАЖЕНИЙ

Искра Н. А., Межень А. Л., Шункевич Д. В.

Кафедра электронных вычислительных машин, Кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {nastassialmezhen, shunkevichdv}@gmail.com, niskra@bsuir.by

В данной статье рассмотрена структура онтологии предметной области пространственных сущностей, взаимосвязь предметной области с другими предметными областями верхнего уровня, а также приведен пример использования отношений, входящих в данную онтологию.

ВВЕДЕНИЕ

При решении задачи понимания и интерпретации изображений на этапах выделения и классификации объектов на изображении, их атрибутов и отношений между ними для получения более полного и точного описания выделенных объектов и их взаимосвязей используются базы знаний и, в частности, онтологии, соответствующие различным предметным областям (ПрО).

Интеграция знаний о структуре изображения и представленных на нем объектах в базе знаний позволяет повысить степень понимания изображения и дополнить результаты анализа новыми знаниями, которые могут быть порождены на основе этих результатов и сведений из базы знаний [1]. Таким образом, имеет смысл расширять базу знаний предметными областями, которые будут содержать информацию, охватывающую широкий спектр возможных ситуаций.

В данной работе рассмотрим модель *Предметной области пространственных сущностей*, в частности, исследуемые в ней отношения, которые позволяют более точно интерпретировать различного вида ситуации с точки зрения положения объектов в пространстве. Для описания модели ПрО будет использоваться подход, рассмотренный в [2], в рамках которого онтология трактуется как спецификация ПрО.

I. ОНТОЛОГИЯ ПРЕДМЕТНОЙ ОБЛАСТИ ПРОСТРАНСТВЕННЫХ СУЩНОСТЕЙ

Максимальным классом объектов исследования рассматриваемой предметной области является понятие *пространственной сущности*.

В качестве *пространственной сущности* может выступать любой класс материальных объектов, временно находящихся в определенном положении в пространстве, поэтому *Предметная область пространственных сущностей* тесно связана с такими предметными областями, как:

– *Предметная область материальных сущностей*;

– *Предметная область ситуаций и событий*;

– *Предметная область временных сущностей*.

Кроме определения объектов и характеризующих их атрибутов, одним из средств описания смысла изображения является установление между объектами пространственных отношений [3].

Основными исследуемыми пространственными отношениями рассматриваемой онтологии, передающими взаимное расположение объектов на изображении с точки зрения наблюдателя, являются:

– **на*** – ориентированное бинарное отношение, первой и второй компонентами связок которого являются знаки *материальных сущностей*, где первая из которых находится на второй.

– **рядом*** – неориентированное бинарное отношение, первой и второй компонентами связок которого являются знаки *материальных сущностей*, которые располагаются с какой-либо стороны относительно друг друга, находятся в непосредственной близости.

– **над*** – ориентированное бинарное отношение, первой и второй компонентами связок которого являются знаки *материальных сущностей*, первая из которых находится выше второй.

– **под*** – ориентированное бинарное отношение, первой и второй компонентами связок которого являются знаки *материальных сущностей*, первая из которых находится ниже второй.

– **внутри*** – ориентированное бинарное отношение, означающая расположение одной *материальной сущности* в другой, где знаки данных сущностей являются соответственно первой и второй компонентами связок отношения.

При этом отношение *рядом** можно уточнить с использованием следующих подклассов отношений:

рядом*

\leq разбиение*:

- {
- быть левее*
- быть правее*
- перед*
- позади*
- }

Благодаря данным подклассам отношений при построении семантических описаний изображений можно конкретизировать их содержание и пространственную взаимосвязь объектов на изображении.

С помощью всех вышеперечисленных отношений можно увеличить количество классов маловероятных ситуаций в рамках рассматриваемой предметной области и повысить качество верификации первоначальных семантических моделей изображений. В рамках дорожных ситуаций примерами маловероятных ситуаций могут быть, например, следующие: машина внутри человека, машина под машиной и др.

II. ПРИМЕР ИСПОЛЬЗОВАНИЯ ПОНЯТИЙ ПРЕДМЕТНОЙ ОБЛАСТИ

Рассмотрим пример использования рассмотренных понятий для ситуации, изображенной на рисунке 1.



Рис. 1 – Исходное изображение

На рисунке 2 выделены объекты, которые будут использоваться для построения семантической модели изображения, с указанием классов объектов, к которым они принадлежат.

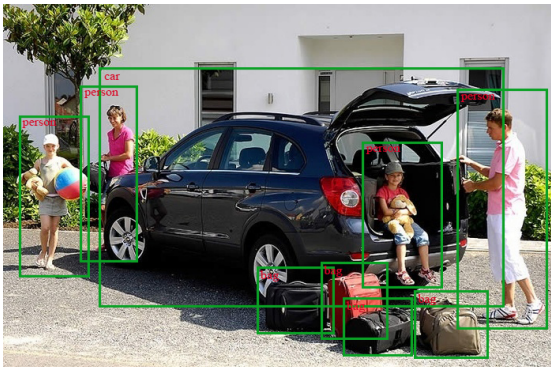


Рис. 2 – Исходное изображение с выделенными объектами

После построения первоначального варианта семантической модели изображения, верификации данного варианта с помощью сведений, хранящихся в базе знаний, и корректирования всех выявленных ошибочных конструкций в результате получаем семантическую модель, представленную на рисунке 3 на языке SCg [4].

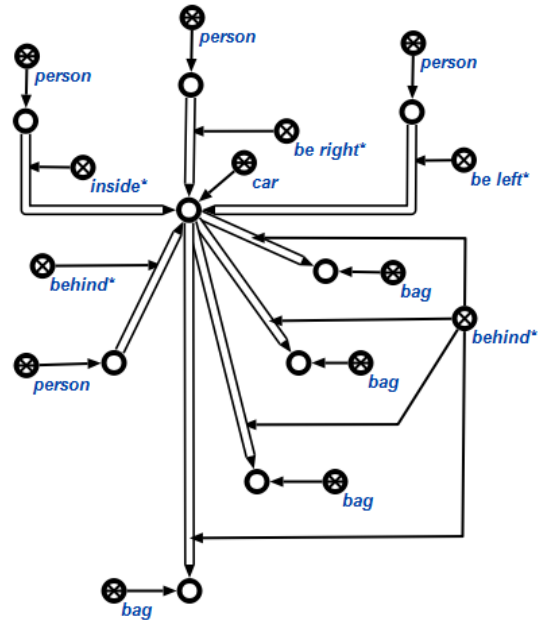


Рис. 3 – Семантическая модель исходного изображения

ЗАКЛЮЧЕНИЕ

В работе рассмотрена Предметная область пространственных сущностей, которая используется в контексте решения задач анализа и понимания изображений.

Так как описание изображения напрямую зависит от наполненности базы знаний, то данная предметная область помогает устанавливать расположение объектов относительно друг друга и других объектов, что позволит системе составлять описания изображений, близкие к естественному.

СПИСОК ЛИТЕРАТУРЫ

1. Iskra, N. A. Neural network based image understanding with ontological approach / N. Iskra, V. Iskra, M. Lukashevich // Open semantic technologies for intelligent systems (OSTIS-2019): materials of IX International.sc.-tech.conf. — Mn.: BSUIR, 2019. — Iss. 3 — P. 113–122.
2. Davydenko, I. T. Semantic models, method and tools of knowledge bases coordinated development based on reusable components / I. Davydenko // Open semantic technologies for intelligent systems (OSTIS-2018): materials of VIII International.sc.-tech.conf. — Mn.: BSUIR, 2018. — Iss. 2. — P. 99–118.
3. Соколова, Е. Г., Болдасов, М. В. Принципы построения семантических аннотаций содержания изображений / Е. Г. Соколова, М. В. Болдасов // Компьютерная лингвистика и интеллектуальные технологии: По материалам ежегодной Международной конференции Диалог. — 2006. — № 5 (12). — С. 474–480.
4. Метасистема IMS [Электронный ресурс]. — Режим доступа: <http://ims.ostis.net>. — Дата доступа: 04.10.2019.

АНАЛИЗ ВАКАНСИЙ ДЛЯ ИССЛЕДОВАНИЯ РЕГИОНАЛЬНОГО РЫНКА ТРУДА

Криштапович А. С., Адуцкевич И. А.

Факультет радиофизики и компьютерных технологий Белорусского государственного университета

Минск, Республика Беларусь

E-mail: krisht96@gmail.com, dutsik@gmail.com

Данная работа посвящена созданию системы анализа вакансий для исследования регионального рынка труда и предоставления сведений о наиболее востребованных знаниях и умениях, которые работодатели хотят видеть в соискателях. В ходе выполнения работы создана система анализа вакансий для исследования регионального рынка труда. Созданная система протестирована для ресурса jobs.tut.by (регионы: Минск и Брест).

ВВЕДЕНИЕ

Рынок труда представляет собой наиболее важный и сложный сегмент совокупного рынка, фундамент, на котором базируется экономика страны и от функционирования которого зависит ее стабильность. Этот рынок является сложной многомерной динамической системой с большим количеством обратных связей [1]. Кроме того, количество информации о вакансиях и претендентах на работу постоянно растет. Все это приводит к тому, что анализ рынка труда традиционными аналитическими средствами становится все более сложным и недостаточно полным.

Мы живем в мире, где все очень быстро меняется. Люди становятся все более мобильными, не привязанными к конкретному населенному пункту, региону и даже стране. Для удовлетворения своих профессиональных амбиций, связанных с карьерным ростом, перспективой новых проектов, повышением уровня зарплаты и квалификации соискатели готовы к изменению места жительства. Кроме того, в настоящее время имеет место отсутствие сбалансированности рынка труда, то есть предложения системы образования не полностью соответствуют запросам бизнеса. Как результат – перенасыщение специалистами одних профессий (юристы, экономисты и т.д.) и недостаток в других областях (IT-специалисты, рабочие). В этой связи актуальным становится своевременное предоставление информации о потребностях рынка труда в разрезе различных регионов и областей с целью обеспечить высокий уровень занятости населения. Наличие подобных данных позволит увидеть, как общую картину, так и взглянуть на каждый сегмент в отдельности. В условиях инновационного развития работодателям важны не только профессиональные знания и навыки специалистов, но и другие навыки (такие, как свободное владение иностранными языками, умение работать с людьми, самостоятельно решать проблемы и принимать решения и др.) [2]. Для эффективного подбора кадров разрабатываются матрицы компетенций персонала – это набор

требований для сотрудников компании, в которой отображается набор компетенций и их уровень для конкретных должностей. Однако потенциальному соискателю сложно быстро охватить весь спектр требований, предъявляемых различными нанимателями. Разрабатываемая система позволяет выявить наиболее распространенные запросы для того или иного города или региона, что повысит эффективность качественной подготовки к собеседованиям и как следствие будет способствовать успешному трудоустройству. Данные собираются с открытых интернет ресурсов кадровых агентств с использованием технологии Web Mining, накапливаются и анализируются, результаты предоставляются в виде отчетов в текстовом и графическом отображении.

I. ЗАДАЧИ РАЗРАБАТЫВАЕМОЙ СИСТЕМЫ

Существует множество систем публикации вакансий, например: indeed.com, jobs.tut.by и другие. Эти системы оборудованы гибкой системой поиска вакансий по всевозможным критериям, в каждой вакансии приводится список требований работодателя, перечень необходимых и желаемых знаний и навыков.

Рассмотрим, например, ситуацию на рынке вакансий в сфере IT. В настоящее время вследствие бурного развития данной сферы вакансий достаточно много, но и список необходимых знаний и навыков весьма внушителен. Для того чтобы хорошо освоить все требуемые навыки может уйти далеко не один месяц подготовки, однако, существуют наиболее востребованные знания и навыки, которые большинство работодателей хотят видеть в соискателе. Имея список подобных наиболее востребованных навыков можно намного ускорить подготовку к важному интервью, а так же существенно повысить свои шансы на успех по сравнению с другими кандидатами. Именно эти важные знания могут быть получены путем сбора необходимых данных о вакансиях и анализа рынка труда. Так же огромное значение имеет региональный анализ рынка вакансий, так как в настоящее время существенно возросла мобильность населения.

В качестве ресурса для сбора данных был выбран веб-сайт jobs.tut.by, так как он является ведущим сайтом для поиска работы на Постсоветском пространстве. Согласно информации, представленной на веб-сайте jobs.tut.by на 2/10/2019 58 573 компании предлагают 28 146 вакансий во всевозможных профессиональных областях. В качестве технологии сбора данных использовался веб-скрапинг. Данная система реализована на языке Python, так как он является высокоуровневым языком программирования общего назначения, ориентированным на повышение производительности разработчика и читаемости кода [3].

II. РЕЗУЛЬТАТЫ РАБОТЫ СИСТЕМЫ АНАЛИЗА ВАКАНСИЙ

Результатом применения данной системы для анализа рынка вакансий Минска (поиск производился для вакансии <<Software Engineer>>) стала нижеприведенная гистограмма.

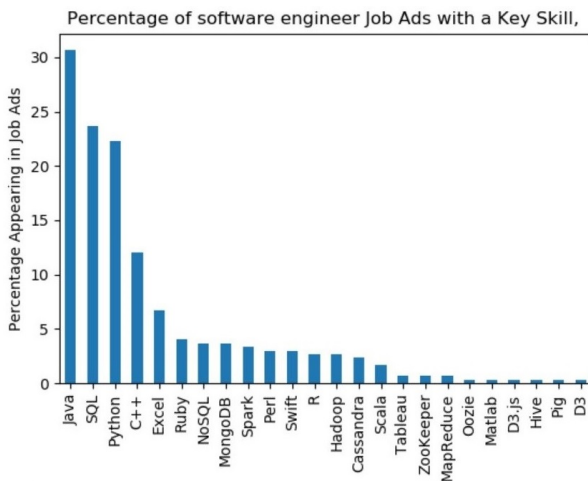


Рис. 1 – Результат анализа рынка вакансий Минска

Видно, что лидирующие позиции среди самых востребованных навыков в Минске удерживают знание языка программирования Java и знание реляционных баз данных и языка SQL. Однако язык программирования Python также очень востребован в данном регионе, из чего следует, что технологии, актуальные для отрасли Big Data становятся все более востребованными в Минске.

Рассмотрим теперь аналогичную гистограмму, полученную при анализе вакансий города Брест. Существенные отличия данной гистограммы от аналогичной, но полученной для Минска, объясняются количеством вакансий, которые доступны в каждом из регионов. На момент написания работы по запросу <<Software Engineer>> на веб-сайте jobs.tut.by для региона

Минск было найдено 304 вакансии, по аналогичному запросу для региона Брест была найдена 51 вакансия.

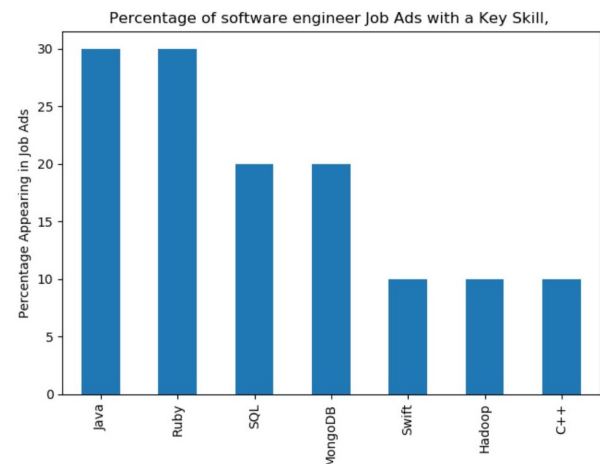


Рис. 2 – Результат анализа рынка вакансий Бреста

Видим, что, как и в Минске, первое место занимает знание языка программирования Java, однако в этом регионе работодатели более заинтересованы в специалистах, знающих язык программирования Ruby. Знание реляционных баз данных и языка SQL является также очень востребованным в данном регионе. Стоит отметить, что знание языка программирования Python и технологий из области Big Data является менее востребованным в Бресте, чем в Минске, что свидетельствует о том, что развитие данной отрасли пока проходит более интенсивно в столице Республики Беларусь. В Бресте работодатели более ориентированы на специалистов, обладающих навыками в веб-разработке.

ЗАКЛЮЧЕНИЕ

В заключение хочется отметить, что получение данных о наиболее востребованных знаниях и навыках в режиме реального времени является действительно актуальной и важной задачей, так как эти данные могут существенно помочь при выборе будущей профессии, поиске нового места работы или определении направления для саморазвития. Анализ и структуризация собранных данных позволяют обнаруживать новые, ранее неизвестные знания, которые в дальнейшем можно будет использовать на практике.

СПИСОК ЛИТЕРАТУРЫ

1. Поиск знаний в Internet / Д. В. Ландэ [и др.]. – СПб.: Питер, 2003. – 173 с.
2. Roussopoulos, N. D. A semantic network model of data bases / N. D. Roussopoulos // New-York: John Wiley & Sons Inc – 2008. – Vol. 35, № 13. – P. 75–80.
3. Скрапинг веб-сайтов с помощью Python / Р. Митчелл. – М.: ДМК Пресс, 2016. – 480 с.

СЕМАНТИЧЕСКИЕ МОДЕЛИ И СРЕДСТВА КОНТРОЛЯ ЗНАНИЙ ПОЛЬЗОВАТЕЛЕЙ ОБУЧАЮЩИХ СИСТЕМ

Ли Вэньцзу, Цянь Лунвэй
Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: lwzggml@gmail.com, qianlw1226@gmail.com

В работе рассматривается подход автоматической генерации вопросов и автоматической проверки ответов в интеллектуальной обучающей системе.

ВВЕДЕНИЕ

Одной из наиболее важных особенностей интеллектуальных обучающих систем является возможность использования базы знаний для автоматической генерации вопросов для пользователей и возможность автоматически проверять ответы пользователей. Однако большинство систем только генерируют вопрос на выбор из вариантов. В свою очередь, методы проверки ответов, как правило, основаны на ключевых словах или векторном расстоянии, которые не учитывают семантическое подобие между ответами. Данная особенность не позволяет современным обучающим системам оценивать подробные развернутые ответы и частично правильные ответы.

В данной работе предлагается использовать технологию OSTIS [4] в сочетании с существующими методами для автоматической генерации различных типов вопросов, а также подход к проверке ответов с учетом их семантики.

I. СУЩЕСТВУЮЩИЕ ПОДХОД И ПРОБЛЕМА

В рамках технологии автоматической генерации вопросов (ТАГВ) рассматриваются различные методы, позволяющие автоматически генерировать вопросы из электронных документов, корпуса текстов и базы знаний с помощью компьютерных технологий [1].

В работах Andreas Papasalouros [2] рассмотрена система автоматической генерации вопросов на выбор, которая в основном использует отношения между родительскими классами и подклассами в онтологии OWL для автоматического генерирования вопросов на выбор.

Li Hui предложен метод использования онтологии для автоматической генерации вопросов на выбор, вопросов на заполнение пробелов и вопросов-суждений [1].

Несмотря на то, что рассмотренные методы используют различные виды отношений и атрибутов в базе знаний для генерации вопросов, они позволяют генерировать только объективные вопросы (те, на которые существует только один стандартный правильный ответ), при этом в настоящее время генерируются самые основные и простые вопросы, которые не позволяют в достаточной степени оценить знания пользователей. С

учетом перечисленных проблем в данной работе предлагаются некоторые методы для автоматической генерации вопросов (как субъективных, так и объективных).

Субъективные вопросы не имеют однозначно правильных ответов, поэтому проверка ответа субъективных вопросов требует установления подобия между стандартным ответом и ответом пользователя. Современные методы установления подобия в основном включают методы, основанные на сравнении семантических графов и текстов естественного языка.

К первому классу относится метод SPICE (Semantic Propositional Image Caption Evaluation), который используется для оценки качества автоматически генерируемых подписей к рисункам. Идея метода заключается в выявлении подобия между автоматически сгенерированными подписями к рисункам и подписями к рисункам, сделанными человеком вручную [3].

К методам сравнения подобия с использованием естественного языка относятся:

1. Метод на основе сопоставления ключевых слов
2. Метод на основе векторного пространства
3. Методы на основе глубокого обучения

Вопросы и стандартные ответы, сгенерированные в обучающей системе, хранятся в базе знаний в виде семантических графов, поэтому метод SPICE может быть положен в основу метода установления подобия семантических графов, записанных в SC-коде, который является языком кодирования информации в рамках технологии OSTIS [3, 4].

II. ПРЕДЛАГАЕМЫЙ ПОДХОД

В работе [1] предложен ряд стратегий генерации вопросов. В рамках данной работы предлагается реализовать эти стратегии с использованием технологии OSTIS, благодаря чему впоследствии могут быть предложены новые методы в сочетании с особенностями конкретной базы знаний. В качестве примера в данной работе выбрана баз знаний по дискретной математике.

- Стратегия генерации на основе классов
Отношение класс-подкласс

$S_i \subseteq C (i \geq 1)$, (S -подкласс, i -номер подкласса, C -родительский класс).

Вопросы-суждения:

Для генерации вопросов-суждений могут использоваться два условия: $S_i \subseteq C$ является истинным или $S_i \subseteq C$ является ложным

Вопросы на заполнение пробелов:

Родительский класс может содержать несколько подклассов. То есть $S_1 \cup S_2 \cup S_3 \dots \cup S_i = C$. Пустая позиция, которую нужно заполнить, является одной или несколькими из S_i .

Вопросы на выбор:

Подкласс S_i используется как правильный вариант, родительские или родственные классы C используются в качестве неправильных вариантов [1, 4]. На рисунке 1 приведен пример вопроса на выбор на языке SCg [4].

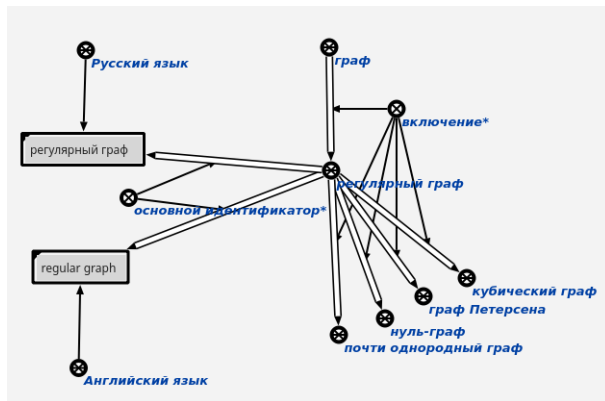


Рис. 1 – Пример вопроса на выбор

Эквивалентные классы

Непересекающиеся классы

$C = D_1 \cup D_2 \dots \cup D_i (i \geq 1, D_i \cap D_j = \emptyset)$.

Вопросы-суждения:

$C = D_1 \cup D_2 \dots \cup D_n$ является истинным или ложным.

- Стратегия генерации вопросов на основе свойств
- Стратегия генерации на основе экземпляров (отношения между экземплярами и связь между классами и экземплярами).
- Сочетание стратегий для генерации сложных вопросов

Операция объединения \cup

$S_1 \subseteq D, S_2 \subseteq D$, то $S_1 \cup S_2 \subseteq D$.

Операция пересечения \cap

$C \subseteq D, C \subseteq E$, то $D \cap E = C$.

- Вопросы и ответы на толкование терминов. Вопросы данного класса чаще всего хранятся в системе в виде готового шаблона.

В работе [3] предложен метод сравнения подобия между текстами по семантическому графу. Комбинируя метод SPICE с некоторыми характеристиками семантических графов, записанных в SC-коде, основанном на теоретико-

множественном подходе, можно установить подобие между такими графами [4, 5].

Для вычисления подобия (F) между семантическими графами подписей-кандидатов и семантическими графами подписей-образцов используется метод кортежей.

Способ преобразования подписи c в семантический граф:

$$G(c) = \langle O(c), E(c), K(c) \rangle \quad (1)$$

где c – подпись кандидата, $S = \{s_1, \dots, s_m\}$, F – подобие, $G(c)$ – семантический граф подписи-кандидата, $G(S)$ – семантический граф подписей-образцов, $G(s_i)$, $s_i \in S$ – семантический граф для одного образца, C – множество классов объектов, R – множество типов отношений, A – множество типов атрибутов, $O(c) \subseteq C$ – множество упоминаний объектов в c , $E(c) \subseteq O(c) \times R \times O(c)$ – множество отношений между объектами, $K(c) \subseteq O(c) \times A$ – множество атрибутов, связанных с объектами.

III. ЗАКЛЮЧЕНИЕ

В работе рассмотрены существующие методы генерации вопросов и проверки ответов в обучающих системах. Рассмотрены методы автоматической генерации субъективных вопросов и объективных вопросов, по сравнению с оригинальными методами были сделаны некоторые улучшения. Некоторые из этих методов были завершены исследователями, поэтому они верны в теории, но необходима практическая апробация. Следующим шагом станет разработка универсальной подсистемы проверки знаний учащихся для обучающей системы по дискретной математике, построенной с использованием технологии OSTIS.

СПИСОК ЛИТЕРАТУРЫ

1. Li, H. Research on item automatic generation based on DL and domain ontology / H. Li // Journal of Changchun University of Technology (Natural Science Edition). – 2012. – Vol. 33(04). – P.461-464.
2. Andreas, P., Konstantinos, K., Konstantinos, K. AUTOMATIC GENERATION OF MULTIPLE-CHOICE QUESTIONS FROM DOMAIN ONTOLOGIES / P. Andreas, K. Konstantinos, K. Konstantinos // IADIS International Conference e-Learning, 2008.
3. Peter, A., Basura, F., Mark, J. SPICE: Semantic Propositional Image Caption Evaluation / A. Peter, F. Basura, J. Mark // Computer Vision and Pattern Recognition (cs.CV), 2016.
4. Голенков, В. В., Гулякина, Н. А. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Часть 1: Принципы создания. / В. В. Голенков, Н.А. Гулякина // Онтология проектирования. – 2014. – №1. – с.42-64
5. Документация. Технология OSTIS. [Электронный ресурс]. – Режим доступа: <http://ims.ostis.net>. – Дата доступа: 28.08.2017

БИБЛИОТЕКА ДЛЯ СОЗДАНИЯ ЭКСПЕРТНЫХ СИСТЕМ НА БАЗЕ АЛГОРИТМОВ НЕЧЕТКОГО ВЫВОДА НА ЯЗЫКЕ TYPESCRIPT

Ляпко А. М., Курочкин А. В.

Кафедра интеллектуальных систем, Факультет радиоп физики и компьютерных технологий,

Белорусский государственный университет

Минск, Республика Беларусь

E-mail: alex.v.kurochkin@gmail.com, andrey.lyapko@gmail.com

Данная работа рассматривает разработку библиотеки для создания экспертных систем на базе алгоритмов нечеткого вывода на языке TypeScript. Исследуются основные особенности работы с нечеткой логикой, рассматривается статическая типизация лингвистических переменных и термов в TypeScript, а также приводятся особенности программного интерфейса разработанной библиотеки.

ВВЕДЕНИЕ

Построение экспертных систем на базе нечеткой логики позволяет описать процесс принятия решения формально, в виде правил, на понятном человеку или эксперту языке. В отличие от популярного на сегодняшний день подхода с использованием машинного обучения с учителем, для построения таких экспертных систем не требуется обширная обучающая выборка, а процесс принятия решений является прозрачным [1, 2].

Для поддержки построения экспертных систем на базе нечеткого вывода на различных языках и платформах существуют встроенные или сторонние библиотеки, предоставляющие соответствующую функциональность. Тем не менее, существующие реализации для языка JavaScript являются крайне ограниченными по функциональности, несмотря на очень высокую популярность языка на сегодняшний день.

Целью этой работы является рассмотрение основных особенностей разработки такой библиотеки, а также рассмотрение основных особенностей разработанного программного интерфейса для задания лингвистических переменных, термов и базы правил системы нечеткого вывода, и реализации непосредственного нечеткого вывода по алгоритмам Мамдани.

I. СИСТЕМЫ НЕЧЕТКОГО ВЫВОДА

Экспертные системы на базе нечеткого вывода – математические модели, которые для моделирования процесса принятия решений используют базу правил, описанную в терминах классической логики высказываний, а также информацию об элементарных переменных и их возможных состояниях в терминах нечетких множеств. На основании базы правил отдельные элементарные высказывания могут быть определены на основании входных данных нечеткой системы и нечетких множеств, соответствующих термам этих высказываний. После этого на основании классического логического вывода строится математическая модель, которая связывает

полученные нечеткие множества с элементарными высказываниями о неизвестных переменных.

Лингвистическая переменная описывается названием, универсальным множеством его возможных значений, а также набором нечетких множеств над универсальным множеством, называемых термами. Поскольку каждый терм лингвистической переменной является нечетким множеством, для элементарного высказывания «переменная = терм» по конкретному значению из универсального множества может быть определена принадлежность (истинность) как значение от 0 до 1 [1].

Разработка и применение систем нечеткого вывода включает в себя ряд этапов, реализация которых выполняется на основе рассмотренных ранее положений нечеткой логики. К таким этапам относятся:

1. Формирование базы правил
2. Фаззификация входных данных
3. Агрегирование подусловий
4. Активизация подзаключений
5. Аккумуляирование заключений
6. Дефаззификация выходных переменных

Рассмотренные этапы нечеткого вывода могут быть реализованы неоднозначным образом: агрегирование может проводиться не только в базе нечеткой логики, активизация может проводиться различными методами нечеткой композиции, на этапе аккумуляции объединение можно провести отличным от max-объединения способом, дефаззификация также может проводиться различными методами. Таким образом, выбор конкретных способов реализации отдельных этапов нечеткого вывода определяет тот или иной алгоритм нечеткого вывода. В настоящее время остается открытым вопрос критериев и методов выбора алгоритма нечеткого вывода в зависимости от конкретной технической задачи [2].

II. РЕАЛИЗАЦИЯ БИБЛИОТЕКИ ДЛЯ ПОДДЕРЖКИ НЕЧЕТКОГО ВЫВОДА

Разработанная библиотека предоставляет программный интерфейс для задания входных и выходных лингвистических переменных и их

термов, формирования базы правил и нечеткого вывода.

Лингвистические переменные задаются в виде объекта с двумя полями для описания входных переменных (`input`) и выходных переменных (`outputs`). Сами же лингвистические переменные представлены в виде объекта, у которого в качестве ключа используется название переменной, в качестве значения – объект с его термами. Каждый объект с термами в качестве ключа использует название термина, а в качестве значения – его функцию принадлежности. Для представления объекта с лингвистическими переменными был использован универсальный объект (`ObjectMap<T>`). Таким образом, поля `input` и `output` имеют один и тот же тип `ObjectMap<ObjectMap<Function>>`. Для перечисления переменных и термов в соответствующем объекте используется функция `Object.keys` [3].

Функция активации отдельного термина задается не аналитически, а в виде полноценной функции одной переменной. Для удобства использования в рамках библиотеки предоставляются реализации типичных функций активации – треугольной, трапецевидной, ступенчатой, сигмоидальной и т.п. с различными параметрами. Кроме того, библиотека предоставляет набор функций высшего порядка для комбинирования базовых функций принадлежности с операциями инверсии, объединения и пересечения на основе \max -нормы. Кроме того, внешний интерфейс описания термина позволяет использовать в качестве функции активации произвольную реализацию, которая должна удовлетворять всем свойствам функции принадлежности.

База правил задается в виде массива элементов `Rule`, которые являются кортежами из условий `Condition` и заключений `Conclusion`. Условия и заключения, в свою очередь, являются нечеткими высказываниями `Statement`. Нечеткое высказывание содержит название нечеткой переменной и значение одного из её термов. С помощью строгой типизации на этапе компиляции подтверждается, что терм для указанной нечеткой переменной является одним из ключей в объявлении переменной. Также в заключении используется дополнительное поле с весовым коэффициентом правила. Данная структура позволяет удобно работать с данными на разных стадиях алгоритма; например, на стадии фаззификации и агрегации удобно пользоваться массивом условий из базы правил, которая использует модель `Rule`.

В рамках разрабатываемой библиотеки реализован алгоритм нечеткого вывода Мамдани. К основным особенностям можно отнести следующие:

- Фаззификация входных переменных осуществляется на основании термов входных

переменных и соответствующих им нечетких множеств;

- Агрегирование подусловий правил нечеткой продукции осуществляется при помощи нечеткой логической операции «И» на основании \min -нормы $A, B : T(A \cap B) = \min(T(A); T(B))$;
- Активизация подзаключений правил нечеткой продукции осуществляется методом \min -активизации $\mu(y) = \min(c; \mu(x))$, где $\mu(x)$ и c – соответственно функции принадлежности термов лингвистических переменных и степени истинности нечетких высказываний;
- Аккумуляция подзаключений правил нечеткой продукции проводится при помощи \max -объединения функций принадлежности $\forall x \in X : \mu_{AB}(x) = \max(\mu_A(x); \mu_B(x))$;
- Дефаззификация проводится методом центра тяжести или центра площади.

Для реализации нечеткого вывода по алгоритму Мамдани полученная нечеткая модель реализует метод `predict`, которому на вход передаются строгие значения входных переменных в виде объекта. В качестве возвращаемого значения выступает объект, полями которого являются названия выходных переменных, а значения – полученными дефаззифицированными выходными переменными.

ЗАКЛЮЧЕНИЕ

Разработка экспертных систем на базе нечеткого вывода позволяет формализовать экспертный процесс принятия решений и использовать полученную систему для решения различных задач. В работе рассмотрена реализация библиотеки для поддержки разработки экспертной системы для языка TypeScript и описаны основные особенности полученного программного интерфейса пользователя. Дальнейшие направления исследований включают возможность более тонкой настройки используемого метода нечеткого вывода, а также поддержка других алгоритмов нечеткого вывода и нейронечетких систем.

СПИСОК ЛИТЕРАТУРЫ

1. Куценко, Д. А. Представление и использование знаний в интеллектуальных САУ при помощи методов теории нечетких множеств / Д. А. Куценко // Вестник БГТУ им. В. Г. Шухова [Электронный ресурс]. – 2015. – № 4. – Режим доступа: <http://nrsu.bstu.ru/chap27.html> – Дата доступа: 17.05.2019.
2. Хаптахбаева, Н. Б. Введение в теорию нечетких множеств: учебное пособие / Н. Б. Хаптахбаева, С. В. Дамбаева, Н. Н. Аюшеева. – Улан-Удэ: Изд-во ВСГТУ, 2004. – Ч.1. – 68 С.
3. Fenton, S. Pro TypeScript: Application-Scale JavaScript Development / S. Fenton, –Apress, 2014. –1st ed. –248 P.

ОНТОЛОГИЧЕСКИЙ ПОДХОД К ОБРАБОТКЕ ТЕКСТОВ КИТАЙСКОГО ЯЗЫКА

Цянь Лунвэй, Ли Вэньцзу
Кафедра интеллектуальных информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: qianlw1226@gmail.com, wzzggml@gmail.com

Как типичный представитель аналитического языка, китайский язык сильно отличается от других языков. Не существует границ между словами в китайских языках, и нет морфологических изменений про слова. Этими характеристиками китайского языка обусловлены существенные различия в технологиях обработки китайского языка по сравнению с другими языками. В данной работе кратко рассмотрены основные трудности обработки китайского языка и проанализированы существующие методы обработки.

ВВЕДЕНИЕ

Разнообразие естественного языка и его неоднозначность создает большие трудности в обработке естественного языка. Основными языками в мире являются английский, китайский, русский, французский и арабский. Каждый язык имеет свои особенности в лексическом и синтаксическом аспектах, такие как, например, морфологические изменения в английском и изменение падежами в русском языке, что приводит к большим различиям в методах обработки языка. В китайском языке нет естественных границ для разделения каждого слова [1]. У каждого китайского слова есть только одна форма, и нет изменяемых форм, таких как множественное число, часть речи, форма времени и т. д. Далее кратко рассмотрим эти трудности и методы их преодоления в технологии обработки китайского языка.

I. СЕГМЕНТАЦИЯ ТЕКСТОВ КИТАЙСКОГО ЯЗЫКА

Сегментация текста – процесс объединения последовательностей иероглифов в последовательности слов в соответствии с определенными спецификациями. В китайском предложении не каждый иероглиф имеет значение, но слово выражает определенное значение. В английском и русском языке между словами пробелы используются в качестве естественных разделителей слов. В китайском языке можно легко разделить текст на отдельные иероглифы, предложения и абзацы, однако на уровне слов в китайском языке нет формального разделителя. Таким образом, в сегментации китайских текстов существует проблема идентификации неоднозначности.

Например, если есть фраза вида ABC, то можно выделить слово AB, но также можно выделить слово BC:

- 老板/有意/见他 (босс встречает его с некоторой целью);
- 老板/对/他/有/意见 (у босса есть мнение о нём).

В свою очередь, слово вида AB может рассматриваться как одно слово, но может быть разделено на два слова:

- 语言学/是/以/人类/语言/为/研究/对象/的/学科 (Лингвистика является предметом, который изучает человеческий язык);
- 其它/语言/学/起来/很难 (Другие языки трудно выучить).

В процессе сегментации китайского текста, помимо идентификации неоднозначности, существует еще одна проблема – идентификация новых слов. Новые слова – это слова, которые на данный момент нет в словаре. Новые слова в целом подразделяются на две категории: 1. Появляющиеся общие слова или технические термины и т. д. 2. Собственные существительные, такие как китайские имена, иностранные переведенные названия, топонимы. В английском и русском языках эти слова могут быть легко оценены по разнице между заглавными и строчными буквами. Из-за отсутствия морфологических изменений в китайском языке возникает проблема идентификации новых слов.

В настоящее время основные методы сегментации на слова делятся на три категории [2].

Сегментация слов на основе словаря. Для набора китайских символов происходит сопоставление его фрагментов с терминами в «достаточно большом» машинном словаре в соответствии с определенной стратегией.

Сегментация слов на основе статистики. Слова состоят из китайских иероглифов, то есть слова представляют собой комбинацию устойчивых китайских иероглифов. Частота комбинаций смежных китайских символов в фразах может быть подсчитана и может отражать вероятность того, что смежные китайские символы могут быть одним словом.

Сегментация слов на основе понимания является одним из методов на основе статистики. С появлением моделей глубокого обучения исследователи применяют векторные мо-

дели слов, которые содержат синтаксическую и семантическую информацию.

Рассмотренные методы реализуют сегментацию китайских текстов и повышают точность сегментации слов в разных аспектах, но не являются идеальными для идентификации неоднозначности. При улучшении точности эффективность сегментации китайских текстов невелика.

II. ОПРЕДЕЛЕНИЕ ЧАСТЕЙ РЕЧИ В КИТАЙСКОМ ЯЗЫКЕ

Одна и та же часть речи может выполнять множество синтаксических ролей без морфологических изменений в китайском языке. В английском и русском словаре, кроме толкования слова и примеров использования каждого слова, приводится также часть речи для каждого слова. Даже без просмотра словаря, часть речи определенных слов можно различить по морфологическим характеристикам слова. Например, в русском языке есть суффикс, который точно определяет часть речи. В китайском языке при определении части речи могут возникать трудности:

这篇文章的重点是第三段 (Внимание в этой статье сфокусировано на третий абзац).

Слово «重点» является существительным.

这篇文章重点讲述机器翻译问题 (В этой статье внимание уделено вопросам машинного перевода).

Слово «重点» является наречием.

В настоящее время методы определения частей речи (частеречевого тегирования) делятся на две категории [3]. **Метод на основе правил** был самым ранним методом, использованным для маркировки части речи. По контексту и отношению в словосочетании между многофункциональным словом и другим словом строятся правила устранения неоднозначности при определении части речи. Ранее правила маркировки части речи обычно создавались вручную, однако по мере увеличения размера аннотированного корпуса текстов извлечение правил вручную становится непрактичным. Для решения данной задачи предложен **метод на основе статистики**. На основе заданной последовательности слов с соответствующей им маркировкой частей речи может быть определена наиболее вероятная часть речи для следующего слова.

Метод на основе статистики позволяет получить более высокую точность маркировки части речи, но требует высокого качества корпуса.

III. ОНТОЛОГИЧЕСКИЙ ПОДХОД

Классические методы имеют высокую скорость в обработке китайского текста, однако точность обработки не может быть сильно улучшена без использования новых методов. Мы предлагаем использовать комбинацию классических методов сегментации и частеречевого тегирования

и методов на основе онтологий. В данной работе предлагается онтологический подход, разработанный на основе технологии OSTIS [4]. Технология OSTIS ориентирована на разработку совместимых компьютерных систем, управляемых знаниями. Одним из ключевых принципов OSTIS технология является использование онтологического подхода.

При сегментации китайского текста комбинация методов на основе словаря и на основе онтологий может сочетать в себе преимущества метода на основе словаря – высокую скорость сегментации, и использовать метод на основе онтологии для повышения точности. При использовании такого комбинированного подхода результаты сегментации сопоставляются с онтологией семантики китайского языка. Это способствует повышению качества идентификации неоднозначности и ее устранения.

При частеречевом тегировании в соответствии с характеристиками китайской грамматики создается онтология синтаксиса китайского языка. Онтология синтаксиса китайского языка используется для проверки результатов выделения частей речи в китайских предложениях. Это позволяет уменьшить зависимость статистических методов от корпуса.

ЗАКЛЮЧЕНИЕ

Китайский язык является языком, в котором порядок слов и функциональные слова используются для выражения грамматического значения. В настоящее время большинство современных моделей обработки естественного языка основаны на исследованиях английского языка. Из-за разнообразия языков необходимо учитывать особенности китайского синтаксиса и семантики, что делает актуальным построение онтологии синтаксиса и семантики китайского языка, которые могут в дальнейшем использоваться для устранения неоднозначности при обработке текстов. Особенно важной задачей становится разработка специальной модели обработки китайского языка.

СПИСОК ЛИТЕРАТУРЫ

1. Huang, C. N. Chinese Word Segmentation: A Decade Review /C. N. Huang. // Journal of Chinese Information Processing. – 2007. – vol.21, № 03. – p. 8–19.
2. Long, S. Q. Zhao, Z. G. Overview on Chinese Segmentation Algorithm /S. Q. Long, Z. W. Zhao. // Computer Knowledge and Technology. –2009. – vol.5, № 10. –p. 2605–2607.
3. Liu, H. M. Research on Chinese parts of speech tagging and POS guessing over unknown word / H. M. Liu. – Nanjing Normal University. 2015.
4. Голенков, В. В. Гулякина, Н. А. Проект открытой семантической технологии компонентного проектирования интеллектуальных систем. Часть 1: Принципы создания. /В. В. Голенков, Н. А. Гулякина // Онтология проектирования. – 2014. № 1. – с.42–64.

ПРОГРАММНАЯ ПОДДЕРЖКА БАЗЫ ЗНАНИЙ В СИСТЕМЕ ЛОГИЧЕСКОГО ПРОЕКТИРОВАНИЯ FLC-2

Романов В. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси
Минск, Республика Беларусь
E-mail: rom@newman.bas-net.by

Предлагается краткое описание подсистемы ведения базы знаний, используемой для системы логического проектирования дискретных устройств FLC-2. Данная подсистема позволяет эксперту подготавливать разнообразные сценарии проектирования без изменения самой САПР.

ВВЕДЕНИЕ

На современном этапе развития САПР логического проектирования дискретных устройств, реализуемых на базе заказных цифровых СБИС, масштабы решаемых ими задач таковы, что интеллектуальная поддержка процессов проектирования становится необходимой. В работе [1] был предложен подход к организации логического проектирования дискретных устройств на основе продукционно-фреймовой модели представления знаний. Там же была описана и программная система FLC, реализующая указанный подход.

В настоящее время эта система получила дальнейшее развитие и, выступая под наименованием FLC-2, отличается от своей предшественницы кроссплатформенной реализацией, основанной на использовании языка программирования C++ и библиотеки Qt [2]. Вместе с тем, существенно поменялся и состав входящих в ядро САПР программных модулей, реализующих проектные процедуры.

I. ПРОЦЕСС ПРОЕКТИРОВАНИЯ И ОБЪЕКТЫ В БАЗЕ ЗНАНИЙ

Собственно процесс логического проектирования, реализуемый в рамках системы FLC-2, можно трактовать как процесс изменения описания схемы, представленного на некотором языке проектирования. Переход из одного состояния к другому осуществляется под воздействием проектных процедур, реализуемых заранее подготовленными программными модулями. Обычно функционально-структурные описания схемы носят иерархический характер, листовые элементы иерархии на заключительном этапе проектирования соответствуют элементам используемой технологической библиотеки, на которой схема реализуется. Проектная процедура может применяться как ко всей схеме, так и к отдельным ветвям дерева (блокам иерархии). В силу указанного, на каждом шаге проектирования решается вопрос о том, какую из доступных проектных процедур применить в текущий момент времени и к какому из существующих блоков иерархии. В некоторых случаях ответ на поставленный вопрос может быть получен на основании автома-

тизированного анализа текущего состояния проектирования. Фактически речь идет о подготовленном сценарии обработки проекта (*стратегии* в терминах системы FLC-2). Сценарий представляется цепочкой команд (*продукций*), задающих требуемое для срабатывания продукции условие и описывающих действие, которое будет выполнено при срабатывании. Условие продукции представляется предикатом, в качестве аргументов которого выступают *атрибуты*, определяющие существенные свойства проектируемой схемы или самого процесса проектирования. В заключении продукции устанавливается выполняемое действие, чаще всего *программный модуль*, реализующий заказанную проектную процедуру.

Таким образом, в базе знаний (БЗ) определены такие типы объектов, как стратегии, продукции, программные модули и атрибуты. Кроме них оказывается удобным размещать в БЗ *меню*, определяющее возможности обслуживания пользователя с точки зрения доступных проектных процедур и целых стратегий, а также *фреймы* и *слоты*, представляющие собой средства классификации и группирования атрибутов, используемые в процессе эксплуатации FLC-2.

II. ПОДСИСТЕМА ОБСЛУЖИВАНИЯ БАЗЫ ЗНАНИЙ

Ведение БЗ в системе FLC-2 возлагается на эксперта – специалиста в рассматриваемой проблемной области, с одной стороны, и знакомого с организацией применяемого программного обеспечения, с другой. Для автоматизации процессов управления БЗ разработана отдельная подсистема ESS-2 (Expert SubSystem), являющаяся развитием ранее используемой подсистемы, описанной в [1].

Используемая в FLC-2 БЗ представлена одним файлом в формате JSON [3]. Этого формата оказывается вполне достаточно для представления информации, обладающей древовидной структурой, где на первом уровне иерархии располагаются типы хранимых объектов, а далее все сводится либо к парам «имя»–«значение», либо к векторному представлению «имя»–«перечень значений». Применение фор-

мата JSON позволяет воспользоваться готовым программным обеспечением, реализующем его обслуживание.

Эксплуатация подсистемы ESS-2 осуществляется в диалоговом режиме. Основное рабочее окно ESS-2 после открытия файла БЗ представлено на рис.1. Пользователь в меню выбирает интересующий его тип данных, в соответствии с чем открывается специальное диалоговое окно определения характеристик объекта и его значений, используемых по умолчанию. Пример такого диалога для работы с программными модулями представлен на рис.2. Часть функционала подсистемы может быть использована путем нажатия кнопок, размещенных на ее основном окне возле представленных на нем списков объектов.

Таким образом можно заказать добавление нового, изменение или удаление текущего объекта, а также изменение порядка следования объектов (продукций отдельной стратегии).

СПИСОК ЛИТЕРАТУРЫ

1. Бибило, П. Н. Логическое проектирование дискретных устройств с использованием продукционно-фреймовой модели представления знаний / П. Н. Бибило, В. И. Романов / – Минск: Беларус. навука, 2011. – 279 с.
2. Шлее, М. Qt 5.3. Профессиональное программирование на C++ / М. Шлее // – СПб.: БХВ-Петербург, 2015. – 928 с.
3. JSON [Electronic resource] / Mode of access: <https://ru.wikipedia.org/wiki/JSON>. Date of access: 09.10.2019.

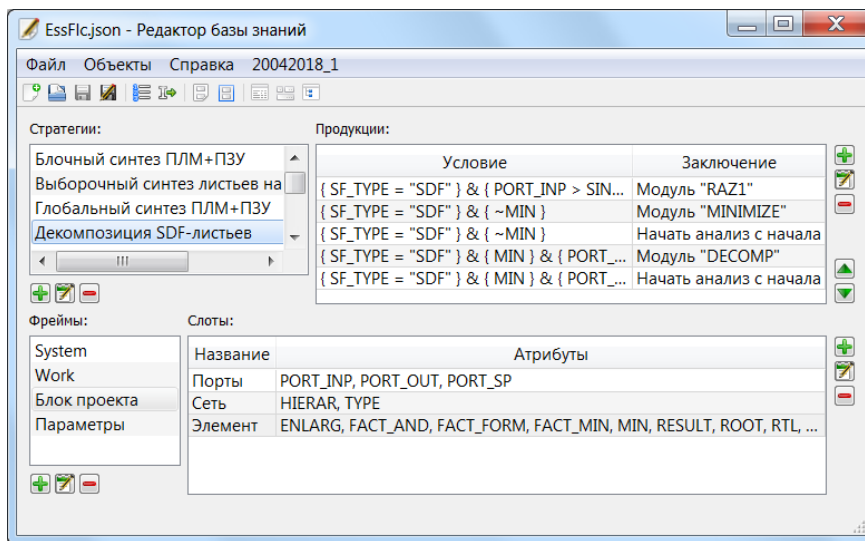


Рис. 1 – Основное рабочее окно подсистемы ESS-2

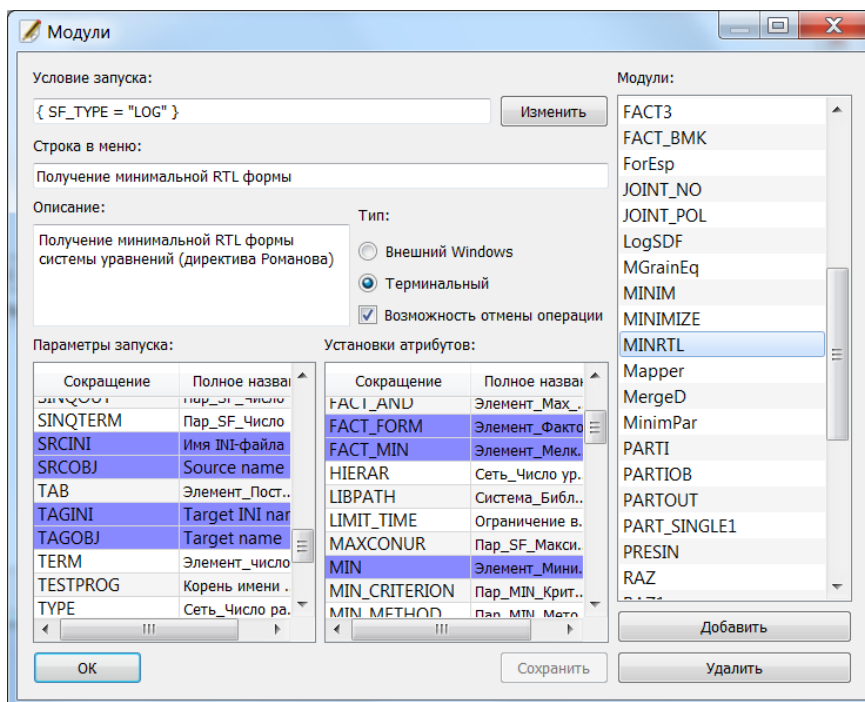


Рис. 2 – Диалоговая панель для установки свойств программного модуля

АВТОМАТИЧЕСКОЕ РАЗРЕШЕНИЕ КОНФЛИКТОВ ПРИ СИНХРОНИЗАЦИИ РЕПЛИК В РАСПРЕДЕЛЁННОЙ СИСТЕМЕ

Сафин К. В.

Кафедра инженерной психологии и эргономики,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: safinkaren7@gmail.com

В данной работе излагаются особенности бесконфликтных реплицированных типов данных, а также описаны технологии операционных преобразований для разрешения конфликтов в приложениях совместного редактирования

ВВЕДЕНИЕ

Совместная работа с такими приложениями как календарь, сервис заметок типа Evernote или сервис совместного редактирования документов, таких как Google Doc, сложна с технической точки зрения, потому что несколько людей могут вносить различные изменения в одни и те же данные в практически одинаковые моменты времени. Так как доставка данных через интернет не осуществляется мгновенно, то для имитации мгновенного отклика в каждый момент времени клиент работает с локальной версией (репликой) редактируемого документа, которая может отличаться от версий других участников. Основная проблема в этом случае – обеспечить консистентность локальных версий.

Существует два основных подхода к решению настоящей проблемы, выбор которых зависит от особенностей реплицируемых данных. *Операционные трансформации* – подход, применяемый обычно при работе с текстом и состоящий в безконфликтном преобразовании слияния данных. Имеет более высокие требования к серверу, более сложные и менее стабильные алгоритмы [3]. Второй подход состоит в использовании *бесконфликтных реплицируемых типов данных*, особенности которых рассмотрены ниже.

1. ОПЕРАЦИОННЫЕ ТРАНСФОРМАЦИИ

Рассмотрим один из алгоритмов операционного трансформирования, применяемый при совместном редактировании текста [5].

Допустим, что все изменения отправляются на сервер в виде набора команд:

$$(\ell_1 \rightarrow \ell_2)[c_1, c_2, c_3, \dots].$$

где:

- ℓ_1 : длина документа до редактирования;
- ℓ_2 : длина документа после редактирования.
- c_1, c_2, c_3, \dots – параметры описания документа после редактирования.

Если c_i – число (или диапазон чисел), то это порядковый номер (номера) символа исходного документа (далее индексы), который останется

после редактирования, а если c_i – символ (или строка), то это вставка новых символов.

Пример:

В исходную строку “” приходят изменения $(0 \rightarrow 2)[“Hi”]$, получается “Hi”. После этого приходят изменения $(2 \rightarrow 5)[1, “ello”]$, получаем “Hello”. Итоговый документ формируется как последовательность таких изменений, применённых по порядку к исходному состоянию общего документа (здесь и далее обозначаемое как X).

Далее несколько реплик могут быть объединены в общий документ путем так называемой операции *слияния* $m(A, B)$, где A и B это команды редактирования от двух разных реплик: $A = (n \rightarrow n_a)[\dots]$ и $B = (n \rightarrow n_b)[\dots]$ соответственно. Операция слияния обязана обладать свойством коммутативности:

$$m(A, B) = m(B, A).$$

Заметим, что для клиента с изменениями A , получившему изменения B , нет смысла вычислять $m(A, B)$, так как $m(A, B)$ применяется к X , а у A текущее состояние $A(X)$. В этом случае, нам нужно вычислить A' и B' , такие, что:

$$B'(A(X)) = A'(B(X)) = m(A, B)(X).$$

Определим функцию вычисления A' и B' :

$$f(A, B)(A) = f(B, A)(B) = m(A, B) = m(B, A)$$

где $m(A, B) = m(B, A) = A(B') = B(A')$. Алгоритм построения $f(A, B)$ состоит из следующих шагов [3]:

- вставки в команде редактирования A становятся индексами в $f(A, B)$;
- вставка в команде редактирования B становятся вставкой в $f(A, B)$;
- совпадающие индексы в A и B переносятся в $f(A, B)$.

Пример:

Дана строка "baseball". Две распределенные системы параллельно вносят свои поправки в свои реплики этой строки. Вычислим результат слияния реплик путем операциональных трансформаций.

$$X = (0 \rightarrow 8)[“baseball”];$$

$A = (8 \rightarrow 5)[0 - 1, "si", 7](A = "basil");$
 $B = (8 \rightarrow 5)[0, "e", 6, "ow"](B = "below").$

Вычисляем:

$A' = f(B, A) = (5 \rightarrow 6)[0, 1, "si", 3, 4];$
 $B' = f(A, B) = (5 \rightarrow 6)[0, "e", 2, 3, "ow"];$
 $m(A, B) = m(B, A) = (8 \rightarrow 6)[0, "esiow"].$
 $m(A, B)(X) = (8 \rightarrow 6)[0, "e", 2, "iow"]$

Таким образом, изначальная строка "baseball" путем совместного редактирования двух распределенных систем, и последующего слияния реплик "basil" и "below" превратилась в строку "besilow".

II. БЕСКОНФЛИКТНЫЕ РЕПЛИЦИРУЕМЫЕ ТИПЫ ДАННЫХ (БРТД)

БРТД - это особый вид данных обладающий строгой конечной консистентностью. Это значит, что эти данные обязаны обладать тремя свойствами [2]:

- идемпотентность: $a \vee a = a$;
- ассоциативность: $(a \vee b) \vee c = a \vee (b \vee c)$;
- коммутативность: $a \vee b = b \vee a$.

Из известных и описанных, стоит отметить следующие [1]:

- G-Counter: (grow-only) монотонно увеличивающийся счётчик;
- PN-Counter: (positive-negative) счётчик, который можно уменьшать;
- LWW-Register: (last-writer-wins) регистр с принципом "последняя запись приоритетнее";
- MV-Register: (multi-value) регистр с несколькими значениями;
- G-Set: множество элементов без удаления;
- 2P-Set: множество элементов с приоритетным удалением;
- PN-Set: множество, которое использует счётчик операций включения-удаления;
- LWW-Set: множество с приоритетом времени операции;
- OR-Set: (observed-remove) множество с идентификаторами.
- Ordered-List: упорядоченный список.
- Graphs: структуры, описывающие вершины и связи в графах, основанные на списках.

Рассмотрим правило манипуляции упорядоченным списком, что бы он считался списком, обладающим строгой конечной консистентностью.

Пусть исходное состояние общего списка X , это список символов "HELLO". Присвоим каждому элементу списка парный идентификатор. Первый элемент идентификатора это порядковый номер элемента а второй - идентификатор списка: " $H_{0x} E_{1x} L_{2x} O_{3x}$ ". При создании реплики, список копируется без изменений.

Допустим, что все изменения отправляются на сервер в виде набора команд из следующего списка возможных:

- для инициализации списка: (" $char$ ") $[id]$; где id - идентификатор первого символа.
- для вставки символа: (" $char$ ") $[id_1, id_2]$; где id_1 - идентификатор элемента перед вставляемым элементом а id_2 - идентификатор вставляемого элемента. Новый идентификатор должен иметь порядковый номер следующий за последним.
- для удаления символа: $[id]$. где id - идентификатор удаляемого элемента.

Например строка "НО" заменяется на "НИ", путем выполнения следующих команд:

$$H_{0x}O_{1x} \rightarrow [1x] \rightarrow (I)[0x, 2x] = H_{0x}I_{2x}$$

Пример для реплик A и B :

1. $h_{0x}e_{1x}l_{2x}o_{3x} \rightarrow (l)[2x, 4a] \rightarrow h_{0x}e_{1x}l_{2x}l_{4x}o_{3x}$;
2. $h_{0x}e_{1x}l_{2x}o_{3x} \rightarrow (!)[3x, 4b] \rightarrow h_{0x}e_{1x}l_{2x}o_{3x}!_{4b}$.

Далее рассмотрим правило объединения двух конфликтующих реплик A и B :

1. $a_{0x}b_{1x}c_{2x} \rightarrow (x)[0x, 3a] \rightarrow a_{0x}x_{3a}b_{1x}c_{2x}$;
2. $a_{0x}b_{1x}c_{2x} \rightarrow (p)[0x, 3b] \rightarrow a_{0x}p_{3b}b_{1x}c_{2x}$.

Требуется вычислить результат слияния $m(A, B)$. Операция слияния БРТД обладает свойством коммутативности как и в случае с операционными трансформациями.

$$m(A, B) = m(B, A).$$

Предлагается следующий алгоритм слияния [4]:

При слиянии реплики B в A используются те же команды модификации что и при $B(X)$ за исключением операции вставки. После этого проверяется что бы идентификатор элемента id_i , следующего за элементом id_1 не был больше чем id_2 . В случае если он больше, повторяем проверку для id_{i+1} . В итоге, найдя такой id_{i+n} , где $id_{i+n} < id_2$, вставляем элемент за id_{i+n} . В случае если такой элемент не найден, вставляем элемент в конце списка. Рассмотрим на примере:

1. $A = a_{0x}x_{3a}b_{1x}c_{2x}, A(X) = (x)[0x, 3a];$
2. $B = a_{0x}p_{3b}b_{1x}c_{2x}, B(X) = (p)[0x, 3b].$
 - $m(A, B) = a_{0x}p_{3b}b_{1x}c_{2x} \rightarrow (x)[0x, 3a] = a_{0x}p_{3b}x_{3a}b_{1x}c_{2x}$ ($3a < 3b$, следовательно, вставляем x после $3b$)
 - $m(B, A) = a_{0x}x_{3a}b_{1x}c_{2x} \rightarrow (p)[0x, 3b] = a_{0x}p_{3b}x_{3a}b_{1x}c_{2x}$

СПИСОК ЛИТЕРАТУРЫ

1. A comprehensive study of Convergent and Commutative Replicated Data Types / M. Shapiro N. Preguiça C. Baquero M. Zawirski - 2011. - Vol. 46, С. 4-40.
2. Key-CRDT Stores / Nuno Manuel Ribeiro Preguiça. - Junho., 2012. - С. 23-40.
3. Convergent and Commutative Replicated Data Types / M. Shapiro, C. Baquer - 2011. - С. 72-84.
4. Etherpad and EasySync Technical Manual / P. Martischka. - Munich, 2011. - С. 2-9.
5. Operational Transformation in Real-Time Group Editors: Issues, Algorithms, and Achievements / C. Sun E. Clarence - School of Computing and Information Technology - Brisbane, Australia, 2018. - С. 1-9.

ИСПОЛЬЗОВАНИЕ ДИСТИЛЛЯЦИИ МЕЖДУ СЛОЯМИ НЕЙРОННОЙ СЕТИ ДЛЯ УЛУЧШЕНИЯ КАЧЕСТВА СЕГМЕНТАЦИИ ДОРОЖНОЙ РАЗМЕТКИ

Сиротко С. И., Фисько Д. В.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: sergeyis@bsuir.by, dmitryfisko@gmail.com

Обучение нейронных сетей для распознавания дорожной разметки трудоёмкий процесс из-за разряженной структуры целевого класса. Без использования дополнительных приёмов при обучении модели, полученные модели зачастую сильно ошибаются в трудных условиях: слабом или слишком насыщенном освещении, при наличии изгибов дорог, объектов на дороге и т.д. Чтобы улучшить качество сегментации, предлагается в ходе обучения модели, после изначального предобучения, дополнить функцию потерь дополнительным слагаемым - метрикой схожести карт активации между последовательными слоями нейронной сети.

ВВЕДЕНИЕ

Распознавание дорожной разметки необходимо для понимания обстановки участниками транспортного движения, а также локализации положения транспортного средства на дорожном покрытии. Применение автоматизированного алгоритма для распознавания дорожной разметки затрудняется всей вариативностью возможного окружения: наличие машин вокруг, колебания освещения и плохих погодных условий, стёртой или полностью отсутствующей разметки.

Зачастую задачу сегментации дорожной разметки сводят к постановке в виде классической задачи сегментации изображения, используя нейросетевые модели с полносвязанными архитектурами. Обучение данных моделей сильно зависит от количества сигнала предсказываемого класса. Т.к. в обучающей выборке линии дорожной разметки зачастую представлены в виде тонких линий, то количество пикселей разметки сильно меньше, чем количество пикселей фона, что сильно затрудняет процесс обучения модели.

I. ОБЗОР СУЩЕСТВУЮЩИХ РЕШЕНИЙ

Один из способов преодоления этого ограничения заключается в искусственном увеличении размеров дорожной разметки в обучающей выборке, но использование данного решения вносит неточности в качестве локализации разметки. Также в литературе были предложены методы многоклассовой сегментации - Multi Task Learning [2] и установка дополнительных соединений между слоями - Message Passing [3]. Метод MLT требует дополнительных усилий при сборе обучающей выборки, потому что помимо необходимого класса дорожной разметки, должны быть существенно представлены области объектов других классов. ML метод работает через добавление дополнительных соединений между слоями нейронной сети и значительно (на 35%)

замедляет производительность вычислительного графа нейронной сети.

II. ДИСТИЛЛЯЦИЯ КАРТ АКТИВАЦИЙ

Предлагаемый метод заключается в добавление нового штрафа в функцию потерь. Тем самым метод не будет затрагивать производительность нейронной сети при непосредственном использовании, а будет увеличивать вычислительную сложность только во время обучения. Цель добавления нового штрафа в функцию потерь заключается в следующем - стимулировать карты активаций нижних слоёв нейронной сети мимикрировать под карты активации верхних слоёв нейронной сети. Процесс обучения мимикрированию называется дистилляцией.

Интуиция этого метода исходит из следующего наблюдения. После предобучения нейронной сети, карты активаций в слоях нейронной сети накапливают достаточное количество разнообразной контекстной информации, которая позволяет визуальной определять разметку дороги. Эту информацию стоит усилить, через добавления дистилляции между соседними слоями. Т.е. карты активации первого слоя должны стремиться предсказывать карты активации второго слоя, а карты активации второго слоя предсказывают карты активации третьего слоя и т.д. Тем самым слои нижнего уровня стремятся извлекать больше контекстной информации.

Как показали эксперименты из таблицы 1, данный шаг даёт дополнительную информацию нейронной сети при обучении и позволяет при использовании нейросетевых моделей с меньшим количеством параметров получать аналогичное большим нейросетевым архитектурам показатели целевых метрик, например по сравнению с архитектурой модели SCNN [3].

В силу неполной изученности метода, его эффективное применение на данный момент достигнуто только на небольших нейросетевых моделях, таких как: ENet [4], ResNet-18 и ResNet-34

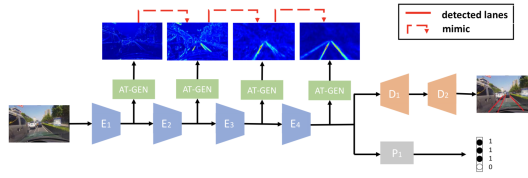


Рис. 1 – Архитектура сети ENet с добавлением дистилляции между блоками энкодеров

[5]. Более детально распишем методологию работы метода на примере модели с архитектурой ENet.

III. МЕТОДОЛОГИЯ

Добавление дистилляции между слоями нейронной сети может происходить в произвольный момент обучения нейронной сети, но добавление должно происходить когда нейронная сеть достаточно обучилась. На примере архитектуры ENet (рисунок 1), добавление штрафа на точность дистилляции происходит следующим образом: между блоками E1-E4 добавляется вычитывание агрегата над активациями нейронной сети и берётся сумма евклидова расстояния между агрегатами последовательных слоёв. За генерирование агрегата на основе карт активации слоя, отвечает блок называемый AT-GEN. Обозначим блок AT-GEN как $\Psi(\cdot)$. Он принимает на вход матрицу активаций m по счёту слоя - матрицу A_m и преобразует её по следующей формуле:

$$\Psi(A_m) = \Phi(\mathcal{B}(\mathcal{G}_{sum}^2(A_m))), \quad (1)$$

где преобразование $\mathcal{G}(\cdot)$ суммирует поканально возведённые в квадрат значения карт активаций слоя, преобразование $\mathcal{B}(\cdot)$ масштабирует размер полученного агрегата через билинейную дискретизацию, а преобразование $\Phi(\cdot)$ является поканальным softmax преобразованием.

Используя данное для \mathcal{G} определение, итоговый вид функция штрафа основанной на точности дистилляции между последовательными слоями нейронной сети будет иметь следующий вид:

$$\begin{aligned} \mathcal{L}_{distill}(A_m, A_{m+1}) \\ = \sum_{m=1}^{M-1} (L_2(\Psi(A_m), \Psi(A_{m+1}))), \end{aligned} \quad (2)$$

где A_m - это матрица активаций слоя m , M это количество слоёв на которых добавляется дистилляция по картам активаций, L_2 это функция евклидова расстояния, а Ψ это функция описанная ранее в формуле 1.

IV. ЭКСПЕРИМЕНТЫ

Для сравнения результатов был выбран метод называемый Deep Supervision [6]. Он заключается в том, чтобы карты активации предсказывали аналогичные целевому классу силуэты. Также тестирование метода проводилось на различных обучающих выборках: CuLane [3] и BDD100 [7].

Таблица 1 – Сравнение результатов

Алгоритм	TuSimple	BDD100K	
	Accuracy	Accuracy	IoU
ENet	93.02%	34.12%	14.64
ENet-Deep	94.69%	35.61%	15.38
ENet-SAD	96.64%	36.56%	16.02
ResNet-34	92.84%	34.12%	12.24
ResNet-34-Deep	94.52%	31.72%	13.59
ResNet-34-SAD	96.24%	32.68%	14.56

ЗАКЛЮЧЕНИЕ

Из полученной таблицы 1 делаем вывод, что добавление дистилляции между слоями нейронной сети для решения задачи распознавания дорожной разметки оправдано как минимум для небольших нейросетевых архитектур. Применимость метода для нейросетевых архитектур большего размера предстоит изучить.

1. Learning Lightweight Lane Detection CNNs by Self Attention Distillation – ICCV 2019. – Режим доступа: <https://arxiv.org/abs/1908.00821>. – Дата доступа: 01.10.2019.
2. VPGNet: Vanishing Point Guided Network for Lane and Road Marking Detection and Recognition – arXiv.org, 2017. – Режим доступа: <https://arxiv.org/abs/1710.06288>. – Дата доступа: 03.10.2019.
3. Spatial as Deep: Spatial CNN for Traffic Scene Understanding – AAAI, 2018. – Режим доступа: <https://arxiv.org/abs/1712.06080>. – Дата доступа: 03.10.2019.
4. ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation – arXiv.org, 2016. – Режим доступа: <https://arxiv.org/abs/1606.02147>. – Дата доступа: 07.10.2019.
5. Deep Residual Learning for Image Recognition – arXiv.org, 2015. – Режим доступа: <https://arxiv.org/abs/1512.03385>. – Дата доступа: 07.10.2019.
6. Holistically-Nested Edge Detection – arXiv.org, 2015. – Режим доступа: <https://arxiv.org/abs/1504.06375>. – Дата доступа: 07.10.2019.
7. BDD100K: A Diverse Driving Video Database with Scalable Annotation Tooling – arXiv.org, 2018. – Режим доступа: <https://arxiv.org/abs/1805.04687>. – Дата доступа: 01.10.2019.

АЛГОРИТМЫ ДЕТЕКЦИИ ОБЪЕКТОВ ДЛЯ АНАЛИЗА ИЗОБРАЖЕНИЙ

Хурсов П. С., Искра Н. А.

Кафедра электронных вычислительных машин,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: xursovp01@gmail.com, niskra@bsuir.by

В данной статье рассматриваются актуальные подходы к решению задачи автоматической детекции объектов, проводится анализ их параметров работы и показателей эффективности, делаются выводы о возможности применения подобных алгоритмов в интеллектуальном анализе изображений.

ВВЕДЕНИЕ

В последние годы качественный скачок произошёл в области классификации изображений [1]. Дальнейшее развитие области анализа изображений происходит в двух направлениях: с одной стороны, это автоматическая детекция множества объектов с определением их меток и месторасположения [2], а с другой - семантическое описание, которое предполагает интерпретацию взаимосвязей объектов на изображении, например в виде текстового описания [3].

Решение задачи понимания и интерпретации изображения требует тесного взаимодействия этих направлений [4]. Так, например, на основании результатов автоматической детекции можно построить графовую модель, отражающую семантические связи между объектами [5].

В рамках данной работы мы рассматриваем актуальные подходы к автоматической детекции объектов как первого этапа в интеллектуальном анализе изображений [6].

I. ОСНОВНЫЕ ПОДХОДЫ К ДЕТЕКЦИИ ОБЪЕКТОВ

Среди всех алгоритмов детекции объектов можно выделить три актуальных подхода к решению данной задачи.

Первым и наиболее вычислительно затратным подходом является метод скользящего окна. Основой данного подхода является пошаговый обход всего изображения, при котором извлекается некоторый регион изображения и происходит попытка классификации объекта, который попал в этот регион.

Следующим подходом является использование так называемых “предложений регионов” (Region Proposals). В этом подходе используется комбинация из двух алгоритмов. Первый из них делает множество предположений (регионов, в которых, вероятно, содержится какой-либо объект), а второй осуществляет классификацию данного объекта.

Третьим подходом является “детекция за один проход” (Single Shot Detection), при котором локализация и классификация происходят одновременно, т.е. за один проход алгоритма.

II. МЕТОД СКОЛЬЗЯЩЕГО ОКНА

В этом классе алгоритмов из изображения последовательно извлекаются регионы прямоугольной формы. Так же для увеличения точности распознавания объектов разного размера может строиться пирамида изображений. После этого к каждому региону применяется классификатор. В зависимости от классификатора можно добиться либо высокой скорости выполнения (метод Виолы-Джонса, каскад Хаара, LBP каскады), либо высокой точности (различного вида нейронные сети).

III. REGION PROPOSALS

Множество алгоритмов использующее данный подход образуют семейство Regions with CNN (R-CNN [7]). Первой стадией является составление “предложений регионов”. Для этого может применяться как отдельный алгоритм (в том числе и метод скользящего окна), так и часть классификатора (в случае со сверточными нейронными сетями). Например, в R-CNN в качестве алгоритма генерации предположений использовался селективный поиск. Данный алгоритм достаточно затратный, поэтому в Faster R-CNN [8] от него отказались. На смену селективному поиску пришла сеть Region Proposals Network (RPN). Это позволило ускорить выполнение алгоритма, поскольку RPN использует общие с классификатором ресурсы.

Следующей стадией является классификация. Для этого выбираются регионы полученные на первой стадии и происходит их классификация. В R-CNN для этого использовалась последовательность CNN и машин опорных векторов (SVM). Вместе с классификацией происходит уточнение координат объекта. Поскольку последовательность CNN + SVM применялась к каждому предположению, а их в среднем порядка 2000 на изображение, то это очень затратно. Поэтому в следующей версии R-CNN (Fast R-CNN [9]) пришли к концепции “регионы интереса” (Regions of Interests). Основной идеей является обработка всего изображения обученной сверточной нейронной сетью. После этого на выходе получается уменьшенная карта высокоуров-

невых признаков, которые хорошо подходят для классификации объектов. После этого из карты извлекаются предсказанные на первой стадии участки (Regions of Interests) и классифицируются. Это позволило ускорить выполнение алгоритма почти в 25 раз и увеличить среднюю точность.

В Faster R-CNN карта признаков используется так же и для генерации Region Proposals, что позволяет увеличить точность и скорость выполнения. Так же к особенностям Faster R-CNN можно отнести то, что данный алгоритм предсказывает несколько ограничивающих регионов прямоугольной формы разного размера и соотношения сторон (2:1, 1:1, 1:2), что позволяет точнее выделять объект.

В таблице 1 приведено сравнение алгоритмов семейства R-CNN по скорости и точности среднего предсказания для классов, при условии, что детекция считается положительной, если отношение площади пересечения предсказанного региона и региона, содержащего объект, к суммарной площади двух регионов больше 0.5(mAP@0.5).

Таблица 1 – Сравнение алгоритмов семейства R-CNN

Алгоритм	mAP@0.5(%)	Время выполнения(с.)
R-CNN	62.4	50
Fast R-CNN	70.0	2
Faster R-CNN	78.8	0.2

IV. SINGLE SHOT DETECTION

Основной идеей данного подхода является попытка одновременно предсказывать и координаты объекта, и его класс. Для этого используются сверточные нейронные сети, поскольку они позволяют уменьшить размерность входного изображения без особой потери информации и с сохранением локализации признаков (Рис. 1). Основными архитектурами использующими данный подход являются YOLO (You only look once) [10] и SSD (Single shot detector) [11].

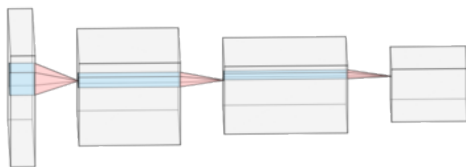


Рис. 1 – Сохранение локализации при свертке

YOLO делит изображение на сетку NxN. Если центр объекта попадает в некоторую ячейку сетки, то эта ячейка ответственна за детекцию данного объекта. Для этого YOLO использует обученную сверточную нейронную сеть.

SSD использует похожий подход, однако использует не только выходной слой сверточной

нейронной сети, но и множество промежуточных слоев для построения предсказаний. Так же SSD использует сверточные слои с ядрами разных размеров, что позволяет лучше распознавать объекты разной величины.

Оба подхода обеспечивают высокую скорость выполнения алгоритма и неплохую точность (Таблица 2).

Таблица 2 – Сравнение архитектур SSD

Алгоритм	mAP@0.5(%)	Время выполнения(мс.)
YOLO	63.7	22
YOLOv3	78.6	15
SSD300	79.6	21
SSD512	81.5	50

V. ВЫВОД

Проведя сравнение трех основных подходов, можно сделать вывод о целесообразности использования архитектуры SSD512 для детекции объектов в интеллектуальном анализе изображений, поскольку помимо высокой скорости выполнения она имеет неплохую точность. Однако если скорость исполнения не важна, то можно применить метод скользящего окна и пирамиды изображений, что обеспечит еще большую точность. Так же хорошим выбором может быть и Faster R-CNN, т. к. она позволяет определить не только местоположение объекта, но и сделать более точное предсказание региона, содержащего объект.

СПИСОК ЛИТЕРАТУРЫ

1. ImageNet Large Scale Visual Recognition Challenge /O. Russakovsky [et.al.] // IJCV – 2015. – Vol. 115, № 3. – P. 211–252.
2. Recent Advances in Convolutional Neural Networks / J. Gu [et.al.] // Pattern Recognition. – 2018. – Vol. 77. – P. 354–377.
3. A Comprehensive Survey of Deep Learning for Image Captioning / M. Z. Hossain [et.al.] // CoRR – 2018.
4. Johnson, J. DenseCap: Fully Convolutional Localization Networks for Dense Captioning / J. Johnson, A. Karpathy, L. Fei-Fei // CCVPR – 2016.
5. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations / R. Krishna [et.al.] // IJCV – 2016. – Vol. 123. – P. 32–73.
6. Iskra, N. Neural network based image understanding with ontological approach / N. Iskra, V. Iskra, M. Lukashevich // Open semantic technologies for intelligent systems (OSTIS-2019): materials of IX International.sc.-tech.conf. – Mn.: BSUIR, 2019. – Iss. 3. – P.113-122.
7. Rich feature hierarchies for accurate object detection and semantic segmentation /R. Girshick [et.al.] // CVPR – 2014. – P. 580–587.
8. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks / S. Ren [et.al.] // NIPS – 2015. – P. 91–99.
9. Fast R-CNN / R. Girshick // ICCV – 2015. – P. 1440–1448.
10. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon [et.al.] // CVPR – 2016. – P. 779–788.
11. SSD: Single Shot MultiBox Detector / W. Liu [et.al.] // ECCV – 2016. – P. 21–37.

Секция «Проектирование
встраиваемых систем»

Section «Embedded systems design»

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ АЛГОРИТМА ВЫДЕЛЕНИЯ ПОДСИСТЕМ «СВЯЗАННЫХ» БУЛЕВЫХ ФУНКЦИЙ

Бибило П. Н.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: bibilo@newman.bas-net.by

Исследуется алгоритм выделения связанных подсистем из системы булевых функций. «Связанность» функций заключается в наличии одинаковых частей в областях определения функций системы. Проведенные эксперименты показывают эффективность применения такого алгоритма при логической оптимизации системы булевых функций, которая осуществляется на основе разложения Шеннона с учетом возможности использования инверсий подфункций. Применение предложенного алгоритма позволяет во многих случаях увеличить быстродействие и уменьшить площадь комбинационных схем из библиотечных КМОП элементов.

ВВЕДЕНИЕ

Формулируется понятие связанности булевых функций (с учетом возможности инверсирования функций) и экспериментально исследуется программа, реализующая алгоритм [1] выделения связанных подсистем функций. Выделение связанных функций является одним из приемов логической оптимизации многоуровневых представлений систем функций. Выделение связанных функций позволяет объединить в одну подсистему те функции, которые целесообразно минимизировать на основе совместного BDDI-представления выделенной подсистемы функций. Проведенные эксперименты по выделению связанных подсистем функций показали, что данную процедуру целесообразно выполнять перед BDDI-оптимизацией [2], являющейся в настоящее время одним из эффективных методов логической минимизации при синтезе логических схем из библиотечных элементов.

I. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ И ФОРМЫ ЗАДАНИЯ СИСТЕМ БУЛЕВЫХ ФУНКЦИЙ

Пусть задана система булевых функций $f(x) = (f^1(x), \dots, f^m(x))$, через x обозначен вектор $x = (x_1, x_2, \dots, x_n)$ аргументов x_1, x_2, \dots, x_n . Характеристическим множеством $M_{f^i}^1$ компонентной функции $f^i(x)$ системы $f(x)$ называется множество наборов булева пространства, на которых функция $f^i(x)$ принимает единичное значение. Через $M_{f^i}^0$ обозначим множество наборов нулевых значений функции $f^i(x)$. Далее под связанностью булевых функций будет пониматься совпадение подобластей в областях определения функций. Обозначим через $|A|$ мощность множества A . Система функций $f(x)$ называется S_p -связанной, если $p_{\max}^n \geq p$, где

$$p_{\max}^n = \left| \bigcap_i^m M_{f^i}^1 \right| + \left| \bigcap_i^m M_{f^i}^0 \right|.$$

Число p_{\max}^n назовем весом связанности системы функций. Для одной булевой функции, зависящей от n переменных, вес связанности — это число, равное сумме мощностей множеств $M_{f^i}^1$, $M_{f^i}^0$, иначе говоря, это число 2^n элементов булева пространства, построенного над переменными булева вектора $x = (x_1, x_2, \dots, x_n)$. Элементами этого пространства являются n -компонентные наборы (векторы) x^* нулей и единиц. Мерой (долей) связанности $p^n(f^1, \dots, f^m)$ системы функций $f(x)$ назовем отношение

$$p^n(f^1, \dots, f^m) = \frac{p_{\max}^n}{2^n},$$

которое может быть задано в процентах. Очевидно, что мера связанности ограничена:

$$0 \leq p^n(f^1, \dots, f^m) \leq 1.$$

Введем в рассмотрение m -компонентный булев вектор $\alpha = (\alpha^1, \alpha^2, \dots, \alpha^m)$, называемый вектором поляризации для компонентных функций $f^i(x)$. Обозначим $\alpha^i = 1$, если рассматривается функция f^i , и $\alpha^i = 0$, если берется инверсия \bar{f}^i функции f^i . Система $f^i(x)$ функций называется S_p^α -связанной, если найдется хотя бы один вектор $\alpha = (\alpha^1, \alpha^2, \dots, \alpha^m)$ поляризации, для которого соответствующая данному вектору система функций является S_p -связанной. Понятие S_p^α -связанности соответствует связанности функций с точностью до инверсии компонентных функций системы.

II. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТА

Экспериментально исследовался модифицированный алгоритм выделения подсистем S_p^α -связанных функций, описанный в [1] и использующий задание систем функций таблицами истинности. Модификация состоит в том, что функции, не вошедшие ни в одну из подсистем связанных функций, не формируют отдельные подсистемы, а образуют одну отдельную «остаточную» подсистему. Исходная си-

Таблица 1 – Площади схем

Имя схемы	Синтез по BDDI исходной системы	Алгоритм выделения подсистем связанных функций				
		Мера связанности, %				
		10%	20%	30%	50%	70%
B2	192655	192331	192331	192331	192331	229120
B9	26081	26081	27281	28720	26901	26081
B12	18358	18213	18213	18911	16874	15881
BR1	23843	25752	25752	25752	25752	25752
BR2	21371	21371	21317	21317	21371	21371
IN0	94620	94620	94620	91836	96618	93521
INTB	272555	330208	284234	245408	310839	271445
M2	45114	45114	45114	45114	45114	49149
M3	52580	52580	52580	52580	52580	58378
M181	19469	18933	18347	18933	17276	16768
P82	19988	19736	19736	19418	20914	19402
ROOT	26109	23676	24976	28413	26059	23977
T3	17276	17276	17276	16288	17119	16729
TIAL	255988	215488	259152	222865	312726	290651
Z5XP1	18442	17159	18001	17147	18436	18442

стема и выделенные подсистемы связанных функций минимизировались в классе BDDI-представлений с помощью программы [2]. Исходные данные и методика синтеза схем в синтезаторе LeonardoSpectrum описаны в [1]. В табл. 1 представлены результаты эксперимента. Суммарная площадь всех элементов каждой из схем задавалась в условных единицах.

ЗАКЛЮЧЕНИЕ

Предложенный алгоритм в десяти случаях из пятнадцати позволил улучшить схемную реализацию по площади, и в тринадцати – по задержке (быстродействию). Его применение целесообразно при реализации исходных описаний, заданных таблицами истинности систем булевых

функций. Как показано в [1], для систем функций с большим числом переменных (несколько десятков), целесообразно использовать оценки меры связности выделяемых подсистем, используя BDD-представление исходной системы функций.

СПИСОК ЛИТЕРАТУРЫ

1. Бибило, П. Н. Разбиение системы булевых функций на подсистемы «связанных» функций / П. Н. Бибило // Известия РАН. Теория и системы управления. – 2019. – № 2. – С. 14–29.
2. Бибило, П. Н. Использование полиномов Жегалкина при минимизации многоуровневых представлений систем булевых функций на основе разложения Шеннона / П. Н. Бибило, Ю. Ю. Ланкевич // Программная инженерия. – 2017. – № 3. – С. 369–384.

ФОРМАТ POSIT КАК ФОРМАТ ПРЕДСТАВЛЕНИЯ ЧИСЛОВЫХ ЗНАЧЕНИЙ В НЕЙРОННЫХ СЕТЯХ НА УСТРОЙСТВАХ ПРОГРАММИРУЕМОЙ ЛОГИКИ

Боровский М. А., Грабко А. Н.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: maksim.borovskij@yandex.ru, softforlutshix@gmail.com

В данной работе рассматривается возможность применения формата POSIT для представления чисел в нейронных сетях, реализуемых на устройствах программируемой логики. FPGA имеют ограниченные ресурсы обработки данных, поэтому при разработке нейронных сетей на FPGA сталкиваются с вопросом представления чисел. Формат представления должен обеспечивать точность результатов в пределах принятых разработчиками норм и предоставлять максимальную производительность при работе с ними. Формат POSIT является кандидатом на роль одного из форматов в области.

ВВЕДЕНИЕ

Многие задачи не могут быть решены с помощью строгих алгоритмов, например, задачи распознавания, классификации, прогнозирования и т.д. При их решении широко используются нейронные сети (НС), которые позволяют обойти ограничения классических вычислительных систем.

В последние десятилетия были проведены многочисленные исследования аппаратных реализаций НС. НС хорошо поддаются представлению на языках описания аппаратуры, например, VHDL и Verilog, т. к. состоят из обособленных компонентов и явных связей между ними. Аппаратные НС дают преимущество в скорости обработки поступающих данных, что ведет к более быстрым процессам обучения и непосредственного использования НС.

При проектировании аппаратных НС важным является вопрос представления числовых значений, т. к. он напрямую влияет на объем необходимого хранилища, число логических элементов для представления АЛУ, ширину полос передачи данных и скорость их обработки.

Имеется некоторое число подходов к представлению числовых значений при реализации НС на FPGA:

1. числа с плавающей точкой (сложная реализация, малая производительность, большая точность);
2. числа с фиксированной точкой (легкая реализация, большая производительность, точность регулируется числом бит дробной части).

Появившийся в 2017 году и зарекомендовавший себя в некоторых областях формат POSIT может обеспечить необходимое соотношение точности и производительности в НС на FPGA.

I. ФОРМАТ POSIT

Предложенный Дж. Густафсоном арифметический подход предлагает три формы пред-

ставления чисел: Unum Type I, II и III. Unum Type III – последняя редакция формата, POSIT. Формат POSIT, как и float, имеет секции знака, экспоненты и дробной части, также обладает дополнительной секцией режима. Формат появился после принятого упрощения по отношению к Unum Type II: последний бит не используется в качестве значащего в дробной части и не используется как ubit (бит в конце дробной части, показывающий, что число является точным либо находится в интервале соседних вещественных чисел). Размеры секций режима, экспоненты и дробной части являются варьирующимися. Значение 0 в бите знака соответствует положительному числу, 1 – отрицательному. Режимом является число повторяющихся бит (нулей или единиц) в начале строки. Например, строка 0000 имеет 4 нуля, значит, режим равен 4, 001 - 2, 1110 - 3 и т.д. Режим имеет соответствующее значение k , которое используется для вычисления масштабирующего фактора. Значение k равно $-m$, если повторяющиеся биты нули, и $m - 1$, если повторяющиеся биты единицы, где m – количество повторяющихся бит. Например, для строки 0001 k равно -3, 1110 - 2 и т.д. Масштабирующий фактор равен $useed^k$, а $useed = 2^{2^{es}}$, где es – максимальное число бит экспоненты. Экспонента является беззнаковым целым числом и предоставляет масштабирование в 2^e раз, где e – значение экспоненты. Данный формат позволяет изменять точность числа и хранить данные об этом в компактном виде. Таким образом, числа, близкие к единице по модулю, будут иметь большую точность, чем числа стремящиеся к бесконечности либо нулю. Стоит отметить, что НС обычно оперируют числами в промежутке $[-1; 1]$. Оставшиеся биты после битов режима и экспоненты представляют дробную часть, она имеет тот же формат, что и в float, но скрытый бит всегда равен 1. Значения по битовому представлению POSIT p вычисляются по следующей формуле:

$$x = sign(p) \times useed^k \times 2^e \times f$$

Однако, при $p = 0$ $x = 0$, а при $p = -2^{n-1}$ $x = \pm\infty$. Например, 5-битное POSIT число с максимальным числом бит экспоненты равным 2 имеет максимальное представимое число 4096, а минимальное по модулю – 1/4096 [1, 2].

II. СРАВНЕНИЕ ЧИСЕЛ POSIT И ЧИСЕЛ С ФИКСИРОВАННОЙ ТОЧКОЙ

Достаточно часто при проектировании НС на FPGA отдают предпочтение формату представления числовых значений с фиксированной точкой, поэтому целесообразным является сравнение данного формата с форматом POSIT.

Для сравнения будут использованы 5-битные представления чисел с фиксированной точкой в прямом коде и числа POSIT (будут представлены в трех модификациях: с максимальным числом бит экспоненты 0, 1 и 2, чтобы увидеть влияние данного параметра).

5-битное представление с фиксированной точкой может представить 31 (из-за положительного и отрицательного нулей, что можно исправить использованием дополнительного кода – 32 значения без двух нулей) значение с постоянной разницей между ними 0.125.

Для быстрого генерирования POSIT последовательностей с указанием полного числа бит и максимального числа бит экспоненты был написан скрипт на языке Python. Сгенерированные три последовательности (для простоты только в положительной области) представлены в таблицах по ссылке [3].

Итак, числа POSIT с нулевой экспонентой от 0 до 1 совпадают с числами с фиксированной точкой в этом же интервале, однако разность между числами > 1 начинает увеличиваться: 0.25, 0.5, 1 и 4 – вместо постоянной 0.125. Таким образом, видно, что точность чисел с приближением к крайним значениям падает (плавная кривая нарастания погрешности). Необходимо сказать, что покрытие числовой прямой обычными числами с фиксированной точкой равномерно и легко предсказуемо, чего нельзя утверждать о числах POSIT.

Представимый интервал также различен, для POSIT с нулевой экспонентой это [-8; 8], для чисел с фиксированной точкой это [-1.875; 1.875]. Принимая во внимание, что выходное значение функций активации НС обычно принадлежит интервалу [-1; 1] (например, сигмоида), можно сделать вывод, что увеличенный интервал представления чисел POSIT в данном случае излишен.

При сравнении чисел с фиксированной точкой и чисел POSIT с экспонентой в 1 и 2 бит видно, что придаваемый масштаб еще больше уменьшает точность чисел, в том числе и на интервале [0; 1].

Число бит экспоненты влияет на диапазон и точность чисел, поэтому для каждой задачи необходимо подбирать данный параметр, что

вносит дополнительную сложность при работе с POSIT. Однако такая особенность дает возможность использовать различное число бит экспоненты в определенных ситуациях, например, для представления результатов функций активации и представления весов сети, однако это приводит к усложнению конвертирования чисел между форматами.

Необходимо отметить, что для реализации арифметико-логического устройства для обработки чисел POSIT требуется в десятки раз больший объем вычислительных ресурсов FPGA [4].

При работе со значениями, для представления которых достаточно чисел с фиксированной точкой и 3–6 битами для дробной части, использование чисел POSIT не дает никаких преимуществ.

ЗАКЛЮЧЕНИЕ

В данной работе были проведены характеристика формата представления чисел POSIT, сравнение с числами с фиксированной точкой для представления чисел в нейронных сетях, реализуемых на FPGA. Преимущества использования формата POSIT не выявлено в случаях, где достаточно чисел с фиксированной точкой и небольшим числом бит дробной части. Также необходимо отметить, что формат POSIT является достаточно новым, представленные в источниках примеры зачастую рассматривают формат со стороны, где он лучше справляется с задачей. Однако существуют примеры, где присутствуют грубые несовпадения результатов вычислений с реальными [5]. Формат имеет достаточно большое число особенностей, которые должны быть учтены перед использованием, и формат не является универсальным, как, например, форматы с плавающей (IEEE 754) или фиксированной точкой.

СПИСОК ЛИТЕРАТУРЫ

1. Posit Arithmetic [Electronic resource] / J. L. Gustafson. – Unum and Posit, 2017. – Mode of access: <https://posithub.org/docs/Posits4.pdf>. – Date of access: 15.09.2019.
2. Beating Floating Point at its Own Game: Posit Arithmetic [Electronic resource] / J. L. Gustafson, I. Yonemoto. – Dr. John L. Gustafson, 2018. – Mode of access: <http://www.johngustafson.net/pdfs/BeatingFloatingPoint.pdf>. – Date of access: 16.09.2019.
3. its-conference-thesis-2019 [Electronic resource] / M. Borovsky. – GitHub, 2019. – Mode of access: <https://github.com/maximka777/its-conference-thesis-2019>. – Date of access: 20.09.2019.
4. Parameterized Posit Arithmetic Hardware Generator [Electronic resource] / R. Chaurasiya, J. L. Gustafson, R. Shrestha. – Unum and Posit, 2018. – Mode of access: https://posithub.org/docs/iccd_submission_v1.pdf. – Date of access: 22.09.2019.
5. Posits: the good, the bad and the ugly [Electronic resource] / F. de Dinechin, L. Forget, J. Muller. – Univ Lyon, 2019. – Mode of access: <https://hal.inria.fr/hal-01959581v3/document>. – Date of access: 22.09.2019.

ЛЕКСИЧЕСКАЯ ОБФУСКАЦИЯ КАК СПОСОБ ВНЕДРЕНИЯ ВОДЯНЫХ ЗНАКОВ В ИСХОДНЫЕ КОДЫ ПРОГРАММ И ПРОЕКТНЫХ ОПИСАНИЙ

Видничук В. Н., Иванюк А. А.

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: vidnichuk@bsuir.by, ivaniuk@bsuir.by

Предложен метод внедрения водяных знаков на основе лексической обфускации исходных кодов программ, его основные положения. Предложены символы использующиеся для обфускации и внедрения водяных знаков.

ВВЕДЕНИЕ

В современном мире, в связи с быстрым ростом производимых компаниями программ и проектных описаний растёт и проблема пиратства. Для решения данной проблемы предлагается рассмотреть метод внедрения водяного знака в метод лексической обфускации программного кода.

Обфускация или запутывание кода — приведение исходного текста или исполняемого кода программы к виду, сохраняющему её функциональность, но затрудняющему анализ, понимание алгоритмов работы и модификацию при декомпиляции.

Основной целью лексической обфускации является переименование названия переменных и идентификаторов на похожие друг на друга. В большинстве языков программирования названия переменных можно представить в виде множества символов:

$$\langle ID \rangle = \{[A..z] \cup [0..9] \cup [_]\},$$

где, название переменной должно начинаться с буквы и может состоять из латинских символов и цифр. Также, некоторые языки программирования, поддерживают кириллические и специальные символы. В связи с этим для выполнения метода переименования переменных в разных языках программирования, используются разные наборы символов.

I. МЕТОД ПЕРЕИМЕНОВАНИЯ ПЕРЕМЕННЫХ

Это один из основных методов лексической обфускации кода. Данный метод заключается в том, что мы переименовываем все переменные и идентификаторы в похожие друг на друга. В начале проверяется количество переменных и выбирается длина для переименованных. Далее с помощью равномерного распределения строится первоначальная переименованная переменная, которая состоит из символов '0' и '1'. В результате получается название переменной вида: '100101001011'. Используя сгенерированное начальное название переменной выполняется генерация остальных названий путём замены '0' и '1'

на символы визуально похожие на них из набора символов замены. При использовании данного метода следует знать, что все идентификаторы должны быть похожи и отличаться друг от друга только на 1 символ, в связи с этим существует возможность внедрения водяных знаков в его, путём выбора мест замены символов в соответствии с водяным знаком.

II. ЛЕКСИЧЕСКАЯ ОБФУСКАЦИЯ И ВНЕДРЕНИЕ ВОДЯНЫХ ЗНАКОВ В VHDL-ОПИСАНИЯ

В языке проектирования VHDL множество символов, доступных для названия переменных представлено следующим множеством:

$$\langle ID \rangle = \{[A..z] \cup [..] \cup [0..9] \cup [\acute{A}..\acute{z}]\},$$

где $\acute{A} .. \acute{z}$ является множеством диакритических и специальных символов, которые на этапе синтеза VHDL-описания приводятся к соответствующему символу без диакритического знака. Например, диакритическая \acute{O} приводится к обычному латинскому символу O , однако, имеют разные коды символов, поэтому при выполнении лексического анализа обфусцированного кода простейшими лексическими деобфускаторами могут возникать ошибки и непонятности. Отсюда следует возможность использования диакритических символов для формирования идентификаторов повышенной сложности. Можно рассмотреть следующий пример обфускации с использованием диакритических символов: идентификатор $O\acute{I}O\acute{O}\acute{I}O\acute{I}O\acute{I}\acute{O}$ при синтезе приведётся к идентификатору $OIOOIIIOIO$, однако, идентификатор $O\acute{I}O\acute{O}\acute{I}O\acute{I}O\acute{I}\acute{O}$ также при синтезе приведётся к идентификатору $OIOOIIIOIO$ и при поиске идентификатора, содержащего диакритический символ найдётся только он сам, отсюда следует, что в объявлении и использовании идентификаторов мы можем применять разные диакритические символы, что затруднит понимание кода, но не повлияет на результат синтеза. Следовательно, с помощью использования данных символов можно запутывать не только человека, но и программы для деобфускации кода. Например, идентификатор

ОЮ’ОШОПОЮ’О, где ’О является кириллическим символом О и идентификатор ОЮ’ОШОПОЮ’О для человека выглядят идентичные, однако, при синтезе являются разными идентификаторами. В связи с этим было выделено следующее множество похожих символов, использующихся для лексической обфускации VHDL-описаний для замены единичных символов:

$$\langle 1mn \rangle = \{1, l, i, j\}.$$

А для замены нулевых символов используется множество:

$$\langle 0mn \rangle = \{0, O, O', Q\}.$$

После формирования первичного обфусцированного идентификатора выполняется внедрение водяного знака. Это осуществляется при формировании остальных идентификаторов. Анализируются количество уникальных идентификаторов в описании, и формируется ключ водяного знака. Данный ключ определяет какие из символов первоначально сгенерированного идентификатора могут изменяться при формировании названий остальных идентификаторов. Далее, для усложнения понимания исходных описаний, случайные символы ’О’, ’Г’, ’Г’ заменяются на диакритические из множеств:

$$\langle DO \rangle = [\acute{O}, \check{O}, \grave{O}, \hat{O}, \text{ , , }, \ddot{O}, \tilde{O}, \emptyset, \text{ , }].$$

$$\langle D1 \rangle = [\acute{I}, \check{I}, \grave{I}, \hat{I}, \text{ , , }, \ddot{I}, \tilde{I}, \text{ , , }, \text{ , }].$$

Что не позволяет легко деобфусцировать и разобрать зашифрованный код.

III. ЛЕКСИЧЕСКАЯ ОБФУСКАЦИЯ И ВНЕДРЕНИЕ ВОДЯНОГО ЗНАКА В ЯЗЫКЕ C/C++

В языке программирования C/C++ множество символов, доступных для названия переменных выглядит следующим образом:

$$\langle ID \rangle = \{[A..z] \cup [0..9] \cup [_]\}.$$

В связи с этим набор символов для замены ограничен и представлен следующим множеством:

$$\langle ID' \rangle = \{O, 0, Q, 1, l, I\}.$$

Для внедрения водяного знака в код предлагается использовать следующий метод: формируется первоначальная переменная, состоящая из ’0’ и ’1’. Далее случайные символы этой переменной меняются на подобные. Получается обфусцированная первоначальная переменная, которая используется для генерации остальных переменных путём замены 1 из символов на подобный по ключу водяного знака.

IV. ЛЕКСИЧЕСКАЯ ОБФУСКАЦИЯ И ВНЕДРЕНИЕ ВОДЯНОГО ЗНАКА В ЯЗЫКЕ JAVA

Множество символов, доступных для использования в названиях переменных на языке программирования Java:

$$\langle ID \rangle = \{[A..z] \cup [.] \cup [0..9] \cup [A..z]\}.$$

В отличие от языка проектирования VHDL, диакритические символы не заменяются на латинские из стандартного алфавита, поэтому усложнить понимание путём внедрения диакритических символов не представляется возможным. Метод лексической обфускации выглядит следующим образом: формируется первоначальный идентификатор, состоящий из ’0’ и ’1’, далее формируются замены из множества замен похожих символов. Для ’0’ множество выглядит следующим образом:

$$\langle ID \rangle = \{[\grave{O}..\grave{o}] \cup [\grave{Q}..\grave{q}]\}.$$

Для ’1’ следующее множество замены:

$$\langle ID \rangle = \{[\grave{I}..\grave{i}] \cup [\grave{J}..\grave{j}] \cup [\grave{L}..\grave{l}]\}.$$

После формирования первичных названий переменных создаются остальные по ключу водяного знака путём замены 1 символа, находящегося на месте возможной замены на один из множеств замены.

V. ВЫВОДЫ

В данной статье был предложен метод лексической обфускации на основе диакритических символов, что усложняет понимание исходных кодов программ и проектных описаний, а также предложен метод внедрения водяных знаков в лексически обфусцированный код.

1. Collberg, C. A Taxonomy of Obfuscating Transformations / C. Collberg, C. Thomborson, D. Low – Auckland: Department of Computer Science, 1997. – 36 p.
2. Сергейчик В. В. Методы лексической обфускации VHDL-описаний / В. В. Сергейчик, А. А. Иванюк // Информационные технологии и системы 2013 (ИТС 2013) : материалы международной научной конференции, БГУИР, Минск, Беларусь, 23 октября 2013 г. = Information Technologies and Systems 2013 (ITS 2013) : Proceeding of The International Conference, BSUIR, Minsk, 24th October 2013 / редкол.: Л. Ю. Шилин [и др.]. – Минск : БГУИР, 2013. – С. 198-199.
3. Видничук, В. Н. Использование лексической обфускации VHDL-описаний для внедрения водяных знаков / В. Н. Видничук // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2019. – С. 191 - 193.
4. Garg S., Gentry C., Halevi S., Raykova M., Sahai A., and Waters B. «Candidate indistinguishability obfuscation and functional encryption for all circuits.» FOCS 2013.

ПСЕВДОИСЧЕРПЫВАЮЩЕЕ ТЕСТИРОВАНИЕ ОЗУ ВСТРАИВАЕМЫХ СИСТЕМ

Леванцевич В. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: lvn@bsuir.by

Анализируются методы тестирования современных запоминающих устройств, в том числе оперативных запоминающих устройств (ОЗУ) встраиваемых систем. Обосновывается использование многократных тестов ОЗУ с изменяемым начальным состоянием для псевдоисчерпывающего тестирования ОЗУ. Приводятся оценки минимальной, средней кратности многократного теста для обеспечения исчерпывающего множества комбинаций для заданного числа ячеек ОЗУ

ВВЕДЕНИЕ

В кодочувствительных неисправностях (Pattern Sensitive Faults – PSF) логическое состояние или изменение логического состояния одного запоминающего элемента ОЗУ, зависит от содержимого (0/1) или логических переходов 0/1 или 1/0 влияющих запоминающих элементов [1]. На практике используют модель граничных кодочувствительных неисправностей (Neighborhood Pattern Sensitive Faults – NPSF), для которой количество элементов k участвующих в неисправности, не превышает 10. Так как логическая топология ОЗУ, как правило, не совпадает с физической, то влияющими ячейками, участвующими в кодочувствительной неисправности, могут быть любые ячейки ОЗУ. Для обнаружения таких неисправностей процедура тестирования должна сформировать в любых k из N ячеек ОЗУ все возможные 2^k двоичные комбинации.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Исчерпывающее тестирование характеризуется максимальной эффективностью обнаружения неисправностей и позволяет сформировать в N ячейках ОЗУ все 2^N возможные комбинации. Однако, исчерпывающие тесты имеют слишком большую временную сложность, что ограничивает их применение. Поэтому на практике используют аппроксимацию исчерпывающего тестирования в виде псевдоисчерпывающих тестов [2]. Псевдоисчерпывающий тест формирует в любых k из N ($k < N$) ячейках все возможные 2^k комбинации, т. е. является «локально исчерпывающим» тестом. Управляемые вероятностные тесты, так же являются аппроксимацией исчерпывающих тестов, однако, такие тесты имеют значительную вычислительную сложность, обусловленную необходимостью перебора потенциальных кандидатов в тестовые наборы, и принятия решения о включении или не включении их в вероятностный тест [5]. Для уменьшения сложности формирования управляемых вероятностных тестов и повышения их эффективности приме-

няют многократные управляемые вероятностные тесты [3]. При этом для повышения эффективности многократных тестов используют два подхода: на каждой итерации многократного теста изменяют либо порядок следования адресов, либо начальные состояния ячеек ОЗУ.

II. РЕАЛИЗАЦИЯ

Проанализируем возможность использования многократных тестов с изменяемыми начальными состояниями и фиксированным порядком следования адресов, для псевдоисчерпывающего тестирования ОЗУ. Введем понятие орбиты, которая представляет собой количество двоичных векторов, формируемых одной фазой маршевого теста. Конкретный набор векторов, составляющих орбиту, будет зависеть от двух основных факторов [3]. Прежде всего, используемый маршевый тест однозначно определяет структуру орбиты, так как он формулирует правила, по которым генерируются двоичные векторы в ячейках запоминающего устройства. Начальное состояние запоминающего устройства является вторым аргументом, который принимается во внимание при формировании конкретной орбиты. Для примера рассмотрим биториентированное ОЗУ, состоящее из $k = 3$ ячеек. На рисунке 1 показан результат многократного выполнения простейшего маршевого теста $\{\uparrow(r0, w0)\}$, выполняющего инвертирование содержимого ячеек памяти, при различных начальных состояниях ОЗУ и фиксированном порядке следования адресов.

O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7
0 0 0 0 0 1 0 1 0 0 1 1 1 0 0 1 0 1 1 1 0 1 1 1	0 0 1 0 0 0 0 1 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 0	0 1 1 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 1 0 1 1 0 0	1 1 1 1 1 1 0 1 0 1 1 0 0 0 1 1 0 1 0 0 0 1 0 0 0				

Рис. 1 – Орбиты для разных начальных состояний ОЗУ

Анализ рисунка позволяет сделать следующие выводы. Во первых: для k ячеек памяти существует всего 2^k возможных орбит. Во вторых:

в каждой орбите существует $k + 1$ двоичный вектор.

Рассмотрим два классических маршевых теста MATS++ $\{\uparrow(w0); \uparrow(r0, w1); \downarrow(r1, w0, r0)\}$ и MarchC- $\{\uparrow(w0); \uparrow(r0, w1); \uparrow(r1, w0); \downarrow(r0, w1); \downarrow(r1, w0); \uparrow(r0)\}$. Можно отметить, что одна итерация теста MATS++ формирует две орбиты, содержащие $k + 1$ различных двоичных вектора, а одна итерация теста March C- формирует четыре орбиты, содержащие $2k$ различных векторов.

Если в идеальном случае, все последующие орбиты теста MATS++ будут включать по $k + 1$ новому двоичному вектору, ранее не использованному в предыдущих орбитах, а все последующие орбиты теста MarchC- будут включать $2k$ новых двоичных векторов, то минимальное количество орбит, необходимое для формирования всех возможных комбинаций в любых k из N двоичных комбинаций для теста MATS++ должно удовлетворять неравенству [3]:

$$Q_{max} \geq \lceil \frac{2^k}{k+1} \rceil$$

а для теста March- неравенству

$$Q_{max} \geq \lceil \frac{2^{k-1}}{k} \rceil$$

где $k = 1, 2, 3, \dots$

Так как начальные состояния ячеек памяти, в случае вероятностных тестов, представляют собой случайные равномерно распределенные значения, то для определения среднего количества орбит, необходимых для формирования в любых k из N ячейках ОЗУ, всех 2^k возможных комбинаций, можно применить известную в комбинаторике задачу собирателя купонов [3]. В этом случае среднее количество орбит можно определить по выражению:

$$Q_{ave} = 1 + \frac{2^k}{2^k - 1} + \frac{2^k}{2^k - 2} + \dots + \frac{2^k}{2} + 2^k = 2^k \times \sum_{n=1}^{2^k} \frac{1}{n}$$

Учитывая количество различных двоичных векторов, формируемых одной итерацией тестов MATS++ и MarchC-, среднее количество орбит $Q_{ave}(\text{MATS}++)$ для теста MATS++ можно определить по формуле:

$$Q_{ave}(\text{MATS}++) = \frac{Q_{ave}}{k+1}$$

а среднее количество орбит для теста $Q_{ave}(\text{MarchC-})$ определяется выражением:

$$Q_{ave}(\text{MarchC-}) = \frac{Q_{ave}}{2k}$$

На рис. 2 и 3 представлены теоретические и экспериментальные оценки кратности для двух множеств многократных маршевых тестов с изменяемыми начальными состояниями MATS++

и MarchC- для произвольных k ячеек памяти. Экспериментальные значения величины Q_{ave} представляет собой среднее значение количества итераций тестов в процентах, полученное на основании 100 000 экспериментальных значений.

k	2	3	4	5	6	7	8	9	10
2^k	4	8	16	32	64	128	256	512	1024
Теор. Q_{ave}	2,77	5,44	10,82	21,64	43,37	86,93	174,20	349,00	699,03
Эксп. Q_{ave}	-	4,44	9,42	19,54	40,90	83,57	170,36	341,10	680,86
Эксп. Q_{min}	-	2	4	7	17	41	96	201	413
Эксп. Q_{max}	-	16	31	60	110	201	366	706	1405

Рис. 2 – Теоретические и экспериментальные значения $Q_{ave}(\text{MATS}++)$

k	2	3	4	5	6	7	8	9	10
2^k	4	8	16	32	64	128	256	512	1024
Теор. Q_{ave}	2,77	3,62	6,76	12,98	25,30	49,67	97,98	193,89	384,47
Эксп. Q_{ave}	-	2,33	4,42	9,34	19,55	40,55	82,62	169,32	339,48
Эксп. Q_{min}	-	2	2	4	7	18	37	86	213
Эксп. Q_{max}	-	7	17	28	65	113	207	391	769

Рис. 3 – Теоретические и экспериментальные значения $Q_{ave}(\text{MarchC-})$

III. ВЫВОДЫ

Анализ полученных данных показывает, что средние значения количества орбит, полученные по предложенным аналитическим выражениям, приближены к значениям, полученным экспериментально. Из рисунков видно, что среднее количество орбит для получения всех исчерпывающих комбинаций в любых k из N ячейках ОЗУ для теста MATS++ почти в два раза больше, чем для теста March C-. Это связано в первую очередь со структурой входящих в тест маршевых элементов. Учитывая, что одна итерация теста MATS++ генерирует две орбиты, а одна итерация теста March C- четыре орбиты, несложно подсчитать количество итераций многократных тестов. Таким образом, предложенные подходы могут быть использованы для построения многократных тестов, обнаруживающих сложные кодочувствительные неисправности ОЗУ, в том числе и ОЗУ встроенных систем.

СПИСОК ЛИТЕРАТУРЫ

1. Barzilai, Z. Exhaustive Generation of Bit Pattern with Application to VLSI Self-Testing / Z. Barzilai, D. Coppersmith, A. Rozenberg // IEEE Transactions on Computers. - 1983. - Vol. C.31, no. 2. - P. 190-194.
2. Das, D. Exhaustive and Near-Exhaustive Memory Testing Techniques and their BIST Implementations / D. Das, M.G. Karpovsky // Journal of Electronic Testing. - 1997. - Vol. 10. - P. 215-229.
3. Ярмолик, В.Н. Псевдоисчерпывающее тестирование ОЗУ / В.Н. Ярмолик, И. Мрозек, В.А. Леванцевич // Информатика. - 2017. - №2(54). - С.58-69.
4. Yarmolik, S.V. The Syntheses of Probability Tests with a Small Number of Kits / S. V. Yarmolik, V.N. Yarmolik // Automatic Control and Computer Science. - 2011. - Vol. 45, no.3. - P. 133-141.
5. Ярмолик, С.В. Многократные неразрушающие маршевые тесты с изменяемыми адресными последовательностями / С.В. Ярмолик, В.Н. Ярмолик // Автоматика и телемеханика. - 2007. - №4. - С. 126-137.

АЛГОРИТМ ДИЗЪЮНКТИВНОГО РАЗЛОЖЕНИЯ СИСТЕМЫ ПОЛНОСТЬЮ ОПРЕДЕЛЕННЫХ БУЛЕВЫХ ФУНКЦИЙ

Кардаш С. Н.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: gold@newman.bas-net.by

Приводится алгоритм дизъюнктивного разложения систем полностью определенных булевых функций, ориентированный на минимизацию сложности получаемых в результате синтеза схем из библиотечных КМОП-элементов. Показано, что сокращение площади схем в результате применения предлагаемых программ может достигать 5,5 процентов

ВВЕДЕНИЕ

В работе [1] предложен алгоритм построения дизъюнктивно-конъюнктивных разложений систем полностью определенных булевых функций. В работе [2] показано, что применение дизъюнктивно-конъюнктивных разложений при синтезе нерегулярных логических схем из библиотечных КМОП-элементов на этапе многоуровневой оптимизации в большинстве рассматривавшихся случаев привело к получению более эффективных схемных решений.

Основное отличие предлагаемого алгоритма от рассмотренного в [1] состоит в том, что компоненты связности отыскиваются не для пар функций, а для их подмножеств возможно большей размерности. Идея алгоритма состоит в выделении из системы из m функций, зависящих от n переменных, некоторой подсистемы функций (компоненты связности) и построении соответствующего дизъюнктивного разложения. Функции, не вошедшие в компоненту связности, образуют подсистему «остаток».

Пусть задан параметр связности ρ^1 (связность по областям единичных значений функций), ρ^0 (связность по областям нулевых значений функций) в процентах к общему числу наборов булева пространства.

Функции системы, конъюнкция всех столбцов значений которых имеет вес не ниже $w = \lfloor \rho^1 \times 2^n \rfloor$, ($\lfloor a \rfloor$ – ближайшее сверху целое к a), будем называть w -совместимыми.

Функцию будем называть w -совместимой с некоторой подсистемой, если конъюнкция ее столбца значений с конъюнкцией всех столбцов значений подсистемы имеет вес не ниже $w = \lfloor \rho^1 \times 2^n \rfloor$.

Шаг 1. Оценивается связность «по единицам» и «по нулям» столбцов значений функций и инверсий функций системы. Каждая из функций f^i рассматривается как в прямой f^i , так и инверсной \bar{f}^i форме.

Перебираются $C_{2m}^2 - m$, $m > 1$ всевозможных неупорядоченных пар функций текущей си-

стемы (пары $\{f^i, f^j\}$, $\{\bar{f}^i, \bar{f}^j\}$, $\{f^i, \bar{f}^j\}$) не рассматриваются).

Шаг 2. Формирование первой подсистемы. Первую пару образуют функции с максимальным значением ρ^1 , если таких пар несколько – выбирается первая из них. Выбранные функции помечаются. Затем происходит процесс наращивания множества функций подсистемы. На каждом шаге последовательно перебираются непомеченные функции и в формируемую подсистему добавляется первая функция (либо ее инверсия), не нарушающая условия w -совместимости.

Процесс добавления прекращается, если среди непомеченных функций не осталось w -совместимых с формируемой подсистемой функций.

Полученная подсистема имеет не менее ρ^1 наборов, на которых все функции подсистемы имеют единичное значение. Для полученной подсистемы проводится дизъюнктивное разложение. Частичные функции (остатки, входящие в дизъюнктивное разложение подсистемы) раздельно минимизируются в классе ДНФ по критерию «число литералов».

Шаг 3. Формирование второй подсистемы. Из исходной системы удаляются помеченные функции. Для оставшихся функций находится подсистема для конъюнктивного разложения. Процесс аналогичен шагу 2 с заменой критерия $\rho^1 \times 2^n$ на $\rho^0 \times 2^n$.

Шаг 4. Если среди оставшихся пар максимальную связность имеет пара по «нулям», то в данную пару добавляются функции для конъюнктивного разложения. Процесс добавления прекращается, если нельзя добавить функцию без нарушения ограничения ρ^0 . Для данной подсистемы проводится конъюнктивное разложение. Частичные функции минимизируются в классе ДНФ раздельно.

Шаг 5 (итеративный). Такой процесс формирования подсистем продолжается до тех пор, пока все функции (либо их инверсии) не будут размещены по подсистемам, либо когда останутся пары, не удовлетворяющие связности ни по ρ^1 ,

ни по ρ^0 . Оставшиеся функции образуют подсистему «остаток».

I. ОРГАНИЗАЦИЯ ЭКСПЕРИМЕНТА

Исследование проводилось в системе FLC [3]. В качестве объектов исследования использовались функциональные описания (системы полностью определенных булевых функций) проектируемых схем, представленные на языке SF [4], который является внутренним языком системы FLC. Исходные функциональные описания схем взяты из библиотеки примеров Berkeley PLA Test Set [5]. Всего для каждого примера рассматривалось два варианта построения логической схемы. В первом случае программы построения дизъюнктивного разложения не использовались. Этому варианту в таблице 1 соответствует столбец с обозначением A1. При втором варианте (столбец таблицы 1 с обозначением A2) применялась многошаговая процедура. На первом шаге проводилось построение дизъюнктивного разложения. В результате получалось многоуровневое SF-описание, каждый уровень которого задавался алгебраическим выражением. После этого осуществлялась конвертация многоуровневых SF-описаний в синтезируемые алгоритмические VHDL-описания. После этапа конвертации выполнялся второй этап – технологическое отображение оптимизированного многоуровневого в заданный базис библиотечных логических элементов, выполненных по КМОП-технологии. Для этого использовался программный пакет LeonardoSpectrum. В результате получалась структурное VHDL-описание (нетлист) комбинационной схемы, реализующей исходную систему функций. Основным критериям при оптимизации схем являлось число ячеек (cell), занимаемых полученным блоком.

II. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

Результаты экспериментального исследования представлены в таблице 1. В приведенной

таблице через k обозначено число элементарных конъюнкций в исходной системе;

через P - параметр, задающий процент числа функций, вошедших в компоненту связности, к общему числу функций исходной системы;

через fun - число элементарных конъюнкций в общей части разложения; через os - число элементарных конъюнкций в остатке;

через cells - число макроячеек в полученной схеме. Как следует из приведенной таблицы, использование программы дизъюнктивного разложения (для примера Max1024 при $P=4$) привело к сокращению числа ячеек в схеме на 5,5 процентов.

ЗАКЛЮЧЕНИЕ

Уменьшить площадь получаемых в процессе синтеза комбинационных нерегулярных логических схем можно с помощью логической оптимизации исходных описаний. Хорошим средством для улучшения конечных схемных решений может служить предварительная обработка исходных систем с помощью программ построения дизъюнктивных разложений. Проведенное исследование показало, что использование разработанных программных средств может сократить площадь получаемых схем до 5,5 процентов.

СПИСОК ЛИТЕРАТУРЫ

1. Бибило, П. Н. Дизъюнктивно-конъюнктивные разложения систем полностью определенных булевых функций / П. Н. Бибило, С. Н. Кардаш / Докл. Восьмой Междунар. науч. конф. «Танаевские чтения», 27–30 марта 2018 г. – Минск: ОИПИ НАН Беларуси, 2018 г. – С. 28–32.
2. Кардаш, С. Н. Экспериментальное исследование алгоритмов дизъюнктивно-конъюнктивного разложения систем полностью определенных булевых функций / С. Н. Кардаш / Тр. междунар. науч. конф. «Информационные технологии и системы 2018», 25 октября 2018 г., Минск. – Минск: БГУИР, 2018. – С. 160–161.
3. Бибило П.Н., Романов В.И. Логическое проектирование дискретных устройств с использованием продукционно-фреймовой модели представления знаний. Изд. 2-е, испр. – М.: ЛЕНАНД, 2014, 256 с.
4. Бибило, П.Н. Кремниевая компиляция заказных СБИС. - Минск: Институт технической кибернетики АН Беларуси, 1996. - 268 с.
5. Berkeley PLA test set [Electronic resource]. – Mode of access: <http://www1.cs.columbia.edu/cs4861/sis/espresso-examples/ex/>. – Date of access: 10.10.2011.

Таблица 1 – Результаты эксперимента

Имя	n	k	m	P	fun	os	A1	A2
							cells	cells
Max512	9	512	6	100	29	483	581	552
Max512	9	512	6	50	113	504	581	544
Max512	9	512	6	25	159	506	581	529
Max512	9	512	6	17	270	503	581	525
Max1024	10	1024	6	100	58	966	1411	1372
Max1024	10	1024	6	50	226	1008	1411	A1
Max1024	10	1024	6	25	318	1012	1411	1338
Max1024	10	1024	6	17	540	1006	1411	1351

УСТРАНЕНИЕ ИЕРАРХИИ ТРАНЗИСТОРНЫХ ОПИСАНИЙ ЛОГИЧЕСКИХ СХЕМ

Кириенко Н. А.

Объединённый институт проблем информатики Национальной академии наук Беларуси
Минск, Республика Беларусь
E-mail: kir@newman.bas-net.by

Рассматривается задача преобразования иерархических структурных описаний транзисторных схем в одноуровневые описания. Представлены форматы описаний транзисторных схем, алгоритм преобразования форматов. Алгоритм положен в основу процедуры, функционирующей в экспериментальной системе логического проектирования.

ВВЕДЕНИЕ

Объекты проектирования, поступающие на вход САПР СВИС, имеют, как правило, иерархическую структуру, т.е. проектировщик представляет задание на проектирование в виде схемы взаимосвязей некоторых логических блоков, каждый из которых также может быть представлен совокупностью взаимосвязанных логических блоков. Листьевыми блоками могут быть некоторые стандартные библиотечные элементы, логика их функционирования может описываться либо логическими уравнениями, либо системой ДНФ, либо структурной схемой из транзисторов. В последнем случае мы имеем дело с транзисторными описаниями объектов проектирования.

В настоящей работе приводятся постановка задачи и алгоритм устранения иерархии транзисторных описаний – преобразования многоуровневых описаний логических схем в одноуровневые транзисторные описания.

I. ФОРМАТ ОПИСАНИЯ ТРАНЗИСТОРНЫХ СХЕМ В СИСТЕМАХ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

Многоуровневое транзисторное описание логической схемы может быть получено в результате преобразования структурного описания устройства, реализованного в базе заданной библиотеки проектирования, в SPICE-формат. Структурное описание может быть результатом синтеза устройства по заданному исходному описанию поведения, представленного в листинге 1, в одной из промышленных систем проектирования, например, Leonardo Spectrum фирмы Mentor Graphics [1].

Листинг 1. Исходное описание поведения логической схемы на языке VHDL.

```
entity BDD_CONN is
port(x1, x2, x3 : in std_logic;
f1:out std_logic);
end BDD_CONN;
architecture BDD_CONN_arch of BDD_CONN is
begin
f1<=((NOT x3 AND NOT x1) OR x2 or x3);
end BDD_CONN_arch;
```

Описание синтезированной схемы представлено в листинге 2. В описании представлены

определения входных (x1, x2, x3), выходных (y) переменных, основных элементов схемы (ix5, ix66, ix68). Для каждого элемента определен его тип, входные и выходные сигналы.

Листинг 2. Синтезированная схема в базе КМОП элементов.

```
entity BDD_CONN is
port (
x1 : IN std_logic ;
x2 : IN std_logic ;
x3 : IN std_logic ;
f1 : OUT std_logic) ;
end BDD_CONN ;
architecture BDD_CONN_arch of BDD_CONN is
signal nx65, nx67: std_logic ;
begin
ix5 : NA3 port map ( Y=>f1, A=>x1, B=>nx65,
C=>nx67);
ix66 : N port map ( Y=>nx65, A=>x2);
ix68 : N port map ( Y=>nx67, A=>x3);
end BDD_CONN_arch ;
```

По описанию синтезированной схемы может быть построен файл в формате SPICE для моделирования в системе схемотехнического моделирования AccuSimII (фирмы Mentor Graphics) [2], который представлен в листинге 3. Описание каждого элемента схемы преобразовано в соответствии с требованиями AccuSimII, но содержит практически ту же информацию, что и синтезированная схема.

Листинг 3. Описание схемы в формате SPICE.

```
.subckt BDD_CONN f1 x1 x2 x3
X_ix5 f1 x1 nx65 nx67 NA3
X_ix66 nx65 x2 N
X_ix68 nx67 x3 N
.ends BDD_CONN
X_BDD_CONN f1 x1 x2 x3 BDD_CONN
```

II. АЛГОРИТМ ПРЕОБРАЗОВАНИЯ МНОГОУРОВНЕВОГО ТРАНЗИСТОРНОГО ОПИСАНИЯ В ОДНОУРОВНЕВОЕ

Описание схемы в формате SPICE содержит три элемента библиотеки проектирования двух типов: NA3, N. Задача устранения иерархии (или компиляции) транзисторных описаний состоит в замене каждой строки описания элемента строками определения этого элемента на транзисторном уровне. Для выполнения этой задачи необходимо иметь описание библиотеки эле-

ментов на транзисторном уровне. Пример описания элемента типа N на транзисторном уровне представлен в листинге 4.

Листинг 4. Описание элемента библиотеки КМОП типа N на транзисторном уровне

```
* Component: N
subckt N Y A
M2 Y A VCC VCC MODP L=3.5e-07 W=4.5e-06
as=1.04e-11 ad=1.04e-11 ps=1.36e-05 pd=1.36e-05
nrs=0.511111 nrd=0.511111
M1 Y A 0 0 MODN L=3.5e-07 W=2.5e-06 as=5.75e-12
ad=5.75e-12 ps=9.6e-06 pd=9.6e-06 nrs=0.92
nrd=0.92
.ends N
```

Описание каждой подсхемы состоит из раздела внешних имен, тела описания и раздела конца описания.

Раздел внешних имен имеет следующий вид:

```
.SUBCKT <имя подсхемы> <имена внешних полюсов>.
```

Имена внешних полюсов схемы – это перечень формальных параметров, которые при компиляции должны быть заменены фактическими из строки обращения к подсхеме. Тело описания может содержать описания как подсхем (вложенные подсхемы), так и транзисторов. Строка описания вложенной подсхемы имеет следующий вид:

```
<имя подсхемы> <имена полюсов> <имя типа
подсхемы>.
```

Имя подсхемы состоит из схемного имени подсхемы, перед которым записывается префикс «X_». В разделе <имена полюсов> должны быть перечислены имена реальных (фактических) переменных, соответствующих формальным переменным подсхемы. Фактические переменные должны быть указаны в том порядке, который был использован в списке имен внешних полюсов (формальных переменных) в заголовке описания соответствующего типа подсхемы (разделе .SUBCKT).

Строка описания транзистора имеет следующий вид:

```
<имя транзистора><имена узлов><физические
характеристики>.
```

Имя транзистора состоит из заглавной латинской буквы M и номера. Далее следуют четыре идентификатора, означающих имена узлов стока, затвора, истока, подложки транзистора, соответственно. Далее следуют тип модели транзистора (MODP или MODN) и его физические характеристики. Раздел конца описания:

```
.ENDS <имя типа подсхемы>.
```

Задача компиляции решается путем поэтапного раскрытия подсхем в исходном описании и замене листовых подсхем их транзисторными описаниями. В описании более высокого уровня подсхема описывается обращением к ней - заголовком, в котором выписаны фактические параметры (имена переменных, подаваемых на входы и снимаемых с выходов схемы). В заголовке описания подсхемы перечислены формальные параметры. Соответствие имен формальных и фак-

тических параметров задается порядком их перечисления. В процессе компиляции схемы обращение к схеме должно быть заменено описанием ее внутренней структуры, причем формальные параметры должны быть заменены на фактические во всем описании. Имена транзисторов и внутренние имена переменных подсхемы должны быть подвергнуты переименованию, так как в противном случае при раскрытии одинаковых типов схем получится некорректное использование одинаковых имен. В качестве новых имен транзисторов используется тот же символ “M” со сквозной последовательной нумерацией. Внутренние имена в транзисторных описаниях элементов заменяются на числовой порядковый номер. Имена 0 и VCC резервируются за узлами земли и питания.

Решение задачи компиляции заключается в поуровневом просмотре многоуровневого описания. По мере раскрытия очередной подсхемы описания ее транзисторов с замененными именами узлов переписываются в результирующее описание, а о каждой подсхеме заполняются таблицы нераскрытых на данном уровне подсхем. Тип подсхемы запоминается в таблице имен типов, а список фактических параметров - таблице фактических параметров. При выполнении компиляции на i-м уровне, составляются таблицы нераскрытых подсхем для (i+1)-го уровня. Если таблицы для (i+1)-го уровня пусты, то компиляция схемы закончена. Если таблицы для (i+1) уровня не пусты, то происходит обработка подсхем из этих таблиц.

Пример скомпилированной подсхемы BDD_CON представлен в листинге 5.

Листинг 5.

```
.subckt BDD_CONN f1 x1 x2 x3
M12 N4 x1 0 0 MODN
M11 N3 nx65 N4 0 MODN
M8 f1 nx67 N3 0 MODN
M9 f1 nx67 VCC VCC MODP
M2 f1 nx65 VCC VCC MODP
M1 f1 x1 VCC VCC MODP
M13 nx65 x2 VCC VCC MODP
M14 nx65 x2 0 0 MODN
M15 nx67 x3 VCC VCC MODP
M16 nx67 x3 0 0 MODN
.ends BDD_CONN
```

Рассмотренный алгоритм положен в основу процедуры, функционирующей в системе логического проектирования CMOSLD [3], разработанной в Объединенном институте проблем информатики НАН Беларуси.

1. Бибило, П. Н. Системы проектирования интегральных схем на основе языка VHDL. StateCAD, ModelSim, LeonardoSpectrum / П. Н. Бибило. – М. : СОЛОН-Пресс. – 2005. – 384 с.
2. Eldo User's Manual / v. 6.5_1. – 2005.
3. Бибило, П. Н. Система логического проектирования функциональных блоков заказных КМОП СБИС с пониженным энергопотреблением / П. Н. Бибило, Н. А. Авдеев, С. Н. Кардаш, Н. А. Кириенко, Ю. Ю. Ланкевич, И. П. Логинова, Д. И. Черемисинов, Л. Д. Черемисинова // Микроэлектроника. – 2018. – № 1. – С. 72 – 88.

ON BI-DECOMPOSITION OF INCOMPLETELY SPECIFIED BOOLEAN FUNCTIONS

Pottosin Yu. V.

United Institute of Informatics Problems of the National Academy of Sciences of Belarus

Minsk, Republic of Belarus

E-mail: pott@newman.bas-net.by

A method for bi-decomposition of incompletely specified (partial) Boolean functions is suggested. The problem of bi-decomposition is reduced to the problem of a weighted two-block covering the edge set of the orthogonality graph of the truth table rows by complete bipartite subgraphs (bicliques). Every biclique is assigned with a set of variables in a definite way. The weight of a biclique is the size of that set. According to each biclique of the obtained cover, a Boolean function is constructed whose arguments are the variables from the assigned set

INTRODUCTION

The bi-decomposition problem is considered here in the following statement. Given a partial (incompletely specified) Boolean function $f(\mathbf{x})$ where the components of the vector $\mathbf{x} = (x_1, x_2, \dots, x_n)$ are Boolean variables forming a set X . A superposition $\phi(g_1(\mathbf{z}_1), g_2(\mathbf{z}_2)) \geq f(\mathbf{x})$ must be found where the components of the vectors \mathbf{z}_1 and \mathbf{z}_2 are the variables in the sets $Z_1 \subset X$ and $Z_2 \subset X$, respectively, and \geq denotes the realization relation, i.e. the values of the function ϕ coincide with the values of the values of f everywhere they are specified. The type of the function ϕ of two variables is given, as well. It can be any of ten Boolean functions depending essentially on two variables. Usually, the sets Z_1 and Z_2 are given and $Z_1 \cap Z_2 = \emptyset$ and such bi-decomposition is called disjoint in contrast to non-disjoint bi-decomposition when $Z_1 \cap Z_2 = \emptyset$ condition is not obligatory. At that, the cardinalities of Z_1 and Z_2 must be less than n .

There are examples of application of bi-decomposition for increasing the performance of a logic circuit [1, 2] and for synthesis of circuits on the base of field programmable gate array (FPGA) [3]. The problem of bi-decomposition with the output function ϕ expressed by XOR operation is considered in [4] that suggests to use logical equations to solve the problem. In [5], the disjoint bi-decomposition with a given partition (Z_1, Z_2) is investigated in detail. An approach to solving this problem using Boolean differential calculus is suggested in [6]. Here, the approach to solving the general problem of parallel decomposition for partial Boolean function described in [7] is suggested to solve the bi-decomposition problem.

I. THE APPROACH TO THE PROBLEM

The approach uses the specification of a partial Boolean function in truth table that is considered as a pair of binary matrices: matrix \mathbf{X} of dimension $l \times n$ and one-column matrix \mathbf{F} with l elements. The rows of the matrix \mathbf{X} give the values of the arguments x_1, x_2, \dots, x_n and the elements of the matrix \mathbf{F} the corresponding values of the

function. The value of the function is not specified at the set of argument values if this set does not exist among the rows of \mathbf{X} . The rows of \mathbf{X} and \mathbf{F} have common numeration.

The graphs $G_X = (V, E_X)$ and $G_F = (V, E_F)$ are considered where V is the set of common numbers of rows of the matrices \mathbf{X} and \mathbf{F} , and E_X and E_F are the sets of pairs of orthogonal rows of the matrices \mathbf{X} and \mathbf{F} , respectively. Two row-vectors are orthogonal if there is a component being 0 in one vector and 1 in the other. The approach from [7] is intended for a system of partial Boolean functions given by a pair of ternary matrices, and the graphs G_X and G_F can have an arbitrary type. In the case of binary matrices, G_X is complete graph and G_F a complete bipartite graph. A system of Boolean functions is given with matrices \mathbf{X} and \mathbf{F} correctly if $E_F \subseteq E_X$, i.e. G_F is a spanned subgraph of G_X . Every edge in E_X is assigned with the variables from the set $X = \{x_1, x_2, \dots, x_n\}$, according to which the corresponding rows of \mathbf{X} are orthogonal. A complete bipartite subgraph (biclique) of G_X is assigned with the set of variables in X taken one by one from each edge of the biclique. A biclique is called *admissible* if the number of variables assigned to it is less than n , and it contains at least one edge in E_F .

S t a t e m e n t. There exists a superposition $\phi(g_1(\mathbf{z}_1), g_2(\mathbf{z}_2))$ realizing the given partial Boolean function $f(\mathbf{x})$ if a cover of the set E_F by two admissible bicliques of the graph G_X exists.

Let B_1, B_2 be bicliques covering the set E_F . Any biclique B_i can be given by a pair of vertex sets $\langle V_i', V_i'' \rangle$. Every function $g_i(z_i)$ of the required superposition is specified by matrices \mathbf{X}_i and \mathbf{F}_i . The matrix \mathbf{X}_i is the minor of \mathbf{X} formed by the columns corresponding to the variables assigned to the biclique B_i . The matrix \mathbf{F}_i consists of one column where the element with number corresponding to the vertex in V_i' is 0, and the element with number corresponding to the vertex in V_i'' is 1 (or vice versa). The function ϕ is given by matrices \mathbf{U} and $\mathbf{\Phi}$. The matrix \mathbf{U} consists of the columns that are the one-column matrices $\mathbf{F}_1, \mathbf{F}_2$, and the matrix $\mathbf{\Phi}$ coincide with \mathbf{F} .

II. A METHOD FOR BI-DECOMPOSITION

To give the type of the function ϕ one must assign to each pair of rows of matrices \mathbf{X} and \mathbf{F} the set of the pairs of the possible values of g_1 and g_2 required according to the value of the given f . For example, when $\phi = g_1 \wedge g_2$ (ϕ is a conjunction), this set is $\{(1, 1)\}$ if $f = 1$, and $\{(0, 0), (0, 1), (1, 0)\}$, if $f = 0$.

Let bicliques B_1 and B_2 constitute a two-block cover of the set E_F , according to which functions g_1 , g_2 and ϕ can be constructed. Such covers have the following two properties.

Property A. A set of vertices of graph G_X (denote it W), each of which corresponds to equal values of functions g_1 and g_2 , is a subset of partite in the bipartite subgraphs B_1 and B_2 .

Property B. If vertices v_i and v_j from the set $V \setminus W$ are in different partite of one of the bicliques, then they are in different partite of the other one. If they are in the same partite of one of the bicliques, then they are in the same partite of the other one.

For linear functions that are exclusive disjunction and equivalence, both the properties are present. For other functions, Property B is not obligatory. We call two-block cover of E_V by admissible bicliques of graph G_X admissible for non-linear function ϕ if it has Property A, and admissible for linear function ϕ if it has properties A and B.

The process of solution of the considered problem with minimizing the sum of the numbers of arguments of the functions g_1 and g_2 consists of the following stages.

1. Finding all maximal admissible bicliques with Property A in graph G_X . The method suggested in [8] and described later in [9] can be used for that. Note that graph G_F in the problem is a biclique of G_X that is not admissible because it is an one-block cover resulting in trivial solution where one of the functions g_1 or g_2 is a constant. The obtained bicliques are assigned with weights equal to the numbers of variables assigned to the bicliques.

2. Obtaining a two-block cover of the set E_F by the found bicliques. The cover must have minimal weight and, for a linear function ϕ , have Property B. The weight of a cover is the sum of the weights of bicliques that constitute the cover. At this stage, the requirement for non-crossing sets Z_1 and Z_2 can be satisfied when a disjoint decomposition is demanded.

3. Constructing Boolean functions $g_1(z_1), g_2(z_2)$ and ϕ . The functions $g_1(z_1)$ and $g_2(z_2)$ are specified by the pairs of matrices \mathbf{X}_1 ,

\mathbf{F}_1 and \mathbf{X}_2 , \mathbf{F}_2 , and the function ϕ by a formula with the given operation of logic algebra.

The "bottle-neck" of the described approach is the search for all the maximum bicliques in a graph. The number of them in a complete graph with n vertices is $2^{n-1} - 1$. Evidently, graph G_X is complete if all the rows of matrix \mathbf{X} are mutually orthogonal. This takes place always when \mathbf{X} is a Boolean matrix. Property A gives a possibility to decrease the number of the considered bicliques significantly with the help of replacing all the vertices of the set W by one vertex by contraction. The number of maximum bicliques in such a graph is $2^{n-|W|} - 1$ where $|W|$ is the size of W . It should be noted although that the complexity of the problem of the search for all the two-block covers is expressed by a polynomial of the second power. So, enumeration of all the two-block covers is not considered as laborious problem.

СПИСОК ЛИТЕРАТУРЫ

1. Cortadella, J. Timing-driven logic bi-decomposition / J. Cortadella // IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems. – 2003. – Vol. 22 – No. 6. – P. 675 –685.
2. Mishchenko, A. An algorithm for bi-decomposition of logic functions / A. Mishchenko, B. Steinbach, M. Perkowski // Proceedings of the 38th Annual Design Automation Conference. DAC. Las Vegas, NV, USA: ACM, June 2001. – P. 103 –108.
3. Chang, S.-C. Technology mapping for TLU FPGA's based on decomposition of binary decision diagrams / S.-C. Chang, M. Marek-Sadowska, T. Hwang // IEEE Trans. Computer-Aided Design. – 1996. – Vol. 15. – No. 10. – P. 1226 –1235.
4. Bibilo, P. N. Decomposition of Boolean Functions Based on Solving Logical Equations / P. N. Bibilo. – Minsk : Belarus. Navuka, 2009. – 211 p. (in Russian).
5. Zakrevskij, A. D. On a special kind decomposition of weakly specified Boolean functions / A. D. Zakrevskii // Second International Conference on Computer-Aided Design of Discrete Devices (CAD DD'97), Minsk, Republic of Belarus, November 12-14, 1997. – Minsk: National Academy of Sciences of Belarus, Institute of Engineering Cybernetics. – 1997. –Vol. 1. – P. 36 –41.
6. Steinbach, B. Vectorial bi-decomposition for lattices of Boolean functions / B. Steinbach, C. Posthoff // Further Improvements in the Boolean Domain. – Cambridge Scholars Publishing, 2018. – P. 175 –198.
7. Pottosin, Yu. Parallel decomposition of a system of partial Boolean functions / Yu. Pottosin // Tomsk State University Journal of Control and Computer Science. – 2018. – No. 45. – P. 83 – 91 (in Russian).
8. Pottosin, Yu. V. Finding maximal complete bipartite subgraphs in a graph / Yu. V. Pottosin // Automation of Logical Design of Discrete Systems. – Minsk : Institute of Engineering Cybernetics of National Academy of Sciences of Belarus, 1991. – P. 19 – 27 (in Russian).
9. Pottosina, S. Finding maximal complete bipartite subgraphs in a graph / S. Pottosina, Yu. Pottosin, B. Sedliak // J. Applied Mathematics. – 2008. – Vol. 1. – No. 1. – P. 75 – 81.

ПРОЦЕССОР ХЭШ-ФУНКЦИИ SALSA20/8

Станкевич А. В., Петровский Н. А., Качинский М. В.

Кафедра электронных вычислительных средств,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {stankevich, nick.petrovsky, kachinsky}@bsuir.by

Salsa20 является криптосистемой для поточного шифрования [1]. Хэш-функция Salsa20 используется в указанной криптосистеме для получения псевдослучайных чисел, которые потом поразрядно складываются по модулю два с открытым текстом. В настоящей работе рассматривается аппаратная реализация хэш-функции Salsa20/8 применительно к системе майнинга криптовалюты Litecoin.

ВВЕДЕНИЕ

Одним из главных отличий криптовалюты Litecoin от ранее появившейся Bitcoin является использование функции Scrypt [2] вместо SHA-256 для доказательства выполненной работы. Основной целью использования Scrypt является усложнение реализации майнинга криптовалюты за счет увеличения вычислительной сложности процесса майнинга и необходимости использования в этом процессе памяти. Благодаря таким особенностям алгоритма предполагалось усложнить разработку и увеличить стоимость ASIC (application-specific integrated circuit) для процесса майнинга криптовалюты.

1. РЕАЛИЗАЦИЯ ХЭШ-ФУНКЦИИ

Функция Salsa20/8 основывается на преобразовании **quaterraund** над четырьмя 32-разрядными словами. Алгоритм Salsa20/8, описанный на языке C, приведен ниже [2]:

Листинг 1 – Описание алгоритма Salsa20/8

```
#define R(a,b) ((a << b)|(a >> (32 - b)))
#define C(y,a,b,c,d) y[a]^= R(y[b]+y[c],d);
void salsa20_word_specification(
    uint32 out[16],uint32 in[16]) {
int i; uint32 x[16];
for (i = 0; i < 16; ++i) x[i] = in[i];
for (i = 8; i > 0; i -= 2) {
    C(x, 4, 0, 12, 7); C(x, 8, 4, 0, 9);
    C(x, 12, 8, 4, 13); C(x, 0, 12, 8, 18);
    C(x, 9, 5, 1, 7); C(x, 13, 9, 5, 9);
    C(x, 1, 13, 9, 13); C(x, 5, 1, 13, 18);
    C(x, 14, 10, 6, 7); C(x, 2, 14, 10, 9);
    C(x, 6, 2, 14, 13); C(x, 10, 6, 2, 18);
    C(x, 3, 15, 11, 7); C(x, 7, 3, 15, 9);
    C(x, 11, 7, 3, 13); C(x, 15, 11, 7, 18);
    C(x, 1, 0, 3, 7); C(x, 2, 1, 0, 9);
    C(x, 3, 2, 1, 13); C(x, 0, 3, 2, 18);
    C(x, 6, 5, 4, 7); C(x, 7, 6, 5, 9);
    C(x, 4, 7, 6, 13); C(x, 5, 4, 7, 18);
    C(x, 11, 10, 9, 7); C(x, 8, 11, 10, 9);
    C(x, 9, 8, 11, 13); C(x, 10, 9, 8, 18);
    C(x, 12, 15, 14, 7); C(x, 13, 12, 15, 9);
    C(x, 14, 13, 12, 13); C(x, 15, 14, 13, 18); }
for (i = 0; i < 16; ++i) out[i]=x[i]+in[i];
}
```

Две соседние строки в приведенном алгоритме описывают **quaterraund**. Внутренний вычислительный цикл образует **doubleround**, состоящий из восьми **quaterraund**. Общее количе-

ство тактов, необходимых для вычисления функции Scrypt при аппаратной реализации, будет существенно зависеть от числа тактов, требуемых для вычисления функции Salsa20/8, поскольку эта функция многократно используется во внутренних циклах алгоритма. Поэтому требуется выбрать вариант архитектурного решения, обеспечивающий максимально возможную производительность при минимально возможных аппаратных затратах с учетом того, что функция ScryptROMix функции Scrypt может вычисляться только итеративно (организовать параллельное выполнение двух групп циклов функции ScryptBlockMix [2] нельзя без очень больших аппаратных затрат).

Анализ алгоритма показывает, что каждый **quaterraund** над четырьмя 32-разрядными словами может выполняться независимо от других, следовательно, вычислительный процесс можно распараллелить. Следует учесть, что вторая половина выражений внутреннего цикла (16 последних выражений вида $x[j] \wedge= R(x[i]+x[k], m)$) может вычисляться только после завершения вычислений первой половины. Рассмотрим варианты возможных архитектурных решений аппаратной реализации функции Salsa20/8.

1. Итеративная архитектура на уровне вычислений выражения вида $x[j] \wedge= R(x[i]+x[k], m)$. В этом случае все вычисления будут проводиться на одном процессорном ядре, на вход которого поочередно мультиплексируются соответствующие входные данные (три операнда) (рисунок 1). Если вычисление приведенного выражения выполнить за один такт, то вычислительный процесс займет 32 такта на один цикл (из четырех) алгоритма или всего $32 \times 4 = 128$ тактов. После этого надо выполнить завершающее сложение (еще один такт). Весь вычислительный процесс займет 129 тактов.

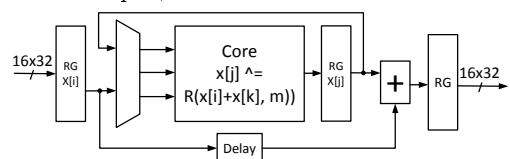


Рис. 1 – Итеративная архитектура

2. Итеративная архитектура на уровне quatteraund. Если вычисления quatteraund реализовать за один такт, то такое решение без учета завершающего сложения сократит количество повторений по сравнению с предыдущим решением в 4 раза. Длительность полученного такта должна быть меньше, чем длительность четырех тактов для первого варианта за счет отсутствия дополнительной задержки на регистре результата (вместо четырехкратного запоминания результата будет однократное). Общее количество тактов с учетом завершающего сложения $32 + 1 = 33$.

3. Параллельно-итеративная архитектура на уровне четырех quatteraund (рисунки 2, 3). В этом случае будет выигрыш по числу тактов без учета завершающего сложения по сравнению с вариантом 2 в четыре раза. Общее количество тактов 9.

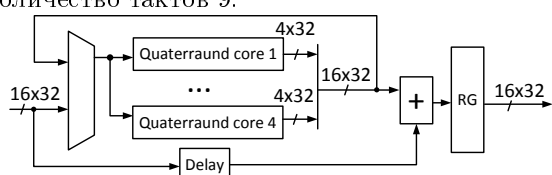


Рис. 2 – Параллельно-итеративная архитектура

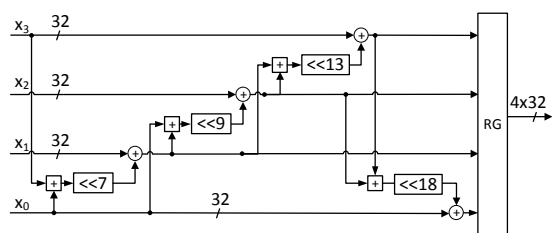


Рис. 3 – Ядро quatteraund

4. Конвейерная реализация. На одной ступени конвейера можно реализовать вычисления выражения $x[j] \wedge = R(x[i] + x[k], m)$ либо quatteraund. В первом случае будет меньше длительность такта, во втором случае будет меньше ступеней конвейера.

5. Параллельно-конвейерная реализация. В этом случае на одной ступени конвейера можно реализовать процессор quatteraund и параллельно будут работать четыре таких конвейерных вычислителя. По сравнению с 3 вариантом затраты будут в несколько раз выше (менее, чем в 8 раз за счет входного мультиплексора в итеративной архитектуре), но за счет отсутствия входного мультиплексора будет некоторый выигрыш по уменьшению длительности такта.

6. Итеративная параллельно-конвейерная реализация. Содержит четыре параллельных вычислителя на уровне quatteraund с реализацией внутри quatteraund четырех ступеней конвейера для вычислений выражения вида $x[j] \wedge = R(x[i] + x[k], m)$.

С учетом того, что функция ScryptROMix будет вычисляться итеративно, первые ступени всех конвейерных реализаций будут простаивать

до тех пор, пока на выходе конвейера не будет получен результат преобразования. В таком решении будет значительное увеличение аппаратных затрат, пропорциональное числу ступеней конвейера, однако за счет удаления входного мультиплексора будет некоторый выигрыш по сравнению с итеративной архитектурой по длительности такта.

Проведенный анализ позволяет выбрать для реализации функции Salsa20/8 параллельно-итеративный вариант архитектуры 3 как обеспечивающий достаточно высокую производительность по сравнению с другими итеративными архитектурами при относительно небольших аппаратных затратах по сравнению с конвейерными вариантами. Предлагается вычислять quatteraund за один такт, поскольку длительность одного такта будет меньше суммы длительностей четырех тактов (вычисления выражения вида $x[j] \wedge = R(x[i] + x[k], m)$ реализуются последовательно) на величину трех задержек переключения регистров для хранения результатов плюс дополнительно при реализации на FPGA добавится задержка на трассировочных ресурсах кристалла.

Проведено прототипирование процессора Salsa20/8 на кристалле XC7Z020-2CLG484 для сравнения с результатами реализации [3]. Характеристики предлагаемой реализации после процедуры синтеза с использованием средств ISE 14.7 приведены на рисунке 4.

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	1552	106400	1%
Number of Slice LUTs	2321	53200	4%
Number of fully used LUT-FF pairs	1424	2449	58%
Number of bonded IOBs	1028	200	514%
Number of BUFG/BUFGCTRLs	1	32	3%

Рис. 4 – Аппаратные затраты процессора Salsa20/8

Число тактов для получения результата – 9. Оценка частоты после процедуры синтеза – 163 МГц.

В реализации [3] указывается, что общее число тактов для вычисления Salsa20/8 равно 34. Полная реализация Scrypt [3] имеет рабочую тактовую частоту 120 МГц. Предлагаемая реализация при сравнимой тактовой частоте требует для получения результата Salsa20/8 почти в 4 раза меньше тактов (9 тактов вместо 34).

1. Daniel J. Bernstein. Salsa20 specification. – [Электронный ресурс]. – Режим доступа: <http://cr.yp.to/snuffle/spec.pdf>. Date of access: 09.10.2019.
2. RFC7914. The script Password-Based Key Derivation Function. – [Электронный ресурс]. – Режим доступа: <https://tools.ietf.org/html/rfc7914>. Date of access: 09.10.2019.
3. Implementation and Analysis of Scrypt Algorithm in FPGA. – [Электронный ресурс]. – Режим доступа: <https://alpha-t.net/wp-content/uploads/2013/11/Alpha-Technology-Scrypt-Analysis-on-FPGA-proof-of-concept.pdf>. Дата доступа 09.10.2019.

ИЗВЛЕЧЕНИЕ ЛОГИЧЕСКИХ УРАВНЕНИЙ ИЗ ОПИСАНИЯ КМОП СХЕМЫ НА SPICE

Черемисинов Д. И., Черемисинова Л. Д.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: cher@newman.bas-net.by

Рассматривается проблема извлечения логических уравнений из описания на SPICE КМОП схемы транзисторного уровня. Задача может служить одним из этапов задачи преобразования плоской КМОП схемы из транзисторов в формате SPICE в иерархическую схему. Последняя задача возникает при верификации лейаута СБИС путем сравнения исходного описания для синтеза транзисторной схемы со схемой, восстановленной из топологии, а также при перепроектировании (reengineering) схем. Описывается метод, реализованный в виде программы на языке C++, распознавания подсхем, являющихся сетями логических КМОП-вентилей, и формирования логических уравнений, задающих функции выходов сети.

ВВЕДЕНИЕ

SPICE (Simulation Program with Integrated Circuit Emphasis – программа моделирования, ориентированная на интегральные микросхемы) – известный симулятор для аналоговых и смешанных (аналого-цифровых) схем. Язык SPICE предоставляет средства для моделирования всех видов электронных устройств. В этом симуляторе можно моделировать все типы независимых и зависимых источников сигналов (прямоугольные, синусоидальные, кусочно-линейные и т. д.). Это позволяет описать любую схему с использованием надлежащих компонентных моделей, и симуляция описания дает очень реалистичное поведение схемы. Однако, моделирование КМОП схем на уровне транзисторов, которое обеспечивает SPICE, позволяет анализировать схемы с относительно небольшим числом транзисторов (порядка нескольких десятков тысяч), что лимитирует использование этого метода для анализа современных СБИС.

Автоматическое распознавание структуры более высокого уровня абстракции в описании схемы транзисторного уровня может быть использовано для многих задачах проектирования интегральных микросхем. Современные интегральные схемы могут содержать более сотни млн. транзисторов, и примерно такое же количество соединений между ними. Для выполнения процедуры анализа за приемлемое время можно построить по электрической схеме иерархическое структурное описание, в котором некоторые блоки заданы на логическом уровне. Блоки, являющиеся сетями логических вентилях, могут быть описаны на языке более высокого уровня, что позволит увеличить скорость моделирования. Это в свою очередь позволит анализировать схемы большей размерности. Например, Electronic Workbench MultiSim [1] позволяет выполнять совместное моделирование VHDL и Verilog с симуляцией SPICE блоков в смешанном режиме. Это позволяет имитировать логические блоки вместе с другими компонентами схемы.

I. РАСПОЗНАВАНИЕ ЛОГИЧЕСКИХ ВЕНТИЛЕЙ В СХЕМЕ ИЗ КМОП-ТРАНЗИСТОРОВ

В статье [2] описывается метод декомпиляции КМОП схемы из транзисторов – замены представления схемы на низком (транзисторном) уровне более высокоуровневым ее представлением (на уровне логических элементов). Исходными данными для программы служит плоский нет-лист КМОП схемы в формате SPICE, имя головной схемы и имена цепей питания. Результатом является иерархическое SPICE-описание, в которое включены модели всех идентифицированных КМОП-вентилей. Для построения иерархического структурного описания в схеме выделяются наборы взаимосвязанных транзисторов в качестве отдельных компонентов. После замены подсхем из транзисторов элементами описания схемы становится двухуровневым. Все шаги алгоритма декомпиляции КМОП схемы из транзисторов выполняются за линейное время от размерности исходных данных, поэтому программа имеет достаточное быстродействие, чтобы обрабатывать схемы из 100 тыс. транзисторов за несколько минут на персональной ЭВМ.

Технология КМОП позволяет реализовать функциональные элементы схемы различными способами. Существует множество «стилей» для реализации логических элементов: static, dynamic, Domino, CVSL, pass transistor logic [3]. Простейшая цифровая схема – это передаточный элемент из одного МОП-транзистора, который обеспечивает управляемую передачу двоичного сигнала. Это пассивный элемент, поскольку он не обеспечивает усиление входного (коммутируемого) сигнала. Усиления двоичных сигналов обеспечивает схема (КМОП-вентиль), в которой потенциал выхода вентиля «подтягивается» к потенциалу шины «земли» или к потенциалу шины питания. Блок, «подтягивающий» потенциал выхода вентиля к цепи нулевого потенциала называется pull-down network, соответственно, блок, «подтягивающий» потенциал выхода вентиля к цепи питания – это pull-up network.

Стили логических элементов различаются архитектурой этих блоков. В любом случае логический вентиль – это группа транзисторов, соединённых по постоянному току (channel-connected component – CCC) [4]. В программе [2] распознаются только логические вентиля статического стиля, как самого распространённого.

II. ФОРМИРОВАНИЕ СЕТЕЙ ЛОГИЧЕСКИХ ВЕНТИЛЕЙ

В формате SPICE электрические схемы состоят из элементов, которые соединены друг с другом цепями, и естественной формальной моделью описания схемы является помеченный неориентированный двудольный граф. Одну долю составляют выводы элементов и порты схемы, а другую – соединения между выводами, т.е. цепи. Метод декомпиляции КМОП схемы из транзисторов [2] строит схему, в которой для каждого логического вентиля известна логическая функция, реализуемая на его выходе.

Сетью логических вентиляей называют такую модель дискретного устройства, которая отражает его внутреннее строение с точностью до функций, выполняемых его элементами. Сеть образуется путем соединения вентиляей: указываются линии связи, по которым выходные сигналы одних вентиляей поступают на входные полюсы других элементов. Задать логическую сеть – значит указать ее входные и выходные полюсы, структуру связей между вентилями и булевы функции, реализуемые этими элементами. Моделью сети логических вентиляей является ориентированный ациклический граф. Каждой вершине графа соответствует некоторая переменная. Переменные, соответствующие входам сети, называются входными, а соответствующие выходам сети – выходными. Переменные, соответствующие остальным вершинам сети (выходам вентиляей), называются промежуточными.

Графы сетей логических вентиляей в двухуровневом SPICE описании формируются в процессе нахождения компонент связности в двудольном графе SPICE описания. Каждая компонента связности в двудольном неориентированном графе трансформируется в ориентированный граф логической сети. Трансформация осуществляется в процессе обхода графа двухуровневого SPICE описания по цепям входов и выходов логических вентиляей.

Этот обход осуществляется методом поиска в ширину (breadth-first search, BFS). Поиск в ширину позволяет построить сеть, ранжированную лексикографически. К нулевому рангу относятся входы сети, к первому рангу относятся промежуточные вершины, у которых входами вентиляей являются вершины нулевого ранга. Входами элементов i -го ранга являются выходы элементов $i-1$ ранга и возможно входные полюсы и выходы элементов ранга меньше $i-1$.

III. ПОСТРОЕНИЕ УРАВНЕНИЙ

От лексикографически упорядоченной сети логических вентиляей легко перейти к формулам логических уравнений, задающих функцию выхода выходных полюсов сети. Аналитическая запись системы передаточных функций сети логических вентиляей строится путем последовательной подстановки выходных функций вентиляей (начиная с элементов первого ранга). При этом в формулу функции, реализуемой рассматриваемым вентиляем, вместо входных переменных вентиля подставляются переменные входных полюсов сети или формулы выходных функций вентиляей, выходы которых соединены с входами рассматриваемого вентиля.

Описанный метод формирования сетей логических элементов использован в программе, преобразующей двухуровневое SPICE описание в трехуровневое, в котором элементами служат найденные сети логических вентиляей. В целом разработанная компьютерная программа для автоматического извлечения иерархии из списка соединений цифровой КМОП схемы транзисторного уровня выполняет следующую последовательность шагов: 1) анализ исходного описания SPICE и построение хеш-таблицы для хранения двудольного цветного графа схемы; 2) факторизация двудольного цветного графа на сети связанных по постоянному току транзисторов (CCC); 3) распознавание правильных CCC, нахождение для них соответствующих логических функций в форме алгебраической формулы и построение хеш-таблицы для хранения экземпляров КМОП вентиляей; 4) генерирование двухуровневого описания SPICE; 5) построения сетей из вентиляей двухуровневого описания; 6) генерирование трехуровневого описания SPICE, в котором элементами служат найденные сети логических вентиляей и построение формул логических уравнений, задающих функцию выхода выходных полюсов сети.

СПИСОК ЛИТЕРАТУРЫ

1. Moon Gi Seok, Daejin Park, Geun Rae Cho, Tag Gon Kim Framework for simulation of the Verilog/SPICE mixed model: Interoperation of Verilog and SPICE simulators using HLA/RTI for model reusability // 22nd International Conference on Very Large Scale Integration, VLSI-SoC, Playa del Carmen, Mexico, October 6-8, 2014. pages 1-6, 2014.
2. Черемисинов Д.И., Черемисинова Л.Д. Декомпиляция КМОП схемы из транзисторов в формате SPICE // Проблемы разработки перспективных микро- и нанoeлектронных систем. – 2018. Сб.трудов / под общ. ред. акад. РАН А.Л. Стемковского. – М.: ИП-ПМ РАН, 2018. Часть 1. – С. 2–8.
3. Rabaey, J. M., Chandrakasan A., Nikolic B. Digital Integrated Circuits – Prentice Hall Press, 2008. – P. 702.
4. Bushnell M., Agrawal Vishwani. Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI – Springer Science Business Media, 2006. – 690 p.

АНАЛИЗ СИММЕТРИЧНЫХ ПУТЕЙ ФИЗИЧЕСКИ НЕКЛОНИРУЕМОЙ ФУНКЦИИ НА FPGA

Шамына А. Ю., Иванюк А. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: shamyna@bsuir.by, ivaniuk@bsuir.by

Известные варианты реализации физически неклоняемой функции (ФНФ) типа арбитр базируются на синтезе конфигурируемых симметричных путей, обеспечивающих уникальную трансляцию тестовых сигналов для каждого изделия. В настоящей статье производится анализ зависимости основных характеристик ФНФ от количества структурных элементов блока симметричных путей (БСП) без учета влияния схемы арбитра на полученные результаты.

ВВЕДЕНИЕ

Для защиты цифровых устройств от нелегального копирования, а также для реализации генераторов случайных числовых последовательностей и других практических приложений широко используются ФНФ. Формально любую ФНФ можно описать как функцию $PUF(C) = R$, где C определяет множество запросов, а R – множество ответов [1]. Однако такие характеристики ФНФ, как уникальность и случайность, не являются идеальными, что заставляет искать новые способы улучшения этих свойств. Важнейшим критерием оценки стабильности ФНФ является ее устойчивость к изменениям внешних условий и сохранение собственных свойств с течением времени. С этой точки зрения представляет интерес для изучения ФНФ типа арбитр [2,3]. Наибольшее влияние на основные для данного типа ФНФ свойства оказывает БСП. В проведенной работе изучено влияние количества звеньев БСП на прохождение тестовых сигналов через него, дана оценка зависимости от этого конечных свойств ФНФ.

I. РЕАЛИЗАЦИЯ ФНФ ТИПА АРБИТР И ЕЕ ПАРАМЕТРИЧЕСКОЙ МОДЕЛИ

Классическая реализация ФНФ типа арбитр подразумевает наличие трех последовательно соединенных составных компонентов (рис. 1): генератора тестовых сигналов (ГТС), блока симметричных путей и арбитра. БСП состоит из N узлов, каждый из которых имеет два входа для тестового сигнала, один управляющий вход, соответствующий одному разряду запроса и два выхода тестового сигнала. Значением сигнала с управляющего входа возможно определение двух конфигураций передачи тестового сигнала через звено БСП: прямой и перекрестной. При N звеньях возможны 2^N различных реализаций двух симметричных путей, что характеризует данную ФНФ как сильную.

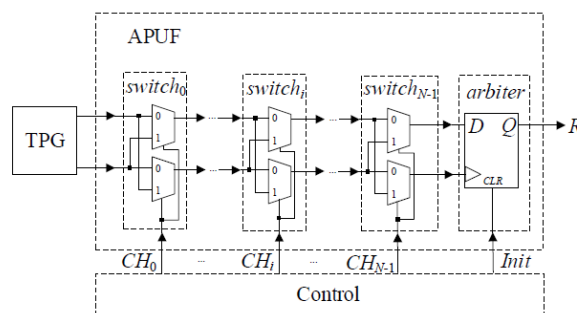


Рис. 1 – Структурная схема ФНФ типа арбитр

Было реализовано несколько вариантов ФНФ типа арбитр с различным числом структурных звеньев в БСП. Схемы ФНФ и тестовые модули были созданы на языке VHDL с использованием САПР Xilinx ISE 14.7. Параметрическая модель устройства была создана средствами САПР с учетом взаимного расположения и индивидуальных задержек элементов на кристалле FPGA Xilinx Artix-7. Отдельно следует отметить, что для генерации псевдослучайных тестовых последовательностей использовался LFSR с примитивным порождающим полиномом, степень которого соответствует требуемой разрядности запроса. LFSR был реализован функционально в testbench-компоненте для исключения влияния схемы LFSR на результаты моделирования.

II. ОЦЕНКА ПРОХОЖДЕНИЯ ТЕСТОВЫХ СИГНАЛОВ ЧЕРЕЗ БСП

Ключевой характеристикой ФНФ типа арбитр является разница между фронтами тестовых сигналов на выходах последнего узла БСП. Моделирование работы тестовых схем производилось с использованием симулятора ISE Simulator. Исследование было выполнено для моделей ФНФ с различным числом звеньев БСП ($N = 8$, $N = 16$, $N = 32$). Для $N = 8$ и $N = 16$ осуществлялся полный перебор возможных значений запроса ФНФ (2^8 и 2^{16} соответственно). Из-за высокой сложности полного пе-

ребора тестовых векторов для схемы с $N = 32$ в качестве генератора М-последовательности был выбран LFSR, с использованием которого было сгенерировано 2^{16} тестовых запросов. На рисунке 2 изображены три графика отсортированных по возрастанию значений временных разниц, наблюдаемых на выходах схемы БСП для трех реализаций: $N = 8$, $N = 16$ и $N = 32$ звеньев БСП. На оси абсцисс представлены не сами значения запросов, а их порядковые номера C_{index} . Выходы последнего звена БСП обозначены как x_{n-1} и y_{n-1} . Отрицательные значения $\Delta(x_{n-1}, y_{n-1})$ на графиках означают, что фронт тестового сигнала с выхода x_{n-1} пришел раньше, чем с выхода y_{n-1} .

Графики значений $\Delta(x_{n-1}, y_{n-1})$ для БСП $N = 32$ являются более симметричными относительно двух координатных осей. Кроме того, большие абсолютные значения разниц прохождения фронтов являются более предпочтительными, т.к. в случае использования в качестве арбитра последовательных схем минимальная разница чревата переходом арбитра в метастабильное состояние и, как следствие, меньшей стабильностью ФНФ. Обусловлено это тем, что при большем числе звеньев БСП увеличивается длина симметричных путей. Также был выполнен частотный анализ более быстрого прохождения каждой из копий тестового сигнала. Так, при $N = 32$ частота более быстрого появления фронта тестового сигнала на выходе x_{n-1} примерно равна этой же величине для выхода y_{n-1} (0.49 и 0.51 соответственно). В то время как для схем с меньшим N ($N = 8$ и $N = 16$) характерна большая асимметрия этих значений: при $N = 16$ (0.4 и 0.6) и при $N = 8$ (0.45 и 0.55). Диапазоны результирующих значений $\Delta(x_{n-1}, y_{n-1})$ при меньшем N также обладают большей асимметрией относительно нулевого значения. При $N = 32$ диапазон равен $[-10665; 10300]$ пс, а для $N = 8$ и $N = 16$ составляет $[-1757; 1561]$ пс и $[-2720; 2210]$ пс

соответственно. Важным преимуществом реализации ФНФ типа арбитр является минимальное число околонулевых значений разниц прохождения фронтов тестового сигнала, что выражается меньшей пологостью графика относительно оси абсцисс. Как видно из рисунка 2, наименьшее число околонулевых значений $\Delta(x_{n-1}, y_{n-1})$ характерно для $N = 32$.

III. ЗАКЛЮЧЕНИЕ

В ходе проведенной работы были изучены зависимости основных характеристик ФНФ от количества структурных элементов блока симметричных путей без учета влияния схемы арбитра на полученные результаты. Моделирование осуществлялось с использованием параметрических моделей ФНФ типа арбитр с различным числом звеньев в БСП. Из полученных в данной работе результатов можно сделать вывод, что наиболее целесообразным является использование ФНФ типа арбитр с большим числом звеньев БСП, т.к. увеличение их количества улучшает характеристики ФНФ, такие как стабильность, уникальность и случайность. Однако полученные результаты нуждаются в многократной верификации на реальном оборудовании.

СПИСОК ЛИТЕРАТУРЫ

1. Secure System Design and Trustable Computing / Springer; editors Ch.-H. Chang, M. Potkonjak. — Switzerland, 2016. — p.537.
2. Иванюк, А. А. Особенности реализации симметричных путей ФНФ типа арбитр на ПЛИС / А. А. Иванюк // Информационные технологии и системы 2018 (ИТС 2018) = Information Technologies and Systems 2018 (ITS 2018) : материалы международной научной конференции, Минск, 25 октября 2018 г. / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : Л. Ю. Шилин [и др.]. — Минск, 2018. — С. 156 - 157.
3. Заливако, С. С. Метод увеличения стабильности физически неклонированной функции типа "арбитр" / С. С. Заливако, А. А. Иванюк, В. П. Клыбик // Информатика. — 2017. — №1(53). — С. 31 - 43.

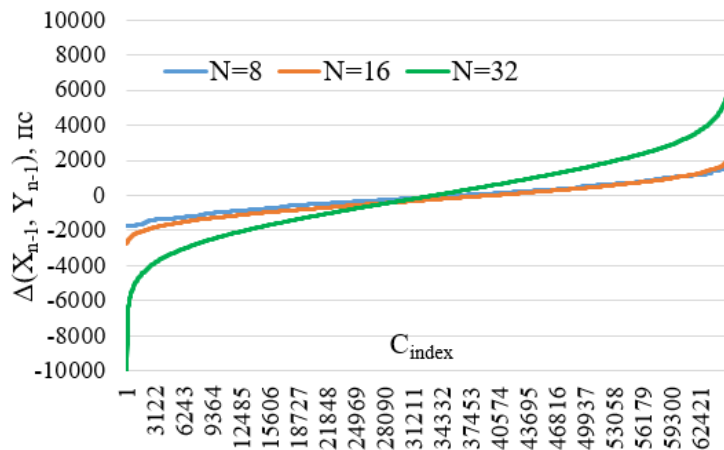


Рис. 2 – Графики значений $\Delta(x_{n-1}, y_{n-1})$ для различных реализаций БСП

Секция «Математическое
моделирование и компьютерная
графика»

Section «Mathematical modeling and
computer graphics»

КОНЦЕПЦИЯ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ ПАВОДОЧНОГО СТОКА

Амельченко Н. П., Шаронова Е. И., Шаронова О. А.
Кафедра вычислительных методов и программирования,
Кафедра инженерной и компьютерной графики
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: seizv@bsuir.by

Эффективное моделирование паводков может быть выполнено исключительно с помощью разработки базы данных о реках. Основа информации - данные о высоте и пропускной способности реки. Коррелируя эти данные с количеством осадков можно спрогнозировать риск наводнений и, как следствие, эвакуации. Проблема является актуальной, т.к. наводнение, как серьезное стихийное бедствие, затрагивает практически все части мира, включая развитые страны. Эти опасности и потери могут быть предотвращены или минимизированы благодаря предоставлению информации о риске наводнения с помощью карт моделирования паводочного стока.

ВВЕДЕНИЕ

Компьютерное моделирование паводочного стока получило развитие, как в нашей стране, так и за рубежом с середины 60-х гг. прошлого столетия. Работы велись преимущественно в двух направлениях: имитация движения воды в речном русле и на склонах речного водосбора. Основу моделей первого направления (речные модели) составляли уравнения неустановившегося движения Навье-Стокса, Сен-Венана; в моделях второго направления (бассейновые модели), кроме указанных уравнений, широко применялись различные модификации уравнений математической физики (прежде всего уравнение диффузии) и различные эмпирические зависимости. Геоинформационные системы в настоящее время – это современная интегрированная система, отвечающая требованиям информатизации общества, применяемая во всех направлениях. Она способствует решению управленческих, экономических задач, основываясь на методах оптимизации и прогнозирования. Основное назначение ГИС - формирование знаний и явлений, процессах на Земле и дальнейшее применение их для решения необходимых задач во всех сферах жизнедеятельности человека.

СОДЕРЖАНИЕ ДОКУМЕНТА

Полезную информацию и руководство по выбору и применению концептуальных моделей в различных гидрологических ситуациях можно найти в документации нескольких международных проектов ВМО, выполненных с 1970-х годов, а именно: а) взаимное сравнение концептуальных моделей, используемых при оперативном гидрологическом прогнозировании (WMO, 1987); б) взаимное сравнение моделей стока весеннего половодья (WMO, 1986); в) имитированное взаимное сравнение гидрологических моделей в режиме реального времени (WMO, 1991a). Многие пакеты гидрологическо-

го программного обеспечения были разработаны научно-исследовательскими институтами и коммерческими компаниями для персональных компьютеров и автоматизированных рабочих мест, использующих системы MS Windows, UNIX и LINUX. Многие модели оборудованы интерфейсом ГИС. Геоинформационные системы (ГИС) успешно используются для визуализации масштабов наводнений, а также для анализа карт наводнений с целью составления карт оценки ущерба от наводнений и карты риска наводнений. Часто термин понимают и в более узком смысле: ГИС – это инструмент (программа) для поиска, анализа и редактирования цифровых карт, позволяющий искать дополнительную информацию о предметах. Трехмерные программные модули в геоинформационной системе позволяют создавать в среде трехмерной местности объекты любой сложности: архитектурные постройки, дорожные конструкции, группы деревьев, и т.д. Трехмерное моделирование дает возможность наилучшим образом описывать реальную местность, объекты окружающего мира и их взаимное расположение. Отличие трехмерных ГИС от трехмерных интерактивных тренажеров или симуляторов в том, что в ГИС любой трехмерный объект имеет географические координаты, то есть осуществляется непосредственная привязка к местности. При этом объект можно выделить мышью, пространственно сравнить с другими объектами, связать с ним базу данных любой сложности, и таких объектов можно создать сколько угодно. С позиций визуальных эффектов трехмерная ГИС и трехмерная игра могут мало отличаться друг от друга, но содержательное отличие значительно. Трехмерное моделирование можно использовать для более эффективного представления района паводков, с учетом рельефа реальной местности. 3Dмоделирование также успешно применяют в конструкторских проектах при создании различных моделей (элементов). Кроме того, оно легко заменит

натуральное макетирование, например, позволит создать модель непосредственно на основе цифровых карт местности. Трехмерное изображение на плоскости, в отличие от двумерного, включает построение геометрической проекции объемной модели на плоскость с помощью специализированных программ (рис. 1). Для создания трехмерной карты требуется произвести привязку трехмерного объекта к его топографическому знаку. Но первоначально необходимо создать сам трехмерный объект. Наличие хорошо сделанной модели объекта позволяет быстро, без «зависания», создавать трехмерную модель любого района. Создание моделей для классификатора ГИС осуществляется с применением программ трехмерной графики. 3D-графика — вид компьютерной графики, визуальное отображение трехмерной сцены или объекта на экране монитора или какого-либо другого устройства. Для создания трехмерного объекта для ГИС возможно использовать следующие программные продукты:

- Autodesk — 3ds Max;
- Blender Foundation — Blender;
- Trimble Navigation — SketchUp.

Трехмерную модель можно построить как для всего отображаемого на двумерной карте района, так и для любого его фрагмента. Доступен просмотр и изменение семантики и метрики для выбранного объекта. При изменении списка данных электронной карты, состава объектов изменяется вид трехмерной модели. Имеется три вида перемещения по трехмерной модели: ручную, по выбранному объекту и в свободном поле по заданной траектории. В трехмерной модели есть возможность сохранения текущего изображения в BMP-файл. Также можно записать AVI-файл с перемещением по трехмерной модели и с изменением ее характеристик. Перемещение по трехмерной модели и по двумерной карте может быть синхронизировано. Поэтому имеющаяся в ГИС задача подключения GPS-приемника делает возможным определение местоположения движущегося объекта как на двумерной карте, так и на трехмерной модели. Входными данными для этой задачи являются данные в формате NMEA0183 (в текстовом ASCII-виде), принимаемые с параллельного порта компьютера, к которому подключено устройство типа GPS-приемника, или с удаленного GPS-устройства через протокол GPRS. Имеется возможность загрузки векторных, растровых и матричных карт из различных форматов, печати загруженных данных. Картам в ГИС отведено особое место. Процесс создания карт в ГИС намного более прост и гибок, чем в традиционных методах ручного или автоматического картографирования. Он начинается с создания базы данных. В качестве источника получения исходных данных можно

пользоваться и оцифровкой обычных бумажных карт. Основанные на ГИС картографические базы данных могут быть непрерывными (без деления на отдельные листы и регионы) и не связанными с конкретным масштабом. На основе таких баз данных можно создавать карты (в электронном виде или как твердые копии) на любую территорию, любого масштаба, с нужной нагрузкой, с ее выделением и отображением требуемыми символами. В любое время база данных может пополняться новыми данными (например, из других баз данных), а имеющиеся в ней данные можно корректировать по мере необходимости. В крупных организациях созданная топографическая база данных может использоваться в качестве основы другими отделами и подразделениями, при этом возможно быстрое копирование данных и их пересылка по локальным и глобальным сетям.

ЗАКЛЮЧЕНИЕ

В результате моделирования паводочного стока могут быть созданы различные программные продукты, позволяющие точно рассчитать риск наводнения и тем самым минимизировать потери в результате стихийного бедствия. Для этого могут служить как пакеты программных продуктов, так и отдельные модули, например карты опасностей, оценочные карты затопления и компьютерное 3D-моделирование отдельных процессов.



Рис. 1 – Основные функции и возможности 3D-программ

СПИСОК ЛИТЕРАТУРЫ

1. https://www.gislounge.com/A_Case_Study_on_Converting_a_Geodatabase_From_ArcGIS_for_Use_in_QGIS_JUNE_4_2019
2. Иванов В.Г., Бородин Н.Д. Основы формирования единого геоинформационного пространства специального назначения с использованием Webтехнологий // САПР и графика. № 3. 2016. С. 1820.
3. Кучмет Л.С. Математическое моделирование речного стока. - Л.: Гидрометеоздат, 1972. - 191 с.
4. <http://gistech.ru>. Сайт «ГИС Техник»
5. GIS techniques for flood modeling and flood inundation mapping [Electronic resource] /— Mode of access: www.researchgate.net/Electronic_Journal_of_Geotechnical_Engineering/. — Date of access: 14.01.2015.

ПРИМЕНЕНИЕ ДЕСКРИПТОРОВ ПРИ НАПИСАНИИ ТРАНСПАЙЛЕРА НА ПРИМЕРЕ ТРАНСПАЙЛЕРА ИЗ C# В DART

Ващилко А. П.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: alexander.armoken@gmail.com

Эта статья может помочь тем, кто пишет конвертер из одного высокоуровневого языка программирования в другой и решил реализовать правила конвертации, например, для конвертации вызовов стандартных библиотек. В качестве примеров таких языков в статье используются C# и Dart.

ВВЕДЕНИЕ

При написании конвертора одного высокоуровневого языка программирования в другой, может возникнуть необходимость интегрировать сконвертированный код в уже существующую кодовую базу. Чтобы это реализовать необходимо указать конвертору какие синтаксические или семантические конструкции нужно сконвертировать таким образом, чтобы их можно было без дополнительных ручных правок сразу же использовать. Для этого необходимо реализовать правила трансляции.

I. ПРАВИЛА ТРАНСЛЯЦИИ

Правило трансляции состоит из синтаксического и семантического описания пары языковых конструкций:

1. входной конструкции на языке C# (является ключом, по которому ищется правило);
2. образец конструкции на языке Dart получаемой на выходе программы.

Поиск правила осуществляется последовательным сравнением языковой конструкции из конвертируемого кода и языковой конструкции, описываемой ключом правила.

Наиболее гибким является использование деревьев (не обязательно синтаксических, так как их проблематично использовать, например, для описания типов) для описания языковых конструкций.

Во время реализации конвертора был использован компилятор с открытым исходным кодом Roslyn [1], поскольку он берет на себя работу по синтаксическому и семантическому анализу кода. Однако синтаксические деревья и графы, описывающие типы и члены типов, предоставляемые Roslyn, хотя и позволяют их сравнивать, но не дают возможности делать следующие вещи:

- ограничивать глубину сравнения;
- делать метки на узлах дерева;
- хранить синтаксическую и семантическую информацию в одной структуре данных.

Чтобы не реализовывать весь функционал Roslyn можно реализовать обертки над его синтаксическими деревьями и графами, описываю-

щими типы и их члены. Эти обертки в дальнейшем будут называться дескрипторами.

II. ДЕСКРИПТОРЫ

Дескрипторы представляют собой реализацию красно-зеленых деревьев [2] в случае синтаксических деревьев и неизменяемый фасад над изменяемыми данными в случае графов, описывающих тип и их члены, так как, например, при использовании шаблонов (CRTP [3]) возможны циклы в графе.

Дескрипторы необходимо использовать только для тех синтаксических конструкций, для интерпретации которых необходима масса семантической информации, для остальных можно использовать адаптеры над конструкциями Roslyn (RoslynExprAdapter). Например, выражения, описывающие цикл for в простых случаях не требует реализации отдельного условия, но бинарное выражение, описывающее сравнение двух переменных, уже требует этого. Данный подход позволяет реализовывать функционал конвертора по мере возникаемых требований.

Для остановки сравнения деревьев стоит использовать затычки (StubDescr), которые при сравнении с любым дескриптом из иерархии своего языка являются эквивалентными ему. Два дескриптора являются эквивалентными не только тогда, когда все их узлы содержат одинаковую информацию о коде, но и когда некоторые из узлов являются затычками.

На все дескрипторы можно ставить текстовые метки, которые должны быть одинаковы в двух частях правила трансляции. Метки можно сделать единственными изменяемыми полями в общем-то неизменяемых дескрипторах.

Как уже можно было понять, для каждого языка необходима своя иерархия дескрипторов, даже если они будут идентичны по своим свойствам и функциям. Это предотвратит будущие архитектурные ошибки и упростит рефакторинг.

III. ПРИМЕНЕНИЕ ПРАВИЛА ТРАНСЛЯЦИИ

Для применения правил трансляции необходимо произвести следующие операции:

1. построить дескриптор языковой конструкции;
2. последовательно сравнить полученный дескриптор с ключами правил трансляции, пока не будет найдено подходящее правило (см. рисунки 1 и 2);
3. скопировать метки из дескриптора ключа в только что построенный дескриптор (см. рис. 2);
4. используя метки и образец выходной конструкции из правила, построить выходную конструкцию (см. рис. 3). Стоит уделить внимание тому, что метки в левой части рисунка 3 точно такие же как и в правой части рисунка 2. Отвечает за консистентность меток разработчик правил, но разработчик транспайлера может помочь ему, если реализует проверку консистентности меток во время запуска транспайлера;
5. обходя граф выходной конструкции, получить строковое представление кода. Эта операция не является обязательной, если планируется использование дескриптора Dart для каких-либо иных манипуляций над кодом.

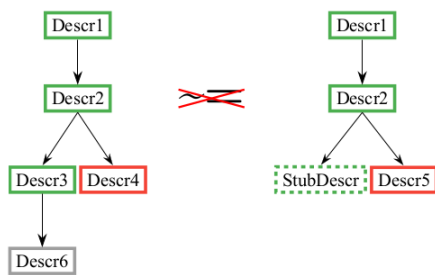


Рис. 1 – Пример неудачного сравнения дескрипторов

ЗАКЛЮЧЕНИЕ

Использование дескрипторов позволяет спроектировать гибкую архитектуру транспайлера с учетом будущих потребностей и ограниченных ресурсов в настоящем. Поскольку необязательно реализовывать дескриптор для каждого вида языковой конструкции, а только лишь для тех, что необходимы в данный момент. Особенно эффективного использования дескрипторов можно достичь при создании узкоспециализированных инструментов разработчиков, если уже есть готовые библиотеки для анализа программного кода.

СПИСОК ЛИТЕРАТУРЫ

1. Компилятор C# Roslyn [Электронный ресурс] – Режим доступа: <https://github.com/dotnet/roslyn>. – Дата доступа: 10.10.2019.
2. Persistence, Facades and Roslyn's Red-Green Trees [Электронный ресурс] / Eric Lippert – Режим доступа: <https://blogs.msdn.microsoft.com/ericlippert/>. – Дата доступа: 10.10.2019.
3. C++ Template Metaprogramming: Concepts, Tools, and Techniques from Boost and Beyond /David Abrahams, Aleksey Gurtovoy. Издательство: Addison-Wesley Professional, 2004. – 390с.

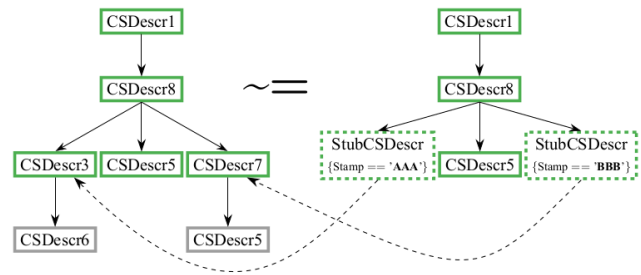


Рис. 2 – Пример удачного сравнения дескрипторов и копирования меток

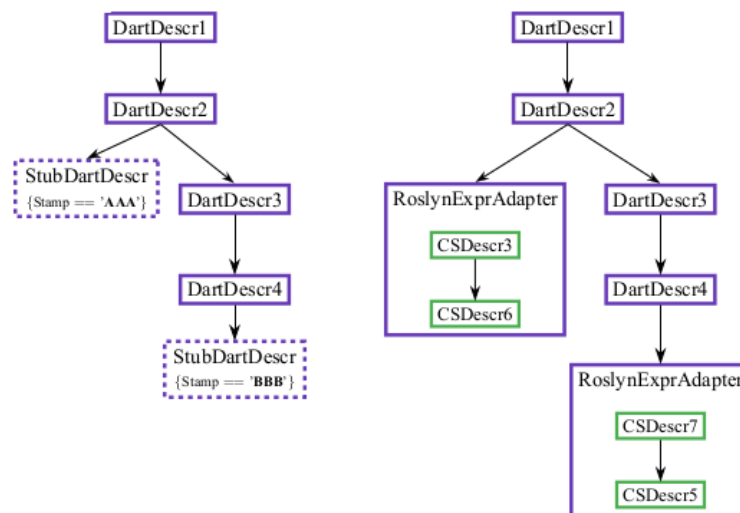


Рис. 3 – Пример вставки отмеченных дескрипторов

ПРОГРАММНО-АППАРАТНАЯ СИСТЕМА ЗАХВАТА ДВИЖЕНИЙ РУК

Видничук В. Н., Альшевский А. В., Бысов С. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: Vidnichuk@bsuir.by, nexus.alekseishshevski@gmail.com, sergbysov@gmail.com

Предложена модель системы захвата движений на основе датчиков положения в пространстве. Описан механизм получения данных и их дальнейшей обработки для использования в программном обеспечении, анализирующем параметры датчиков и обрабатывающем значения для использования программным обеспечением виртуализации движения руки человека.

ВВЕДЕНИЕ

На данный момент разработка новых интерфейсов взаимодействия пользователя с компьютерными системами является одним из ведущих направлений в области компьютерных технологий (КТ). Существующие решения в области устройств ввода, такие как клавиатура или компьютерная мышь, постепенно перестают удовлетворять потребностям быстро растущих рынков, использующих КТ. В связи с бурным развитием технологий виртуальной реальности как никогда проявляется необходимость в введении новых методов взаимодействия с виртуальными моделями. Одной из существующих задач в области цифрового захвата движений является распознавание движений кистей рук, т.к. руки человека являются самыми подвижными конечностями, что вызывает существенные сложности в построении точной модели руки пользователя в реальном времени для использования в системах виртуальной реальности.

ПРЕДЛАГАЕМАЯ РЕАЛИЗАЦИЯ

Одной из возможных реализации такой системы является использование датчиков, способных определять положение в пространстве каждой отдельно взятой точки кисти руки, что позволит точно проецировать не только хватательные движения, но и мелкую моторику руки. Подобные устройства уже существуют, но на данный момент их точность весьма ограничена ввиду того, что количество точек считывания существенно ограничено.

Цель данной работы — реализовать устройство ввода, способное проецировать модель кисти человека на основе датчиков положения в пространстве, установленных так, чтобы отслеживать изменения в положении фаланг пальцев в пространстве. Точность проецирования будет обеспечена использованием датчиков на каждой фаланге, а также на запястье, предплечье и плече. Определив точное расстояние между датчиками, установленными на руке, можно будет построить проекцию руки человека в пространстве, основываясь на анатомических параметрах каж-

дого из пальцев и положении каждой из точек в пространстве. На основе крена, рысканья и тангажа на каждой из верхних плоскостей фаланг и известного расстояния между точками считывания программное обеспечение будет иметь возможность рассчитать трехмерную модель положения кисти в пространстве, после чего эти данные могут быть использованы сторонним программным обеспечением для визуализации данной модели.

Крен - отклонение плоскости симметрии отслеживаемого объекта по вертикали, что можно трактовать как сгиб фаланги пальца. Тангаж - отклонение плоскости симметрии отслеживаемого объекта по горизонтали, что можно трактовать как поворот самого пальца, который может являться следствием поворота руки. Рысканье - угловое движение объекта относительно вертикально оси. Это можно трактовать как отклонение движения руки в горизонтальной плоскости.

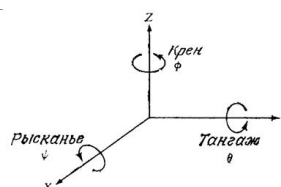


Рис. 1 – Крен, тангаж, рысканье

Для реализации данного проекта будет использован следующий набор аппаратных решений:

- Датчики MPU-9250, достаточно компактные и точные, будут использоваться для считывания положения верхних плоскостей фаланг пальцев в пространстве.
- Устройства первоначальной обработки данных на основе FPGA (ПЛИС). Собственная архитектура, созданная для конкретных задач, существенно повысит скорость работы всей системы. А также благодаря возможности перепрограммирования схемы минорное обновление устройства можно будет производить без демонтажа и замены уже существующих узлов.

- Одноплатный микрокомпьютер (напр. Raspberry Pi), чьей задачей будет анализ и подготовка полученных данных, а также отправка полученных значений ПО, выполняющему построение 3D модели в виртуальном пространстве.

MPU-9250 - девятиосевой датчик положения в пространстве. Обладая трёхосевыми магнитометром, гироскопом и акселерометром, данный датчик позволит решить множество проблем связанных с последующей фильтрацией данных, т.к. имеющийся в корпусе магнетометр позволит решить проблему дрейфа рысканья.

Передача данных между датчиками и фильтрующей FPGA-схемой будет осуществляться посредством интерфейса I2C, который уже реализован в датчиках MPU-9250. Полученные сырые данные будут обработаны с помощью реализованного на аппаратном уровне Фильтра Махони. Формула фильтра Махони выглядит следующим образом:

$$\dot{\mathbf{C}} = \mathbf{C}(\boldsymbol{\Omega} + \boldsymbol{\Omega}')$$

$$\boldsymbol{\omega}' = k_g \hat{\mathbf{g}}_{xyz} \times \mathbf{C}^T \mathbf{g}_{NED} + k_m \hat{\mathbf{m}}_{xyz} \times \mathbf{C}^T \mathbf{m}_{NED}$$

Где слагаемое в первой части является векторным произведением ускорения свободного падения на истинное ускорение. При этом второй множитель требуется преобразовать к той же системе координат, что и первый. Второе слагаемое - аналогичное произведение векторов магнитных полей. Благодаря знанию ориентации двух неколлинеарных векторов в разных системах мы можем точно установить взаимную ориентацию этих систем.

Задача фильтра будет заключаться в коррекции дрейфа тангажа, крена и рысканья для каждого из датчиков. Также благодаря фильтру можно будет нивелировать низкочастотный дрейз датчиков. Далее данные при помощи интерфейса UART будут переданы на один из вспомогательных микрокомпьютеров, чьей задачей является накопление, окончательная фильтрация и отправка данных на центральный микрокомпьютер, через который осуществляется связь со средством визуализации.

Согласно предварительным расчётам, для передачи данных между узлами системы не потребуется скорости соединения более 30 КБайт/сек, что полностью покрывается выбранными интерфейсами. Выбор I2C обусловлен наличием такового в корпусе датчиков MPU-9250, благодаря чему можно избежать монтажа дополнительных радиоэлектронных схем в ограниченных размерах. Интерфейс UART обладает достаточной простотой реализации с точки зрения FPGA-программирования, а также присутствует в одноплатном микрокомпьютере Raspberry Pi любой модификации, что дополнительно облегчит разработку.

Далее для визуализации полученных данных о движении датчиков будет использован инструментарий Unity 3D. Данное программное обеспечение обладает весомыми преимуществами для нашего проекта: наличием визуальной среды разработки и возможностью выполнения на различных платформах. Также используется модульная система компонентов, с помощью которой происходит конструирование виртуальных функциональных объектов. В отличие от механизмов наследования, объекты в Unity создаются посредством функциональных блоков, а не помещения в узлы дерева наследования. Такой подход ускоряет создание прототипов.

Алгоритм работы системы можно представить следующим образом:

- инициализация соединений и компонентов, а также их калибровка;
- поочередное считывание данных из гироскоп-акселерометров;
- предварительная фильтрация сырых данных;
- передача данных на микроконтроллер;
- отправка данных в концентратор;
- преобразование данных в сферические координаты и дополнительная фильтрация при необходимости;
- передача данных в среду трёхмерного отображения;
- интерпретация полученных сферических координат в положение виртуальных объектов в виртуальном пространстве.

ЗАКЛЮЧЕНИЕ

В итоге будет получена система, которая может снимать данные о положении рук человека в реальном пространстве и визуализировать их в виртуальной среде в реальном времени. Данную систему можно использовать в широком спектре задач: например, в кинопроизводстве (в качестве более точной замены Motion Capture), или в промышленности и медицине (для осуществления управления манипуляторами или для реабилитации).

СПИСОК ЛИТЕРАТУРЫ

1. Peterson, E. Developing tamper resistant designs with Xilinx Virtex-6 and 7 series FPGAs, San Jose, CA, USA, Oct. 2013.
2. Львовский, С. М. Набор и вёрстка в системе LaTeX / С. М. Львовский // Издательство: МЦНМО, 2006. – 448 с.
3. MPU9250 Datasheet [Электронный ресурс]. Режим доступа: <https://www.invensense.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf>
4. Unity User Manual [Электронный ресурс]. Режим доступа: <https://docs.unity3d.com/Manual/index.html>
5. Raspberry Pi Documentation [Электронный ресурс]. Режим доступа: <https://www.raspberrypi.org/documentation/>

ОЦЕНКА КАЧЕСТВА РЕЗУЛЬТАТОВ АЛГОРИТМОВ ТРЕХМЕРНОЙ РЕКОНСТРУКЦИИ СЦЕН

Головатая Е. А., Садов В. С.

Кафедра интеллектуальных систем, Белорусский государственный университет

Минск, Республика Беларусь

E-mail: katerina-golovataya@yandex.ru, sadov@bsu.by

Работа посвящена рассмотрению задачи оценки результатов трехмерной реконструкции. Особую сложность представляет анализ реконструкции систем, к которым затруднен прямой доступ или имеющих сложные конструктивные особенности. В работе приводятся подходы, позволяющие оценить качество полученных моделей.

ВВЕДЕНИЕ

Трехмерная реконструкция объектов по изображениям является важной задачей компьютерного зрения. Одним из основных применений трехмерной реконструкции является генерация объектов, которые трудно или долго моделировать вручную. Трехмерная реконструкция может применяться во многих областях, таких как, архитектура и археология, инженерия и строительство, криминалистика, картография и т.д. Одним из перспективных направлений являются медицинские приложения, позволяющие повышать точность процедур благодаря сопоставлению нескольких видов исследований, оценивать поверхности, производить планирование и моделирование хирургических вмешательств, лучевой терапии и многое другое. В то же время данный подход позволяет ускорить процесс генерации контента для графических приложений, таких как игры, киноиндустрия, виртуальные среды и виртуальный туризм. Использование подобных моделей в виртуальной реальности и симуляциях позволяет анализировать работу систем, обучать персонал, проводить эксперименты.

Одной из важных задач трехмерной реконструкции является оценка корректности полученной модели. Особую сложность представляют такие типы систем, в которых объект является труднодоступным для проведения даже отдельных прямых измерений. В работе рассматриваются подходы к анализу модели, полученной алгоритмами трехмерной реконструкции по данным медицинских видеоэндоскопических систем.

I. ПОДХОДЫ К ОЦЕНКЕ КАЧЕСТВА ПОЛУЧЕННОЙ МОДЕЛИ

Качество трехмерной модели заключается в ее точности, что соответствует степени схожести полученного результата с моделируемым объектом или сценой. Можно выделить следующие методы оценки качества трехмерной реконструкции [1]:

- сравнение физических параметров напрямую по известным объектам в кадре,

- сравнение параметров внешнего ориентирования камер по известным данным позиционирования и ориентации (например, GPS и данные гироскопа),
- анализ на основе сравнения с моделирующей средой.

Для реализации сравнения физических параметров напрямую по известным объектам в кадре необходимо для исходного исследуемого набора данных, по которому осуществляется реконструкция, иметь также возможность проведения объективных измерений находящихся в кадре объектов. При трехмерной реконструкции на основе метода связок, информацию об отдельных измерениях можно вводить в исходную оптимизируемую матрицу связок в виде линейных уравнений, отражающих зависимости между теми или иными ключевыми точками, присутствующими в матрице. Аналогичным образом может использоваться известная информация о параметрах внешнего ориентирования камеры. В частности, большинство современных мобильных устройств предоставляет возможности по добавлению метаданных о снимке, которая может включать в себя координаты GPS и углы отклонения от горизонта с гироскопов и акселерометров устройства в момент получения снимка. Результаты отдельных измерений или известные параметры внешнего ориентирования могут быть добавлены в виде дополнительных уравнений непосредственно в матрицу связок для увеличения точности оптимизации, либо использоваться после оптимизации для оценки отклонения фактического и ожидаемого значения.

При наличии набора из n измерений M_i координат точек или камеры, или углов ориентации камеры, и соответствующих им значений A_i , определенных по некоторой трехмерной модели, отдельные отклонения $\delta_i = |M_i - A_i|$ описываются случайной величиной с нормальным распределением и математическим ожиданием, равным 0. Для оценки качества реконструкции для полученных выборочных измерений может быть построен доверительный интервал отклонения относительно среднего арифметического или относительно 0. Кроме того, для таких из-

мерений в качестве абсолютной характеристики может быть использован средний квадрат ошибки.

II. ОЦЕНКА КАЧЕСТВА ТРЕХМЕРНОЙ РЕКОНСТРУКЦИИ С ИСПОЛЬЗОВАНИЕМ МОДЕЛИРУЮЩЕЙ СРЕДЫ

Основным недостатком оценки качества реконструкции на основе сравнения измерений и параметров внешнего ориентирования камер является значительное влияние систематических измерительных погрешностей, а также невозможность, в некоторых случаях, провести объективные измерения отдельных объектов или получить информацию об ориентировании с камеры. Одной из областей, в которых такая оценка затруднительна, являются видеоэндоскопические исследования.

В контексте видеоэндоскопических исследований известными объектами сцены могут выступать инструменты операционного канала эндоскопа и сама трубка эндоскопа. Кроме этого, размеры некоторых объектов самих внутренних органов могут быть дополнительно определены на основании других видов медицинских исследований: томографических (с помощью компьютерной или магнитно-резонансной томографии) или ультразвуковых (с помощью внешнего ультразвукового исследования или эндосонографии). Тем не менее, в большинстве случаев эти данные недоступны, либо погрешность соответствующих измерений является слишком высокой для объективного оценивания качества реконструкции.

Для решения этой проблемы предлагается использовать моделирующую среду, способную воссоздать окружение, отражающее основные свойства исследуемого пространства, самого объекта исследования и оптической системы. Для анализа данных видеоэндоскопических исследований разработана моделирующая среда на основе цилиндрического примитива, имитирующего внутреннюю поверхность пищевода. В модели заданы основные анатомические особенности, такие как среднестатистический диаметр, сужение и деформация отдельных частей модели [2]. Для отражения неидеально ровной поверхности производится процедурная генерация рельефа в заданных ограничениях гладкости. Для текстурирования полученных поверхностей используются алгоритмы переноса стиля на основе сверточных нейронных сетей и применяется глянцевый шейдер [3] для имитации отражающих поверхностей внутренних органов. Также в моделируемом пространстве располагается камера с заданными оптическими свойствами, соответствующим реальным системам, и источник освещения. Движение камеры задается на основании уравнение Пуассона для имитации кусочно-равномерного движения. Одновременно с камерой движется источник освещения,

при чем их направление и ориентация изменяются в установленных пределах с течением времени. Также предусмотрена возможность ручного задания дополнительных объектов, соответствующих определенным типам заболеваний. Рендеринг и получение видеоряда осуществляется на основе метода трассировки лучей стандартными возможностями движка рендеринга V-Ray.

Моделирующая среда является удобным средством проверки качества модели, так как позволяет осуществлять сравнение координат всех ключевых точек, положения и ориентации всех камер. В качестве критерия качества реконструкции могут быть использованы среднеквадратичные ошибки оценки координат ключевой точки, координат положения камеры и углов ориентации камеры. Поскольку ошибки являются взаимозависимыми и отражают разные величины, в качестве итогового критерия может использоваться некоторая взвешенная сумма, в которой веса каждой ошибки подбираются исходя из важности определения конкретного параметра для итоговой модели. Недостатком такого подхода является невозможность оценки качества реконструкции по абсолютному значению. С другой стороны, полученный критерий может использоваться для сравнения результатов реконструкции по двум различным алгоритмам, либо для сравнения влияния отдельных параметров алгоритма на точность итоговой реконструкции.

ЗАКЛЮЧЕНИЕ

Оценка качества трехмерной реконструкции по нескольким снимкам является нетривиальной задачей. Для решения этой задачи может использоваться сравнение с результатами известных измерений или параметров ориентирования камеры, однако, в отдельных задачах такая информация может быть недоступна. Для решения этой задачи предложено использовать моделирующую среду, позволяющую осуществлять визуализацию сцены с известными пространственными характеристиками, которые в дальнейшем могут использоваться для оценки качества реконструкции по снимкам, полученным из визуализированной сцены. Моделирующая среда позволяет рассматривать отдельные параметры алгоритма и исследовать их влияние на точность итоговой реконструкции, а также определять устойчивость алгоритма к различным характеристикам сцены.

СПИСОК ЛИТЕРАТУРЫ

1. Luhmann, T. Close-range photogrammetry and 3D imaging / T. Luhmann, S. Robson, S. Kyle, J. Boehm // De Gruyter STEM: 2nd ed. – 2014. – 684 p.
2. Liao, D.-H. Gastrointestinal tract modelling in health and disease / D.-H. Liao, J.-B. Zhao, H. Gregersen // WJG. – 2009. – Vol. 15, № 2. – P.169
3. Bailey, M. Graphics shaders: theory and practice / M. Bailey, S. Cunningham. – CRC Press: 2nd ed., – 2012. – 490 c.

О НЕКОТОРЫХ МЕТОДАХ КЛАССИФИКАЦИИ (ПСЕВДО)РИМАННОВЫХ ОДНОРОДНЫХ ПРОСТРАНСТВ С ПОМОЩЬЮ СИСТЕМ КОМПЬЮТЕРНОЙ МАТЕМАТИКИ

Грабко А. Н., Боровский М. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный
университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: softforlutshix@gmail.com, maksim.borovskij@yandex.ru

В случае малой размерности для изучения (псевдо)римановых однородных многообразий применяются методы компьютерной математики. В данной работе рассмотрены некоторые из них. Первый метод основан на исследовании инвариантных тензорных полей с помощью анализа структурных констант алгебры Ли группы изометрий и компонент метрического тензора. Второй метод основан на изучении пространства орбит левинвариантных римановых метрик групп Ли и анализе структурных констант базисов Милнора для (псевдо)римановых однородных многообразий.

ВВЕДЕНИЕ

(Псевдо)римановы многообразия исследовались многими математиками. Данный класс пространств содержит многообразия Эйнштейна ($r = \lambda g$) и их прямые произведения, локально симметричные пространства ($\nabla R = 0$), Риччи параллельные многообразия ($\nabla r = 0$) и конформно плоские многообразия ($W = 0$) (см. [1]).

В случае малой размерности для изучения (псевдо)римановых однородных многообразий применяются различные методы компьютерной математики. Один из них основанный на анализе структурных констант алгебры Ли группы изометрий и компонент метрического тензора, предполагает последовательное рассмотрение всех возможных типов Сегре оператора Риччи для данного пространства (см. [2]). Так же существует альтернативный подход, основанный на обобщенных базисах Милнора.

I. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Тензор Схоутена–Вейля SW (псевдо)риманова многообразия (M, g) размерности $n \geq 3$ определяется формулой:

$$SW(X, Y, Z) = \nabla_Z A(X, Y) - \nabla_Y A(X, Z)$$

где $A = \frac{1}{n-2} \left(r - \frac{sg}{2(n-1)} \right)$ – тензор одномерной кривизны (тензор Схоутена), s – скалярная кривизна. Если $n \geq 4$, то тензор Схоутена–Вейля связан с дивергенцией тензора Вейля через уравнение (см. [1]):

$$SW = -(n-3)divW$$

Если скалярная кривизна является константой, то следующие условия эквивалентны:

$$SW = 0 \Rightarrow \nabla_Z r(X, Y) = \nabla_Y r(X, Z) \quad (1)$$

Ключевым шагом к решению проблемы классификации однородных (псевдо)римановых

многообразий с нулевым тензором Схоутена–Вейля является последовательное рассмотрение всех возможных типов Сегре оператора Риччи.

Пусть $(M = G/H, g)$ – однородное (псевдо)риманово многообразие размерности m . Обозначим через \mathfrak{g} алгебру Ли группы G , через \mathfrak{h} – подалгебру изотропии, а через $\mathfrak{m} = \mathfrak{g}/\mathfrak{h}$ (необязательно редуктивное) – дополнение к \mathfrak{h} в \mathfrak{g} .

Пара $(\mathfrak{g}, \mathfrak{h})$ однозначно определяет представление изотропии $\phi : \mathfrak{h} \rightarrow gl(\mathfrak{m})$ правилом $\phi_X(Y) = [X, Y]_{\mathfrak{m}}$. Инвариантной (псевдо)римановой метрике на G/H соответствует невырожденная билинейная форма g на \mathfrak{m} такая, что:

$$(\phi_X)^t * g + g * \phi_X = 0, \forall X \in \mathfrak{h} \quad (2)$$

где $(\phi_X)^t$ – транспонированная матрица. Эта форма однозначно определяет связность Леви–Чивита $\nabla : \mathfrak{g} \rightarrow gl(\mathfrak{m})$ правилом

$$\nabla_X(Y_{\mathfrak{m}}) = \frac{1}{2}[X, Y]_{\mathfrak{m}} + v(X, Y)$$

где отображение $v : \mathfrak{g} \times \mathfrak{g} \rightarrow \mathfrak{m}$ определяется формулой

$$2g(v(X, Y), Z_{\mathfrak{m}}) = g(X_{\mathfrak{m}}, [Z, Y]_{\mathfrak{m}}) + g(Y_{\mathfrak{m}}, [Z, X]_{\mathfrak{m}})$$

Тензору кривизны связности ∇ соответствует отображение $R : \mathfrak{m} \times \mathfrak{m} \rightarrow gl(\mathfrak{m})$ такое, что

$$R(X, Y) = [\nabla_Y, \nabla_X] + \nabla_{[X, Y]}$$

Тензор Риччи r определяется формулой

$$r(x) = tr(Z \rightarrow R(X, Z)Y)$$

II. АЛГОРИТМ НА ОСНОВЕ АНАЛИЗА ГРУППЫ ИЗОМЕТРИЙ

Пусть $e_1, e_2, \dots, e_h, u_1, u_2, \dots, u_m$ – базис \mathfrak{g} , где e_i и u_i базисы \mathfrak{h} и \mathfrak{m} соответственно. Положим

$$[u_i, u_j]_{\mathfrak{m}} = c_{ij}^k u_k, [u_i, u_j]_{\mathfrak{h}} = C_{ij}^k e_k,$$

$$[h_i, u_j]_m = \bar{c}_{ij}^k u_k$$

где $c_{ij}^k, C_{ij}^k, \bar{c}_{ij}^k$ – массивы соответствующих размеров.

Первым шагом вычислим представление изотропии ϕ на базисных векторах \mathfrak{h} :

$$(\phi_i)_j^k = (\phi(e_i))_j^k = \bar{c}_{ij}^k$$

и запишем систему уравнений (2).

Далее, с помощью уже известных структурных констант и матрицы метрического тензора, найдем компоненты связности Леви–Чивита ∇ :

$$T_{ij}^k = \frac{1}{2} (c_{ij}^k + g^{sk} c_{sj}^l g_{il} + g^{sk} c_{si}^l g_{jl}),$$

$$\bar{T}_{ij}^k = \frac{1}{2} \bar{c}_{ij}^k - \frac{1}{2} g^{sk} c_{is}^{-l} g_{jl}$$

где $\nabla_{u_i} u_j = T_{ij}^k$, $\nabla_{h_i} u_j = \bar{T}_{ij}^k$ и g^{ij} – матрица, обратная к матрице $\{g_{ij}\}$.

Следующим шагом является вычисление компонент тензора кривизны R и тензора Риччи r :

$$R_{ijks} = (T_{ij}^l T_{jl}^p - T_{jk}^l T_{il}^p + c_{ij}^l T_{lk}^p + C_{ij}^l \bar{T}_{lk}^p) g_{ps},$$

$$r_{ik} = R_{ijks} g^{js}$$

Далее находятся компоненты ковариантной производной тензора Риччи

$$r_{ij,k} = r_{sj} T_{ki}^s + r_{is} T_{kj}^s$$

и выписывается система уравнений (1): $r_{ij,k} = r_{ik,j}$, которая дополняется системой (2) и условием выполнения тождества Якоби. Полученная система решается относительно структурных констант алгебры Ли.

Описание и пример применения данного алгоритма для классификации пространств размерности 4 приводится в статье [2]. Также отметим, что данный алгоритм отличается от техники использования обобщенных базисов Милнора (см. [3]).

III. АЛГОРИТМ НА ОСНОВЕ ОБОБЩЕННЫХ БАЗИСОВ МИЛНОРА

Основным инструментом построения обобщенных базисов Милнора будет теорема, доказанная в [4].

Пусть

$$\bar{\mathfrak{M}} \cong GL_n(\mathbb{R})/O(n)$$

где $\bar{\mathfrak{M}}$ – множество классов эквивалентности $\bar{\mathfrak{M}}$ по отношению изометрии метрик, $\mathfrak{B}\bar{\mathfrak{M}}$ – множество классов эквивалентности $\bar{\mathfrak{M}}$ по отношению изометрии метрик с точностью до умножения на константу.

Пусть \mathfrak{g} – алгебра Ли, $\langle *, * \rangle_0$ – скалярное произведение в \mathfrak{g} , $\{e_1, \dots, e_n\}$ – ортонормированный базис алгебры относительно данного скалярного произведения, \mathfrak{U} – множество представителей $\mathfrak{B}\bar{\mathfrak{M}}$.

Тогда для любого скалярного произведения $\langle *, * \rangle_0$ в алгебре \mathfrak{g} существуют константа $\lambda > 0$, автоморфизм $\phi \in Aut(\mathfrak{g})$ и представитель $g \in \mathfrak{U}$ такие, что базис:

$$\{\phi g e_1, \dots, \phi g e_n\}$$

ортонормирован относительно $\lambda \langle *, * \rangle$ (обобщенный базис Милнора).

Рассмотрим применение данной теоремы на примере решения задачи об изотропности тензора Вейля (см. [5]). Приведем алгоритм решения задачи на основе обобщенных базисов Милнора.

1. Из классификации [6] находим вид ненулевых скобок Ли.
2. С помощью уравнения (1) выписываем матрицу метрического тензора.
3. Находим компоненты тензора Вейля W , используя обобщенный базисы Милнора.
4. Вычисляем квадрат длины тензора Вейля, $\|W\|^2$
5. Решаем систему уравнений $\|W\|^2 = 0, W \neq 0$.

Пример и результаты применения данного алгоритма в случае пространства размерности 3 приведены в работе [5].

Выводы

В данной работе рассмотрены основные методы исследования (псевдо)риманновых однородных пространств с применением систем компьютерной алгебры. Область применения данных методов ограничена условием малой размерности пространств. Для решения задачи в общем случае необходима разработка новых методов и алгоритмов.

СПИСОК ЛИТЕРАТУРЫ

1. Besse, A. Einstein manifolds / A. Besse // Springer-Verlag, Berlin-Heidelberg. – 1987.
2. Клепиков, П. Применение систем компьютерной математики к исследованию однородных (псевдо)риманновых многообразий с тривиальным тензором Схоутена–Вейля // П. Н. Клепиков // Известия АлтГУ. – 2018.
3. Milnor, J. Curvatures of left invariant metrics on Lie groups // J. Milnor // Adv. Math. – 1976. – Vol. 21.
4. Alekseevskiy, D. Groups of conformal transformations of Riemannian spaces //
5. Клепикова, С. Локально однородные псевдоримановы многообразия размерности 4 с изотропным тензором Вейля / С. В. Клепикова, О. П. Хромова // Известия АлтГУ. – 2018. D. V. Alekseevskiy // Math. Sb. – 1985. – V. 89, No 1.
6. Komrakov, B. Einstein–Maxwell equation on four-dimensional homogeneous spaces // B. B. Komrakov // Lobachevskii J. Math. – 2001. – V. 8.

ОДНОРОДНЫЕ ПРОСТРАНСТВА СО СВЯЗНОСТЯМИ НЕНУЛЕВОЙ КРИВИЗНЫ

Можей Н. П.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: mozhey@bsuir.by

В работе проводится описание инвариантных аффинных связностей ненулевой кривизны на трехмерных однородных пространствах. Используется алгебраический подход к проблеме исследования однородных пространств с аффинными связностями, а также соединение различных методов дифференциальной геометрии, теории групп и алгебр Ли и теории однородных пространств. Полученные результаты могут быть использованы в дифференциальной геометрии, теории дифференциальных уравнений, топологии, в теории представлений и теоретической физике.

ВВЕДЕНИЕ

Решение задач классификации однородных пространств, описания инвариантных аффинных связностей на однородных пространствах сегодня важно как для самой теории, так и для приложений, однако эти задачи не были решены даже в малых размерностях. Важный подкласс среди всех однородных пространств формируют изотропно-точные пространства. В частности, этот подкласс содержит все однородные пространства, допускающие инвариантную аффинную связность. Инвариантные связности на однородных пространствах независимо изучались П. К. Рашевским, М. Куритой, Е. Б. Винбергом и Ш. Кобаяши, К. Номидзу [1]. Рассматриваемая тема имеет также многочисленные приложения, например, связности – важный физический объект, к которому приводит геометрическая формулировка теории поля, А. З. Петров [2] дал алгебраическую классификацию полей тяготения, связанную со структурой тензора кривизны пространства. Инвариантные аффинные связности на трехмерных однородных пространствах с неразрешимой группой преобразований изучались в работе [3]. В данной работе изучаются однородные пространства, допускающие связности только ненулевой кривизны.

ОПИСАНИЕ ОДНОРОДНЫХ ПРОСТРАНСТВ И СВЯЗНОСТЕЙ НА НИХ

Пусть M – дифференцируемое многообразие, на котором транзитивно действует группа \bar{G} , $G = \bar{G}_x$ – стабилизатор произвольной точки $x \in M$. Проблема классификации однородных пространств (M, \bar{G}) равносильна классификации (с точностью до эквивалентности) пар групп Ли (\bar{G}, G) , где $G \subset \bar{G}$, так как M может быть отождествлено с многообразием левых смежных классов \bar{G}/G . Изучая однородные пространства, важно рассматривать не саму группу \bar{G} , а ее образ в $Diff(M)$, т. е. достаточно изучать только эффективные действия группы \bar{G} на многообразии M . Пусть $\bar{\mathfrak{g}}$ – алгебра Ли группы Ли \bar{G} , а

\mathfrak{g} – подалгебра, соответствующая подгруппе G . Пара $(\bar{\mathfrak{g}}, \mathfrak{g})$ алгебр Ли называется *эффективной*, если подалгебра \mathfrak{g} не содержит отличных от нуля идеалов $\bar{\mathfrak{g}}$. *Изотропное действие* группы G на касательном пространстве $T_x M$ – это фактордействие присоединенного действия G на $\bar{\mathfrak{g}}$: $s \cdot (x + \mathfrak{g}) = (Ads)(x) + \mathfrak{g}$ для всех $s \in G$, $x \in \bar{\mathfrak{g}}$. При этом алгебра \mathfrak{g} действует на $T_x M = \bar{\mathfrak{g}}/\mathfrak{g}$ следующим образом: $x \cdot (y + \mathfrak{g}) = [x, y] + \mathfrak{g}$ для всех $x \in \mathfrak{g}$, $y \in \bar{\mathfrak{g}}$. Пара $(\bar{\mathfrak{g}}, \mathfrak{g})$ называется *изотропно-точной*, если точно изотропное представление \mathfrak{g} . Это означает, что естественное действие стабилизатора \bar{G}_x , $x \in M$ на $T_x M$ имеет нулевое ядро. Там, где это не будет вызывать разночтения, будем отождествлять подпространство, дополнительное к \mathfrak{g} в $\bar{\mathfrak{g}}$, и факторпространство $\mathfrak{m} = \bar{\mathfrak{g}}/\mathfrak{g}$. *Аффинной связностью* на паре $(\bar{\mathfrak{g}}, \mathfrak{g})$ называется такое отображение $\Lambda : \bar{\mathfrak{g}} \rightarrow \mathfrak{gl}(\mathfrak{m})$, что его ограничение на \mathfrak{g} – изотропное представление подалгебры, а все отображение является \mathfrak{g} -инвариантным. Инвариантные аффинные связности на однородном пространстве (M, \bar{G}) находятся во взаимно однозначном соответствии с аффинными связностями на паре $(\bar{\mathfrak{g}}, \mathfrak{g})$. Необходимое условие существования аффинной связности состоит в том, что представление изотропии для G должно быть точным, если \bar{G} эффективна на \bar{G}/G . Поскольку тензоры кривизны и кручения инвариантны относительно действия группы Ли G , они однозначно определяются тензорами на касательном пространстве к многообразию, причем, эти тензоры инвариантны относительно изотропного действия. Тензоры кручения $T \in InvT_2^1(\mathfrak{m})$ и кривизны $R \in InvT_3^1(\mathfrak{m})$ для всех $x, y \in \bar{\mathfrak{g}}$ имеют соответственно вид $T(x_{\mathfrak{m}}, y_{\mathfrak{m}}) = \Lambda(x)y_{\mathfrak{m}} - \Lambda(y)x_{\mathfrak{m}} - [x, y]_{\mathfrak{m}}$, $R(x_{\mathfrak{m}}, y_{\mathfrak{m}}) = [\Lambda(x), \Lambda(y)] - \Lambda([x, y])$. Переформулируем теорему Вана об алгебре группы голономии инвариантной связности: алгебра Ли группы голономии инвариантной связности $\Lambda : \bar{\mathfrak{g}} \rightarrow \mathfrak{gl}(3, \mathbb{R})$ на паре $(\bar{\mathfrak{g}}, \mathfrak{g})$ – это подалгебра алгебры Ли $\mathfrak{gl}(3, \mathbb{R})$ вида $V + [\Lambda(\bar{\mathfrak{g}}), V] + [\Lambda(\bar{\mathfrak{g}}), [\Lambda(\bar{\mathfrak{g}}), V]] +$

..., где V – подпространство, порожденное множеством $\{\Lambda(x), \Lambda(y) - \Lambda([x, y]) | x, y \in \bar{\mathfrak{g}}\}$.

Будем описывать пару $(\bar{\mathfrak{g}}, \mathfrak{g})$ при помощи таблицы умножения алгебры Ли $\bar{\mathfrak{g}}$. Через $\{e_1, \dots, e_n\}$ обозначим базис $\bar{\mathfrak{g}}$ ($n = \dim \bar{\mathfrak{g}}$). Будем полагать, что подалгебра Ли \mathfrak{g} порождается векторами e_1, \dots, e_{n-3} , а $\{u_1 = e_{n-2}, u_2 = e_{n-1}, u_3 = e_n\}$ – базис \mathfrak{m} . Для нумерации подалгебр используем запись $d.n$, а для нумерации пар – запись $d.n.m$, здесь d – размерность подалгебры, n – номер подалгебры в $\mathfrak{gl}(3, \mathbb{R})$, а m – номер пары $(\bar{\mathfrak{g}}, \mathfrak{g})$. Поскольку ограничение $\Lambda : \bar{\mathfrak{g}} \rightarrow \mathfrak{gl}(\mathfrak{m})$ на \mathfrak{g} – изотропное представление подалгебры, связность определяется своими значениями на \mathfrak{m} . Выпишем ее через образы базисных векторов $\Lambda(u_1), \Lambda(u_2), \Lambda(u_3)$.

Прямыми вычислениями получаем, что, например, все трехмерные однородные пространства, такие, что $\bar{\mathfrak{g}}$ и \mathfrak{g} неразрешимы, допускающие связности только ненулевой кривизны, локально имеют следующий вид:

3.5.2	e_1	e_2	e_3	u_1	u_2	u_3
e_1	0	e_3	$-e_2$	$-u_3$	0	u_1
e_2	$-e_3$	0	e_1	$-u_2$	u_1	0
e_3	e_2	$-e_1$	0	0	$-u_3$	u_2
u_1	u_3	u_2	0	0	e_2	e_1
u_2	0	$-u_1$	u_3	$-e_2$	0	e_3
u_3	$-u_1$	0	$-u_2$	$-e_1$	$-e_3$	0

3.4.3	e_1	e_2	e_3	u_1	u_2	u_3
e_1	0	e_2	$-e_3$	u_1	0	$-u_3$
e_2	$-e_2$	0	e_1	0	u_1	u_2
e_3	e_3	$-e_1$	0	u_2	u_3	0
u_1	$-u_1$	0	$-u_2$	0	$-e_2$	e_1
u_2	0	$-u_1$	$-u_3$	e_2	0	e_3
u_3	u_3	$-u_2$	0	$-e_1$	$-e_3$	0

5.2.3	e_1	e_2	e_3	e_4	e_5	u_1	u_2	u_3
e_1	0	$2e_2 - 2e_3$	e_4	$-e_5$	u_1	$-u_2$	0	0
e_2	$-2e_2$	0	e_1	0	e_4	0	u_1	0
e_3	$2e_3 - e_1$	0	e_5	0	u_2	0	0	0
e_4	$-e_4$	0	$-e_5$	0	0	0	$u_1 + \alpha e_4$	0
e_5	$e_5 - e_4$	0	0	0	0	0	$u_2 + \alpha e_5$	0
u_1	$-u_1$	0	$-u_2$	0	0	0	$\alpha u_1 - e_4$	0
u_2	$u_2 - u_1$	0	0	0	0	0	$\alpha u_2 - e_5$	0
u_3	0	0	$0 - u_1 - \alpha e_4 - u_2 - \alpha e_5 - \alpha u_1 + e_4 - \alpha u_2 + e_5$	0	0	0	0	0

6.1.3.	e_1	e_2	e_3	e_4	e_5	e_6	u_1	u_2	u_3
e_1	0	$2e_2$	$-2e_3$	0	e_5	$-e_6$	u_1	$-u_2$	0
e_2	$-2e_2$	0	e_1	0	0	e_5	0	u_1	0
e_3	$2e_3$	$-e_1$	0	0	e_6	0	u_2	0	0
e_4	0	0	0	0	e_5	e_6	u_1	u_2	0
e_5	$-e_5$	0	$-e_6$	$-e_5$	0	0	0	0	u_1
e_6	e_6	$-e_5$	0	$-e_6$	0	0	0	0	u_2
u_1	$-u_1$	0	$-u_2$	$-u_1$	0	0	0	0	$-e_5$
u_2	u_2	$-u_1$	0	$-u_2$	0	0	0	0	$-e_6$
u_3	0	0	0	0	$-u_1$	$-u_2$	e_5	e_6	0

Аффинная связность в этих случаях имеет вид:

$$3.4.3. \begin{pmatrix} 0 & p_{1,2} & 0 \\ 0 & 0 & p_{1,2} \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} -p_{1,2} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & p_{1,2} \end{pmatrix},$$

$$3.5.2. \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & p_{2,3} \\ 0 & -p_{2,3} & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & -p_{2,3} \\ 0 & 0 & 0 \\ p_{2,3} & 0 & 0 \end{pmatrix},$$

$$6.1.3. \begin{pmatrix} 0 & 0 & p_{1,3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & p_{1,3} \\ 0 & 0 & 0 \end{pmatrix},$$

$$5.2.3. \begin{pmatrix} 0 & 0 & q_{2,3} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & q_{2,3} \\ 0 & 0 & 0 \end{pmatrix},$$

ЗАКЛЮЧЕНИЕ

Проведена локальная классификация трехмерных однородных пространств, допускающих связности только ненулевой кривизны. Описаны все инвариантные аффинные связности на каждом найденном однородном пространстве.

В работе используется алгебраический подход к проблеме исследования однородных пространств с аффинными связностями, а также соединение различных методов дифференциальной геометрии, теории групп и алгебр Ли и теории однородных пространств.

Полученные результаты могут быть использованы не только в различных разделах математики, но и в классической и квантовой механике, квантовой теории поля и других областях теоретической физики, поскольку многие фундаментальные задачи в этих областях связаны с изучением инвариантных объектов на однородных пространствах.

СПИСОК ЛИТЕРАТУРЫ

1. Nomizu, K. Invariant affine connections on homogeneous spaces/ K. Nomizu// Amer. J. Math.– 1954.– Vol. 76.– P. 33–65.
2. Petrov, A. Z. New methods in the general theory of relativity/ A. Z. Petrov.– М., 1966.
3. Mozhey, N. P. Invariant affine connections on three-dimensional homogeneous spaces with non-solvable transformation group/ N. P. Mozhey// Lobachevskii Journal of Mathematics.–2014.– Vol. 35.– P. 218–240.

ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ РЕШЕНИЯ ЗАДАЧ МНОГОМЕРНОЙ ОПТИМИЗАЦИИ

Нестеренков С. Н., Наливко В. Н.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: nsn@bsuir.by, vnalivko@mail.ru

Сформулирована задача многомерной оптимизации и предложено ее решение, базирующееся на генетическом алгоритме. Рассмотрены основные достоинства и недостатки данного подхода.

ВВЕДЕНИЕ

Одной из задач многомерной оптимизации является задача о рюкзаке [1]. Цель многомерной задачи о рюкзаке состоит в том, чтобы увеличить сумму значений элементов, которые должны быть выбраны из некоторого заданного набора, с учетом ограничений по нескольким ресурсам. Эта проблема широко изучалась на протяжении многих десятилетий. В основном задачу можно сформулировать следующим образом [2]:

$$\max f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n p_j x_j. \quad (1)$$

При условии:

$$\sum_{j=1}^n w_{ij} x_j \leq c_i, \quad (2)$$

$$i = 1, \dots, m; \quad x_j \in \{0, 1\}; \quad j = 1, \dots, n;$$

$$p_j > 0; \quad w_{ij} \geq 0; \quad c_i \geq 0.$$

Для корректной постановки опишем условные обозначения:

1. n – количество объектов;
2. m – количество ранцев;
3. w_{ij} – потребление ресурса i для объекта j ;
4. c_i – вместимость i -го ранца;
5. p_j – прибыль;
6. x_j – переменная решения.

Многомерная задача о рюкзаке является частным случаем классической задачи о рюкзаке 0-1 и имеет более одного ограничения. Классическая задача о рюкзаке состоит в том, чтобы выбрать подмножество из бесконечного набора предметов, что повышает линейную функцию выбранных предметов в зависимости от одного ограничения неравенства. Многомерная задача о рюкзаке является особенно сложной проблемой целочисленного программирования, поскольку матрица ограничений, состоящая из w_{ij} , является плотной. С другой стороны, для многомерной задачи о рюкзаке есть реальное решение при $x_j = 0$; $j = 1, \dots, n$, тогда как найти реальное решение может быть так же сложно, как найти оптимальное решение в общем целочисленном программировании [3].

I. ГЕНЕТИЧЕСКИЙ АЛГОРИТМ

Генетические алгоритмы, которые находят применение в биоинформатике, вычислительной науке, экономике, химии, производстве, математике, физике и других областях, являются алгоритмами поиска, основанными на естественном отборе и генетике. Эти алгоритмы принадлежат к большому классу эволюционных алгоритмов, которые генерируют решения для задач оптимизации с использованием методов, основанных на естественной эволюции: наследование, мутация, отбор и кроссовер. Можно сказать, что у самых сильных особей в популяции будет больше шансов передать свои гены следующему поколению.

В генетическом алгоритме популяция возможных решений проблемы оптимизации развивается в сторону лучших решений. Каждое возможное решение имеет набор свойств, которые могут быть видоизменены. Традиционно решения представляются в двоичном виде (0, 1), но возможны и другие кодировки.

Эволюция обычно начинается с популяции случайно сгенерированных индивидов и происходит поколениями. В каждом поколении оценивается пригодность каждого индивида в популяции, более подходящие индивиды стохастически выбираются из текущей популяции, и геном каждого индивида модифицируется для формирования новой популяции. Новая популяция затем используется в следующей итерации алгоритма. Обычно алгоритм завершается, когда было произведено максимальное количество поколений или достигнут удовлетворительный уровень пригодности популяции [4]. Многие сложные проблемы оптимизации могут быть решены или преобразованы с помощью последовательности подзадач, например, с помощью методов решения задач о рюкзаке [5].

Воспроизведение включает следующие способы:

- Чистое воспроизведение – особь копируется непосредственно в следующее поколение;
- Кроссовер – выбраны две особи, их гены в какой-то момент пересекаются, поскольку первая часть новой особи происходит от одного родителя, а последняя часть – от другого;

- Мутация – индивидуум выбран, и один бит изменен.

II. ОБЩЕЕ ОПИСАНИЕ РАБОТЫ АЛГОРИТМА

Генетический алгоритм состоит из следующих шагов [6–11]:

1. Каждое из m -ограничений обрабатывается отдельно, и его оптимальное решение находится методом динамического программирования. Найдены общие частоты появления, которые находятся в векторах решения, затем они отсортированы в порядке убывания, и получена индексная последовательность I ;
2. Первые n -элементов начальной совокупности устанавливаются таким образом, что элемент, относящийся к текущему индексу, берется до тех пор, пока он не превышает вместимость рюкзака, начиная с i -го элемента последовательности индекса I ($1 < m < i$) на каждом шагу;
3. Каждое из m -ограничений обрабатывается отдельно, и вычисляются значения: p_j/w_{ij} , ($1 \leq i \leq m$). Расслабленные решения каждого ограничения найдены, затем получается индексная последовательность J путем сортировки частот ввода решения каждого элемента в порядке убывания;
4. Другие n -элементов начальной совокупности устанавливаются таким образом, что элемент, относящийся к текущему индексу, берется до тех пор, пока он не превышает вместимость рюкзака, начиная с j -го элемента последовательности J ($1 \leq j \leq n$);
5. Коэффициенты целевой функции p_j сортируются в порядке убывания, и получается индексная последовательность K ;
6. Каждый индивид, состоящий из $2n$ -элементов, пересекается со всеми остальными индивидами. Если есть элемент, который можно взять для сгенерированного индивида, элемент, относящийся к текущему индексу, берется до тех пор, пока он не превышает вместимость рюкзака, начиная с первого элемента последовательности индекса K ($1 \leq k \leq n$). Индивид, имеющий максимальное значение целевой функции в популяции, назначается в качестве записи;
7. Предыдущий шаг повторяется до тех пор, пока номер итерации не станет n ;
8. Запись записана, и алгоритм заканчивается.

ЗАКЛЮЧЕНИЕ

Многомерная проблема задач о рюкзаке возникает в различных приложениях, таких как погрузка груза, упаковка в контейнеры, финансовый менеджмент, для решения задачи составления расписания пар преподавателей, причем в данную задачу можно внести еще дополнительные переменные, такие как часы занятости

преподавателей другими видами деятельности, а также часы занятости студентов (в случае посещения ими занятий в других организациях) [12]. Рассмотренный алгоритм дает оптимальные решения для всех случаев. В отличие от техники классического генетического алгоритма, начальная популяция не генерируется случайным образом в этом алгоритме через шаги 1–4. Таким образом, пространство решения сканируется намного эффективнее. Кроме того, вместимость рюкзака влияет на время работы [13–16].

СПИСОК ЛИТЕРАТУРЫ

1. Абакаров А. Ш., Сушков Ю. А. Статистическое исследование одного алгоритма глобальной оптимизации. — Труды ФОРА, 2004.
2. Djannaty F., Doostar. S. A Hybrid Genetic Algorithm for the Multidimensional Knapsack Problem. International Journal Contemporary Mathematical Sciences. 2008. P. 443–456.
3. Lin E. A bibliographical survey on some well-known non-standard knapsack problems, 2004. P. 274–317.
4. Goldberg D. E. Genetic Algorithms in Search, Optimization and Machine Learning // Addison–Wesley.
5. Haul C., Voss S. Using surrogate constraints in genetic algorithms for solving multidimensional knapsack problems / Woodruff D. L. Advances in Computational and Stochastic Optimization, Logic Programming, and Heuristic Search // Kluwer Academic Publishers. 2002 P. 235–251.
6. Gotlieb J. On the effectivity of evolutionary algorithms for multidimensional knapsack problem, Proceedings of the 4th European Conference of Artificial Evolution. // Dunkerque, France. P. 23–27.
7. Gavish B., Pirkul H. Efficient Algorithms for Solving Multiconstraint Zero-One Knapsack Problems to Optimality // Mathematical Programming 31. P. 78–105.
8. Freville A. The multidimensional 0–1 knapsack problem. // European Journal of Operational Research 155. 2004 P. 1–21.
9. Garey M. R., Johnson D. S. Computers and Intractability: A Guide to the Theory of NP-Completeness. P. 338.
10. Kellerer H., Pferschy U., Pisinger D. Knapsack Problems. 2004 P. 546.
11. Khuri S., Back T., Heitkotter J. The Zero/One Multiple Knapsack Problem and Genetic Algorithms // ACM Symposium on Applied Computing. P. 188–193.
12. Нестеренков С. Н. Метод определения персональных весовых коэффициентов преподавателей при распределении их нагрузки / С.Н. Нестеренков // Вести Института современных знаний - 2015. N1 (62). С. 74–80.
13. Vasquez M., Yannick V. Improved results on the 0–1 multidimensional knapsack problem. // European Journal of Operational Research 165. 2005 P. 70–81.
14. Vasquez M., Hao J. K. A hybrid approach for the 0–1 multidimensional knapsack problem. // Proceedings of the Int. Joint Conference on Artificial Intelligence, Seattle, Washington. 2004 P. 328–333.
15. Raidl Gunther R., Gottlieb J. Empirical analysis of locality, heritability and heuristic bias in evolutionary algorithms: A case study for the multidimensional knapsack problem. // Evolutionary Computation Journal 13. 2005 P. 441–475.
16. Raidl Gunther R. An improved genetic algorithm for the multiconstrained 0–1 knapsack problem / Fogel D. et al., eds. // Proceedings of the 5th IEEE International Conference on Evolutionary Computation. P. 207–211.

НЕКЛАССИЧЕСКАЯ ИГРА ЗМЕЙКА НА OPENGL

Панкевич Д. С., Кривоносова Т. М.

Кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: pankevich.daniil@yandex.ru

Рассматривается пример развития классического концепта игры «Змейка» в оригинальную идею, раскрывающая потенциал заложенных в игру основных механик.

I. КЛАССИЧЕСКАЯ ИГРА «ЗМЕЙКА»

Игрок управляет существом, напоминающим змею, которое ползает по плоскости, собирая еду, избегая столкновения с собственным хвостом. Каждый раз, когда змея съедает кусок пищи, она становится длиннее, что постепенно усложняет игру. Игрок управляет направлением движения головы змеи, а хвост змеи движется следом. В некоторых версиях игры на поле присутствуют дополнительные препятствия, разные бонусы. Есть версии игры с несколькими уровнями или с одним уровнем, где необходимо набрать как можно больше очков.

II. РАЗРАБОТКА СВОЕЙ ВЕРСИИ ИГРЫ

Разрабатываемая версия игры будет иметь следующие особенности:

- На игровом поле может находиться несколько единиц еды одновременно;
- От длины Змейки зависит её скорость;
- В игре присутствует несколько уровней;
- У змейки есть несколько жизней на текущий уровень;
- При столкновении с телом у игрока отнимается одна жизнь и та часть тела, которая была «отрезана» от тела, становится стеной;
- При столкновении со стеной или когда у Змейки не останется жизней уровень начнётся заново.

III. ВЫБОР ПЛАТФОРМЫ

Выбор платформы я остановил на программном интерфейсе OpenGL и языке программирования C++. При выборе я отталкивался от того, что этого будет достаточно для воплощения в жизнь своего концепта.

IV. ОСНОВНАЯ СТРУКТУРА

Игровое поле - двумерный массив объектов перечисления(enum). На игровом поле могут располагаться следующие объекты(в скобках указан цвет): стена(серый), еда(желтый, пустота(черный), тело змейки(белый) и голова змейки(зависит от количества жизней). Каждый из этих объектов - именованная константа из перечисления. Каждый кадр по определённому алгоритму массив с объектами отображается на

экране в виде сетки с расположенными на ней квадратами разного размера и цвета, в зависимости от свойства объектов.

V. СТРУКТУРА ЗМЕЙКИ

Змейка - двусвязная очередь объектов, хранящих в себе две целочисленных координаты. Эта очередь каждый игровой тик пересчитывается и вносит изменения в игровое поле. При пересчёте голова перемещается на одну клетку вперёд в направлении текущего движения, а все остальные элементы становятся на место следующих. Координаты последнего элемента змейки на игровом поле превращаются в объект пустоты.

VI. УПРАВЛЕНИЕ

У змейки есть две переменные, хранящие в себе одно из четырёх направлений. Одна переменная хранит текущее направление – то направление, куда змейка движется, а другая переменная хранит пользовательское направление – то направление, которое хочет задать игрок. Если пользовательское направление не является противоположным текущему – то оно становится подтверждённым и змейка меняет направление своего текущего движения. В ином случае змейка не реагирует на изменение пользовательского направления и продолжает движение в сторону текущего.

Захват управления осуществляется путём считывания состояния старшего бита адреса клавиш клавиатуры.

Стрелки на клавиатуре – задать пользовательское направление движения;

Spacebar – поставить игру на паузу.

Escape – выход из игры.

Цифры от 1 до 5 – позволяют переключаться между пройденными уровнями.

VII. ЦИКЛ ПРОГРАММЫ

Каждый игровой тик выполняется набор определённых действий:

1. Обработка нажатий клавиш. На этом этапе происходит обработка состояния каждой из задействованных в игре клавиш, и в зависимости от их текущего и прошлого состояния им присваивается одно из трёх новых:

- нажатие, удержание, или отжатие клавиши.
2. Обновление игровых состояний. На этом этапе происходит пересчёт игрового поля по соответствующим правилам расположенных на нём объектов.
 3. Отрисовка экрана. На этом этапе происходит отрисовка игрового поля в виде сетки с разноцветными, разными по размеру клетками.

VIII. ОПИСАНИЕ РЕАЛИЗАЦИИ МЕХАНИК

У змейки всего 3 жизни на уровень, которые можно восстановить, собирая еду. За каждые 17 единиц еды восстанавливается одна жизнь.

При врезании в сегмент тела у змейки отнимается одна жизнь и начинается поиск по очереди элемента с координатами столкновения. Когда нужный элемент найден - происходит удаление всех элементов от хвоста до того сегмента, с которым столкнулась голова, при этом координаты удаленных сегментов на поле заменяются на объекты стен. Также отбирается столько еды, сколько было удалено сегментов тела. То бишь, у змейки отпадает часть от хвоста до того сегмента, с которым столкнулась голова, а отпавшая часть превращается в стену и остаётся лежать на игровом поле, как препятствие.

Если у змейки не останется жизней, то уровень начинается заново. Индикатор количества жизней - цвет головы змейки. Всего три цвета: синий(3 жизни), фиолетовый(2 жизни) и розовый(1 жизнь).

При столкновении с краем игрового поля голову змейки переносит на противоположный по текущей координате столкновения конец этого поля.

При врезании в стену змейка умирает и, с небольшой задержкой на кат-сцену, уровень начнётся заново. Каждый уровень представляет собой функцию, которая заполняет массив игрового поля объектами перечисления в определённой последовательности. Для хранения текущего уровня используется указатель на функцию заполнения.

Скорость змейки линейно зависит от количества собранной еды. При увеличении скорости змейки вокруг её тела образуется красная обводка, и чем больше скорость - тем она лучше видна. Чтобы перейти на следующий уровень игроку необходимо набрать определённое количество еды, для каждого уровня оно своё.

При врезании змейки в объект еды он удаляется с игрового поля, счётчик количества собранной еды увеличивается на единицу и у змейки добавляется один элемент в конец очереди. Координатам этого элемента присваиваются координаты предыдущего элемента очереди. После того, как объект еды был удалён, составляется карта пустых клеток игрового поля, затем случайным образом из неё выбирается одна клетка,

и по координатам этой клетки на карте на соответствующих координатах игрового поля создаётся объект еды.

IX. ОПИСАНИЕ ДИЗАЙНА УРОВНЕЙ

В игре есть несколько заготовленных уровней, которые ставят перед игроком определённые испытания:

1. Игроку даётся пустое поле, обнесённое по периметру стенами. Тут игрок понимает, можно ли врезаться в стены, в себя, и что от набранной массы меняется скорость змейки;
2. Уровень построен таким образом, что игроку обязательно необходимо привыкнуть и применять новую механику с врезанием в стены, чтобы пройти уровень;
3. Смешанный тип уровня. Игрок может долгое время находится на одной половине уровня, пока на нём есть еда, а затем перебраться на другую половину уровня используя стены;
4. На уровне нет стен и есть очень большое количество еды. Длина змейки, необходимая для прохождения уровня, в несколько раз выше предыдущих уровней, и игрок разовьёт на змейке максимальную скорость, что вызовет у него очередное испытание, даже не имея статичных препятствий на уровне, т.к. само тело змейки и есть главное препятствие;
5. Уровень со случайной генерацией стен у краёв экрана и свободным пространством ближе к центру. Добавляет разнообразия привычным симметричным уровням.

Отталкиваясь от классического концепта можно прийти к чему-то оригинальному, и это довольно частая практика в мире игровой индустрии. Не смотря на простоту исходной идеи, она имеет безграничные возможности по расширению и добавлению новых механик, правил, фишек, игровых режимов и многого другого. Идя от меньшего к большему – мы создаем не только что-то особенное, но и в целом продвигаем вперёд игровую индустрию.

СПИСОК ЛИТЕРАТУРЫ

1. The OpenGL Graphics System: A Specification (Version 4.5 (Core Profile) - August 11, 2014).
2. Шилдт, Герберт С++: базовый курс / Герберт Шилдт – М.: Вильямс, 2008. - 624 с.
3. Лафоре Р. Приложение Е // Лафоре Р. Объектно-ориентировочное программирование на с++. — Санкт-Петербург: Питер, 2004. — С. 836—843
4. Скотт Мейерс. Эффективное использование STL = Effective STL. — Питер, 2002. — С. 224. — (Библиотека программиста). — ISBN 5-94723-382-7.
5. Дэвид Р. Мюссер, Жилмер Дж. Дердж, Атул Сейни. С++ и STL: справочное руководство = STL Tutorial and Reference Guide: C++ Programming with the Standard Template. — 2-е издание. — М.: «Вильямс», 2010. — С. 432. — (серия С++ in Depth). — ISBN 5-89818-027-3.

ТЕХНИЧЕСКАЯ РЕАЛИЗАЦИЯ И ДИДАКТИЧЕСКАЯ ЦЕННОСТЬ РАЗРАБОТАННОЙ ЛОГОПЕДИЧЕСКОЙ ИГРЫ

Россолова К. С., Бычков П. А., Шилин Л. Ю.

ГУО «Ясли-сад №2 агр. Буйничи» Могилёвской области Могилёвского района
Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники
Могилёв, Республика Беларусь; Минск, Республика Беларусь
E-mail: k-klyuchnikova.93@mail.ru, pashaby4ok@mail.ru

Рассматриваются общие технические сведения логопедической игры, логопедические упражнения, реализация переключения уровней.

ВВЕДЕНИЕ

Образовательная логопедическая игра «Котёнок Звуковкин» объединяет в себе разнообразные творческие задания и упражнения по закреплению правильного произношения.

I. ОБЩИЕ ТЕХНИЧЕСКИЕ СВЕДЕНИЯ ОБ ИГРЕ

Цель проекта: разработка образовательной логопедической 2D-игры на платформе Android для дошкольников с речевыми нарушениями.

Для достижения поставленной цели были выделены 4 глобальные задачи:

1. Разработать механику логопедических уровней;
2. Озвучить логопедические уровни;
3. Спроектировать удобный интерфейс;
4. Придать игре образовательную ценность.

Разработка проекта была разбита на создание отдельных модулей:

- модуль инициализации приложения;
- звуковой модуль;
- модуль вывода изображения и звука;
- модуль интерфейса;
- модуль обработки касаний;
- модуль интерактива с пользователем;
- модуль переключения уровней;
- модуль сохранения данных.

В качестве игрового движка был выбран Unity, в качестве среды разработки - Visual Studio 2015. Для создания картинок использовался Adobe Photoshop CS. Для обработки звука лучше всего подошёл Sound Forge Pro 11.0. Игровое приложение создавалось под операционную систему Android. Язык программирования, на котором написаны скрипты – C#.

Звук записан в формате .wav со скоростью потока в 1411 кбит/сек.

II. РЕАЛИЗАЦИЯ ПЕРЕКЛЮЧЕНИЯ УРОВНЕЙ

Для того, чтобы загрузился новый уровень, используется следующий скрипт:

```
public void buttonLevel1Click()
{
    Application.LoadLevel("level1");
}
```

В кавычках указано «название» уровня, который будет загружен. В данном случае – это уровень 1. Название техническое и для пользователя оно не будет иметь никакого значения, так как ему не будет его видно.

Далее «навешивается» написанный ранее скрипт на кнопку. В Unity у кнопок есть функция On Click (), куда мы можем «навесить» скрипт и выбрать функцию скрипта, которая будет срабатывать при нажатии на кнопку.

Однако данная функция будет срабатывать лишь после того, как пользователь нажмёт и отпустит кнопку. Поэтому для ускорения работы используется Event Trigger, в котором можно задать загрузку уровня сразу после нажатия кнопки.

Кроме скрипта так же необходимо добавить уровни в Build Settings. Это можно сделать простым перетаскиванием или нажатием на кнопку Add Open Scenes для добавления открытых, на текущий момент, сцен.

III. ДИДАКТИЧЕСКАЯ ЦЕННОСТЬ

Прототип стартового меню смотрите на рисунке 1.

В этом меню выделяется окошко «Артикуляционная гимнастика», так как любое логопедическое занятие начинается с Артикуляционной гимнастики, то есть, с разминки языка. Также в меню пользователь может выбрать окошко в соответствии с речевым нарушением: свистящие звуки, шипящие и сонорные. Такая последовательность не случайна и обусловлена соответствием речевому онтогенезу (появлению звуков в речи).

Рассмотрим упражнения на примере обрабатываемого шипящего звука [Ч].

- Артикуляционная гимнастика. Данный блок представлен в виде упражнений в стихотворной форме, в которых можно найти указания, что и как выполнять. Кроме того, на каждом уровне данного блока можно увидеть маленькие парные картинки, на которых показаны положение губ и языка в этом упражнении. Каждое

упражнение озвучено, для его выполнения ребенку даётся время.

- Автоматизация звука [Ч] изолированно и в слогах. Данный блок представлен упражнениями «Паровозик» и «Волшебная дорожка», что позволяет закрепить навык правильного произношения звука [Ч] изолированно, а также в обратных и прямых слогах.
- Автоматизация звука [Ч] в словах. Данный блок представлен упражнениями «Назови картинку», «Подбери слова на слоги», «Поиск», «Закончи предложения».
- Слоговой анализ слов. Данный блок представлен упражнением «Веселый поезд». Ребенку предлагается помочь пассажирам отправиться в путешествие на весёлом поезде, предварительно назвав каждого пассажира, выделяя звук [Ч], затем прохлопать в ладоши количество слогов в каждом слове, чтобы узнать, кто в каком вагоне.
- Развитие логического мышления. Данный блок представлен игрой «Четвертый лишний». Ребенку среди четырех картинок, в названиях которых есть звук [Ч], необходимо найти лишнюю. При выборе правильного ответа, ребенок вместе со Звуковкин переходит к выполнению следующего задания, в противном случае Звуковкин предлагает попробовать еще раз.
- Заключительная часть. В данном блоке представлен прием рефлексии. Котенок Звуковкин обращается к ребенку с вопросом, в котором уточняется, понравилась ли ему игра, и для ответа дошкольнику необходимо выбрать клубок ниток соответствующего цвета.

Вывод

Конечно же, данное приложение не сможет заменить квалифицированного учителя-логопеда, однако «карманным логопедическим помощником» эта игра вполне сможет стать.

СПИСОК ЛИТЕРАТУРЫ

1. Ключникова, К. С. Образовательная игра «Котёнок Звуковкин» / К. С. Ключникова, П. А. Бычков // Информационные технологии и системы 2018 (ИТС 2018) = Information Technologies and Systems 2018 (ITS 2018) : материалы международной научной конференции, Минск, 25 октября 2018 г. / Белорусский государственный университет информатики и радиоэлектроники ; редкол. : Л. Ю. Шилин [и др.]. – Минск, 2018. – С. 174 - 175. ;
2. Ключникова, К. С. Разработка и монетизация логопедического приложения / К. С. Ключникова, П. А. Бычков // 55-я юбилейная научная конференция аспирантов, магистрантов и студентов учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»: материалы конференции по направлению 2: Информационные технологии и управление, Минск, 22–26 апреля 2019 г. / редкол.: Л. Ю. Шилин [и др.]. – Минск: БГУИР, 2019. – С. 89. ;
3. Комарова, Л.А. Автоматизация звуков Ч, Щ в игровых упражнениях. Альбом дошкольника / Л.А. Комарова. – М.: Издательство ГНОМ и Д, 2009. – 32 с.;
4. Нищева, Н.В. Веселая артикуляционная гимнастика. Наглядно-дидактическое пособие / Н.В. Нищева. – СПб.: ООО «ИЗДАТЕЛЬСТВО «ДЕТСТВО-ПРЕСС», 2009;
5. Волкова, Л.С. Логопедия: учебник для студентов дефектол. фак. вузов / Л.С. Волкова, С.Н. Шаховская. - М.: Гуманит. изд. центр ВЛАДОС, 1998. – 680 с.;
6. Филичева, Т.Б., Соболева А.Р. Развитие речи дошкольника: Методическое пособие с иллюстрациями / Т.Б. Филичева, А.Р. Соболева. - Екатеринбург: Изд-во «АРГО», 1996. - 80 с.



Рис. 1 – Прототип стартового меню

БИБЛИОТЕКА ДЛЯ ПАРАЛЛЕЛЬНОГО ОБУЧЕНИЯ НЕЙРОННЫХ СЕТЕЙ В МУЛЬТИПРОЦЕССОРНЫХ СИСТЕМАХ С ПОМОЩЬЮ МЕТОДА РОЯ ЧАСТИЦ

Рудикова Л. В., Колосов А. А.

Кафедра современных технологий программирования, Гродненский государственный университет имени Янки Купалы

Гродно, Республика Беларусь

E-mail: rudikowa@gmail.com, arseniy.kolosov@gmail.com

В работе представлен подход к построению и обучению слоистых нейронных сетей в мультипроцессорных системах с мощью метода роя частиц. Предполагается, что представленный алгоритм значительно ускорит время обучения небольших нейронных сетей.

ВВЕДЕНИЕ

В настоящее время тема нейронных сетей является актуальной в связи с увеличением объёмов данных, появлением более мощных вычислительных систем и спросом на нейронные сети в науке и рынки информационных технологий. Но прежде чем применить нейронную сеть на практике необходимо ее обучить, что является крайне требовательным к вычислительным мощностям процессом. Большие нейронные сети могут состоять из десятков миллионов синапсов, представляемых, как правило, в виде матриц, а обучение с помощью самого распространенного алгоритма – обратного распространения ошибки требует перемножения этих матриц. Перемножение больших матриц может в разы ускориться с помощью графических процессоров, но скорости обучения более малых нейронных сетей графические процессоры могут только навредить. Так же не редко для обучения нейронных сетей применяются вычислительные устройства, включающие в себя несколько процессоров и видеокарт, использование которых одновременно накладывает множество ограничений на алгоритм обучения. Таким образом, для эффективного обучения нейронных сетей на мультипроцессорных системах необходимо создать новый алгоритм обучения.

I. ПРОЕКТИРОВАНИЕ БИБЛИОТЕКИ. ВЫБОР ПРОГРАММНЫХ СРЕДСТВ И ОСОБЕННОСТИ РЕАЛИЗАЦИИ

C++ - язык программирования, являющийся компилируемым и имеющий статическую типизацию. Основной причиной выбора языка программирования в пользу C++ является возможность управлением каждого аспекта кода, экономить память и вычислительные ресурсы, поддержка OpenCL – фреймворка для написания подпрограмм, использующих параллельные вычисления на CPU и GPU, позволяя программировать на мультипроцессорных системах [1].

Многослойные нейронные сети являются самой распространенной архитектурой нейрон-

ных сетей, обучение и применений которой требующее, как правило, умножения матриц.

Стадия обучения нейронной сети может проводиться на всех чипах CPU и GPU внутри одного устройства. Таким образом, задача должна разбиться на подзадачи, решаемые каждым процессором по отдельности с учетом дорогостоящего обмена информацией между графическим процессором и центральным процессором (см. рис. 1), а также отсутствие обмена данными между чипами. Алгоритм обучения должен исключать возможность обмена данными между процессорами, а также минимизировать поток обмена данных между ЦП и остальными процессорами. Более того, задача должна обладать свойством массового параллелизма, необходимое для эффективных вычислений на видеокарте.

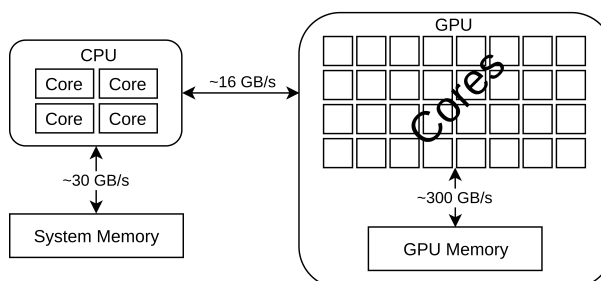


Рис. 1 – Схема скорости обмена данными между чипами

Метод роя частиц – метод численной оптимизации, который можно адаптировать под обучение нейронной сети [2]. Формально, алгоритм представляет из себя агентную систему, где каждый агент (частица) является отдельным решением исходной задачи. Частицы перемещаются в пространстве решений, передвигаясь по направлению к лучшему решению среди всех агентов и собственных лучших результатов с определенной долей вероятности. Более подробное описание алгоритма:

Для работы метода роя частиц необходимо задать некую целевую функцию – функция многих переменных $f : \mathbb{R}^n \rightarrow \mathbb{R}$, экстремум которой необходимо найти. Необходимо так же задать пространство решений - множество $\mathbb{S} \subseteq \mathbb{R}$.

Требуется задать K частиц, i -ая из которых представлена координатой из множества решений $x_i \in \mathbb{R}$ и начальной скоростью $v_i \in \mathbb{R}$. Пусть $loc_i \in \mathbb{R}$ – лучшее состояние, принимаемое когда-либо частицей i , а $glob \in \mathbb{R}$ – лучшее состояние, принимаемое всеми частицами. Общий вид метода роя частиц:

- Для каждой частицы $i \in \overline{1, K}$
 - Сгенерировать случайно равновероятно начальное состояние $x_i \in \mathbb{S}$.
 - Присвоить это значение лучшему локальному состоянию $loc_i = x_i$.
 - Сгенерировать случайным равновероятным образом начальное состояние $v_i \in \mathbb{S}$.
 - Если $f(loc_i) < f(glob)$, то обновить наилучшее глобальное состояние $glob = loc_i$.
- До тех пор, пока не выполнено нужное число итераций или значение $f(glob)$ не будет меньше некоторого заданного значения:
 - Для каждой частицы $x_i \in \mathbb{S}$.
 - Сгенерировать случайные векторы $r_{loc}, r_{glob} \in [0..1]^n$.
 - Обновить скорость частицы $v_i = v_i + k_{loc}r_{loc}(loc_i - x_i) + k_{glob}r_{glob}(glob - x_i)$.
 - Обновить координату частицы $x_i = v_i + x_i$.
 - Если $f(x_i) < f(loc_i)$, то $loc_i = x_i$ и
 - Если $f(loc_i) < f(glob)$, то $glob = loc_i$.
- $glob$ - лучшее найденное решение.

Коэффициенты $s, k_{loc}, k_{glob} \in [0..1]$ – константы, влияющие на замедление, значимость лучшего локального состояния и лучшего глобального соответственно.

Алгоритм обучения:

- Инициализировать все частицы, выборку и задачи.
- До тех пор, пока не выполнено нужное число итераций или нейронная сеть не обучится до приемлемого уровня:
 - Для каждой задачи посчитать значения частиц на каждом примере из выборки (параллельно для каждой задачи).
 - Для каждой задачи посчитать ошибку, полученную на каждом примере (параллельно для каждой задачи).
 - Для каждой задачи высчитать среднюю ошибку для каждой особи (параллельно).
 - Для каждой задачи найти лучшую частицу (параллельно).
 - Обновить лучшие локальные состояния в каждой частице для каждой задачи (параллельно).

- Загрузить новое лучшее глобальное решение в каждую задачу (после инициализации - это единственное место обмена осязательными объемами данных, хоть и оно – минимально).
- Для каждой частицы каждой задачи обновить состояния (сделать смещение на вектор скорости, обновив вектор скорости).
- Лучшее глобальное решение – результат обучения.

Из-за особенностей фреймворка OpenCL накладывается ограничение на многомерные массивы данных, что усложняет разработку библиотеки (см. рис. 2).

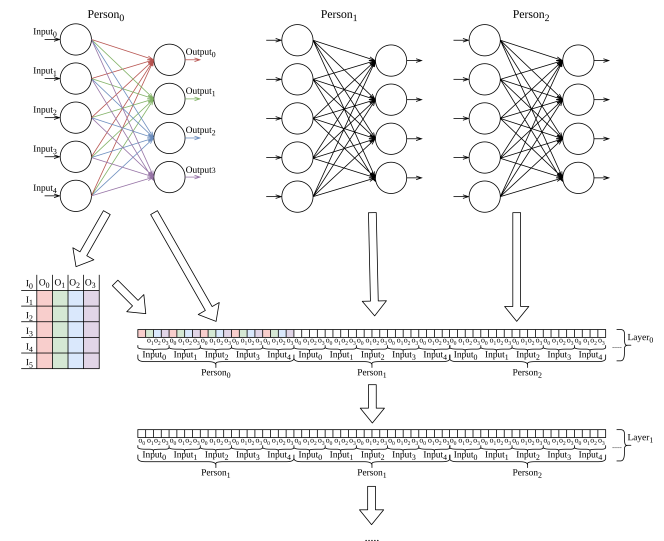


Рис. 2 – Схема хранения нейронных сетей в памяти чипов

II. ЗАКЛЮЧЕНИЕ

Таким образом, предполагаемый подход к обучению нейронных сетей может быть рассмотрен к применению. Предполагаемый алгоритм обучения может значительно ускорить процесс обучения небольших нейронных сетей с минимальными затратами к памяти, максимизируя преимущества вычислительного устройства с одними или несколькими процессорами и видеокартами, нивелируя слабые места таких вычислительных устройств, при этом не требуя наличия определенных операционных систем или специфичных аппаратных средств конкретных производителей.

СПИСОК ЛИТЕРАТУРЫ

1. Документация к OpenCL – [Электронный ресурс]. – Режим доступа: <https://www.khronos.org/op>.
2. Clerc M. (2012). Standard Particle Swarm Optimization.

МОДЕЛИРОВАНИЕ ПЕРЕХОДНЫХ ПРОЦЕССОВ В ЭЛЕКТРИЧЕСКИХ ЦЕПЯХ МЕТОДОМ ПЕРЕМЕННЫХ СОСТОЯНИЯ ПОСРЕДСТВОМ PYTHON

Сманцер Д. С.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: smantserdima@gmail.com

Рассмотрена возможность моделирования переходных процессов в электрических цепях на основе метода переменных состояния с использованием языка программирования Python. Предложенная методика учитывает все особенности действующих в цепи электрических величин и в то же время является наиболее просто программно реализуемой.

В настоящее время достигнуты значительные успехи в математическом моделировании электрических цепей. Математическое моделирование является наиболее удобным аппаратом для исследования физических процессов, так как значительно сокращает время и расходы на проведение экспериментальных исследований, получая при этом достаточно полный объем информации.

При математическом моделировании переходных процессов в электротехнике в последнее время ведутся исследования в двух направлениях: моделирование переходных процессов для решения конкретных исследовательских или прикладных задач и разработка или усовершенствование методов математического моделирования для упрощения вычислений.

Однако известные и применяемые методики расчета переходных процессов в электрических цепях решают узкие прикладные задачи и их применение невозможно для математического моделирования переходных процессов в сложных электрических цепях при действии периодических несинусоидальных токов. Широко применяемые методы анализа и расчета переходных процессов в электрических цепях — это классический метод, операторный метод, метод расчета с помощью интеграла Дюамеля, спектральный метод, метод переменных состояния [1].

Классический и операторный применяются для решения задач любой сложности, но классический более физически прозрачен, чем операторный, в то время как операторный значительно упрощает расчеты. Применение обоих методов зависит от квалификации пользователя и оба метода тяжело поддаются использованию при высоких порядках уравнений. Метод на основе интеграла Дюамеля применяется при сложной функции напряжения во времени, но метод громоздок при вычислении для уравнений высоких степеней. В этом случае целесообразно использовать спектральный метод, но он имеет ряд ограничений в виде требований к точности анализа и моделирования переходных процессов.

При построении математических моделей переходных процессов электрических цепей в основном применяются методы численного анализа. Их разделить на две группы: одношаговые и многошаговые. Из одношаговых наибольшее распространение получили такие методы, как метод Эйлера, метод Эйлера-Коши, метод Рунге-Кутты, метод Рунге-Кутты-Гила, из многошаговых — метод Адамса-Штермера, метод Милна, метод Хэмминга, метод Релстона. Необходимость и возможность применения любого из методов численного интегрирования определяется особенностями решаемой задачи и требуемой точностью вычислений в соответствии с поставленной задачей.

Кроме перечисленных основных методов существует также ряд современных разработок методов для анализа и моделирования переходных процессов, которые являются вариациями существующих методов в сочетании с численными методами расчета. Практически все применяемые в настоящее время методы нуждаются в адаптации и усовершенствовании в целях упрощения вычисления для математического моделирования переходных процессов в электрических цепях.

Рассматривая вышеуказанные методы с точки зрения реализации с помощью программных средств и точности численного расчета переходных процессов для сложных электрических цепей наиболее удачным в применении является метод переменных состояния, так как он наиболее поддается алгоритмизации. Уравнениями состояния электрической цепи называют любую систему уравнений, которая описывает режим этой цепи. Например, система уравнений по законам Кирхгофа является уравнениями состояния электрической цепи, для которой она составлена.

Метод переменных состояния (или пространства состояний) представляет собой упорядоченный способ нахождения состояния системы в функции времени, использующий матричный метод решения системы дифференциальных

уравнений первого порядка, записанных в форме Коши (нормальной формы). Метод переменных состояния - один из методов анализа именно переходных процессов. Применительно к электрическим цепям под переменными состояниями понимают обычно величины, определяющие состояние цепи, т.е. токи через индуктивности и напряжения на ёмкостях (независимые начальные условия). Значения этих величин предполагаются известными к началу процесса, т.е. они находятся из схемы электрической цепи в докоммутационном установившемся режиме. Уравнения состояния формы Коши для схемы электрической цепи могут быть получены из системы уравнений Кирхгофа путем их преобразования.

Для электрических цепей можно составить матричные уравнения вида:

$$[X'(t)] = [A] \cdot [X(t)] + [B] \cdot [F(t)]$$

$$[Y(t)] = [C] \cdot [X(t)] + [D] \cdot [F(t)]$$

где

$[X'(t)]$ – матрица-столбец производных от токов в индуктивностях и напряжений в ёмкостях (n – элементов);

$[A]$ – квадратная матрица коэффициентов при переменных состояния (n строк, n столбцов);

$[X(t)]$ – матрица-столбец переменных состояния (n элементов);

$[F(t)]$ – матрица-столбец (независимых) источников ЭДС и тока (m элементов);

$[B]$ – прямоугольная матрица связи, состоящая из коэффициентов перед источниками ЭДС и тока (n строк, m столбцов);

$[Y(t)]$ – матрица-столбец выходных величин (k элементов);

$[C]$ – прямоугольная матрица связи выходных величин с переменными состояния (k строк, n столбцов);

$[D]$ – прямоугольная матрица связи выходных величин с источниками (k строк, m столбцов) [2].

На основании принципа наложения решение имеет вид:

$$[X(t)] = e^{[A]t} [X(0)] + \int_0^t e^{[A](t-\tau)} [B] \cdot [F(\tau)] d\tau$$

где $[X(0)]$ – матрица начальных значений.

Первое слагаемое в формуле решения описывает свободные процессы в системе, второе – принуждённые при нулевом исходном состоянии.

Решение выходных величин имеет вид: $[Y(t)] = [C] \cdot e^{[A]t} [X(0)] + \int_0^t [C] e^{[A](t-\tau)} [B] \cdot [F(\tau)] d\tau + [D] \cdot [F(t)]$

Последовательность расчета переходного процесса методом переменных состояния:

1. Выполняется расчет схемы в установившемся режиме до коммутации, определяются независимые начальные условия $i_L(0)$ и $u_C(0)$.

2. Выделяются в электрической цепи после коммутации индуктивные и ёмкостные элементы.

3. Заменяются ёмкостные элементы источниками напряжения, индуктивные элементы заменяются источниками тока.

4. Для полученной резистивной схемы составляется система дифференциальных уравнений по законам Кирхгофа (можно использовать любой другой удобный метод анализа электрических цепей – метод наложения либо метод контурных токов).

5. Методом исключения переменных система уравнений Кирхгофа преобразуется в систему уравнений Коши, составляются матрицы коэффициентов.

6. Выбирается метод интегрирования, применимый для программной реализации.

7. Выполняется решение.

8. Выходная функция получается в виде графической диаграммы $x=f(t)$ или в виде таблицы координат функций для заданных моментов времени.

Для программной реализации моделирования переходных процессов можно предложить язык программирования Python, т.к. он упрощает анализ переходных процессов в электрических цепях, делает его наглядным. Python имеет развитые библиотеки, прост и гибок [3]. Язык Python можно рекомендовать любым пользователям, которые используют вычислительную технику и программирование в своей работе.

Учитывая хорошо развитые в библиотеке SciPy численные методы, можно выполнять моделирование переходных процессов при коммутации в сложных электрических цепях средствами этой библиотеки. Численные решения дифференциальных уравнений средствами Python значительно упрощают моделирование переходных процессов в электрических цепях, делают его наглядным и позволяют сосредоточиться на результатах, а не анализе методов решения уравнений.

Таким образом, моделирование переходных процессов методом переменных состояния учитывает все особенности действующих в цепи электрических величин и в тоже время является наиболее просто программно реализуемой, а для программной реализации наиболее эффективным является применение средств программной среды Python.

СПИСОК ЛИТЕРАТУРЫ

1. Галанин М.П., Савенков Е.Б. Методы численного анализа математических моделей / П.М.Галанин, Е.Б. Савенков - М.: МГТУ им. Н.Э. Баумана – 2010. – 591с.
2. Бессонов Л.А. Теоретические основы электротехники: Электрические цепи. Учеб. для студентов электротехнических, энергетических и приборостроительных специальностей вузов / Л.А.Бессонов – 7-е изд., перераб. и доп. –М.: Высш. шк., 1978.
3. Сузи Р.А. Python / Р.А.Сузи – СПб: БВХ-Петербург – 2002. – 768с.

ПРИМЕРЫ ИНТЕЛЛЕКТУАЛЬНОГО АНАЛИЗА И ВИЗУАЛИЗАЦИИ ГЕОДАННЫХ

Таранчук В. Б.

Кафедра компьютерных технологий и систем, Факультет прикладной математики и информатики,
Белорусский государственный университет
Минск, Республика Беларусь
E-mail: taranchuk@bsu.by

Обсуждаются примеры использования программного комплекса составителя цифровых геологических моделей. Описаны возможности интерактивной графической визуализации геоданных, подготовки представительных наборов данных для возможных интерпретаций экспертов. Приведены алгоритмы получения оценок, которые позволяют подтвердить обоснованность интерпретаций, анализировать разными методами интеллектуальной обработки данные компьютерных геологических моделей.

ВВЕДЕНИЕ

Цифровые геологические, геоэкологические модели в настоящее время являются обязательной составляющей экспертизы во многих сферах деятельности. Основными этапами информационного обеспечения геологических моделей являются загрузка из различных источников и предварительная обработка данных, корреляция, накопление и систематизация информации по характеристикам среды, интерактивный анализ данных, визуализация с помощью графики, картографирование.

Используемые в геологических и геоэкологических моделях данные являются представительной частью геоданных [1], технологическая особенность которых состоит в том, что их не получают на основе непосредственных измерений, а они формируются в результате постобработки измеренной информации. Системная особенность заключается в том, что после формирования геоданные представляют собой интегрированную в единый комплекс совокупность параметров и описаний разных типов и структур, в которых учитываются пространственные отношения, временной и тематический факторы. Информационная особенность обусловлена тем, что геоданные представляют собой новый информационный ресурс, при этом данные группируют по трём характеристикам: месту, времени, теме. Также особенностью является реализуемое автоматически взаимовлияние графических и атрибутивных данных, что обеспечивает основу для пространственного визуального анализа и управления.

Объемы геоданных растут с очень большой скоростью. Соответственно, естественным является применение технологий «больших данных» (конкретика для геоданных в [2]), автоматизированного интеллектуального анализа данных (ИАД). В [3] акцентируется одна из главных целей ИАД – обнаружение в «сырых» (первичных) массивах данных ранее неизвестных, нетривиальных, практически полезных и понятных интерпретации знаний. В формулировке автора [3]

«интеллектуальный анализ данных не исключает человеческое участие в обработке и анализе, но значительно упрощает процесс поиска необходимых данных из сырых данных, делая его доступным для широкого круга аналитиков, не являющихся специалистами в статистике, математике или программировании.»

Различных действующих и применяемых программных средств интеллектуального анализа данных много, например, в [3] выделены 7 классов систем ИАД, перспективы развития можно проследить по [4]. Разнообразие предлагаемых методик и программных средств обуславливает необходимость оценки качества геоданных, определения их основных характеристик. Критерии определения свойств геоданных обсуждаются в [5]; отмечено, что разнообразие геоданных порождает проблему их согласования, эффективной эксплуатации баз данных географических информационных систем и связанных с ними внешних баз данных; значение оценки качества существенно возрастает при переходе от информационных к интеллектуальным технологиям. Ряд вопросов анализа, оценок качества имеющихся и формируемых пространственных данных можно решать с использованием компьютерного комплекса «Генератор геологической модели залежи (ГГМЗ)» [6–15].

ОСНОВНЫЕ ПОЛОЖЕНИЯ

Интегрированный программный комплекс ГГМЗ, разработанный путем интеграции возможностей системы компьютерной алгебры и геоинформационной системы, алгоритмов и «умных» методов адаптации моделей в процессе их эксплуатации, предназначен и позволяет выполнять тестирование и оценки точности геологических моделей, обоснование интерпретаций исходных геоданных. В комплексе ГГМЗ реализованы методические решения и соответствующие программные инструменты, которые позволяют имитировать замеры, подтвердить обоснованность интерпретаций, рассчитать числовые значения погрешностей получаемых разными методами результатов интеллектуальной предоб-

работки данных, включаемых и используемых в компьютерных геологических моделях. Программные компоненты, предложенные и реализованные методические и технические решения, которые включены ГГМЗ, описаны в [6–8]. Основная идея и цель разработки комплекса ГГМЗ состоит в предоставлении эксперту инструментов выбора и обоснования метода обработки пространственных данных путем сопоставления эталонного цифрового поля и восстановленного по «наблюдениям». Эталонное цифровое поле рассматривается, как явно задаваемая аналитическая функция, определяемая в области с двумя независимыми переменными. Примеры подготовки разных эталонных цифровых полей и представительных моделей, варианты графической визуализации результатов даны в [6–10]. Отдельное внимание уделяется алгоритмам и программным инструментам получения результатов сопоставления, оценок точности, способов визуализации погрешностей при разных вариантах формирования цифровых описаний методами аппроксимации по рассеянному множеству данных замеров [10–13]. Несколько разных алгоритмов получения представительных данных, имитации наблюдений путем «искажения» эталонной модели добавлением «шумов», используя разные генераторы случайных чисел, изложены в [11–16]. В [14–16] приведены стандартные, но достаточно эффективные методы анализа данных с помощью искусственных нейронных сетей. Примерами иллюстрируются разные возможности рассматриваемых алгоритмов и программных средств при обработке данных, предназначенных для использования в специфических моделях микробиологии (например, [17]), промышленных объектов типа [18].

ЗАКЛЮЧЕНИЕ

В докладе будут конспективно изложены и проиллюстрированы представительными примерами основные из перечисленных выше методов анализа исходных данных для компьютерных геологических моделей. Основное внимание будет уделено сопоставлению результатов, получаемых с использованием инструментов классической статистической обработки данных и аппарата искусственных нейронных сетей.

СПИСОК ЛИТЕРАТУРЫ

- Савиных, В. П. Геоданные как системный информационный ресурс / В. П. Савиных, В. Я. Цветков // Вестник Российской академии наук. –2014. –Т. 84, № 9. –С. 826–829.
- Tsvetkov, V. Ya. Big Data as Information Barrier / V. Ya. Tsvetkov, A. A. Lobanov // European Researcher. –2014. V. 78, № 7-1. –P. 1237–1242.
- Шайтура, С. В. Интеллектуальный анализ данных геоданных / С. В. Шайтура // Перспективы науки и образования. –2015. № 6 (18). –С. 24–30.
- Golenkov, V. V. Ontology-based design of intelligent systems / V. V. Golenkov // Open semantic technologies for intelligent systems (OSTIS-2017) : materials of VII Intern. sc.-tech. conf., Minsk, 16–18 Febr. 2017 / Belarus. State Univ. of Informatics a. Radioelectronics. –Minsk, –2017. –P. 37–56.
- Дышленко, С. Г. Анализ и разработка характеристик качества геоданных / С. Г. Дышленко // Перспективы науки и образования. –2016. № 2 (20). –С. 23–27.
- Таранчук, В. Б. Программный комплекс адаптации геологических моделей. Концепция, решения, примеры реализации / В. Б. Таранчук // Проблемы физики, математики и техники. –2017. –№ 3 (32). –С. 81–90.
- Таранчук, В. Б. Интегрированный программный комплекс тестировщика геологических моделей / В. Б. Таранчук // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. –2017. –№ 16 (265). –С. 148–159.
- Taranchuk, V. B. The integrated computer complex of an estimation and adapting of digital geological models / V. B. Taranchuk // Studia i Materiały. Volume 14. –2017. –№ 2. –С. 73–86.
- Taranchuk, V. The integrated program complex of the composer of geological models. The concept, solutions / V. Taranchuk // Computer Algebra Systems in Teaching and Research. –2017. –Volume VI. –P. 186–203.
- Таранчук, В. Б. Методы и инструментарий оценки точности компьютерных геологических моделей / В. Б. Таранчук // Вестник БрГТУ, № 2, –2018. –С. 64–70.
- Таранчук, В. Б. Средства и примеры интеллектуальной обработки данных для геологических моделей / В. Б. Таранчук // Проблемы физики, математики и техники. –2019. –№ 3 (40). –С. 117–122.
- Таранчук, В. Б. Методы и примеры интеллектуальной обработки данных для геологических моделей / В. Б. Таранчук // Научные ведомости Белгородского государственного университета. Серия: Экономика. Информатика. –2019. –№ 3 (46). –С. 511–522. DOI 10.18413/2411-3808-2019-46-3-511-522.
- Taranchuk, V. Methods and Tools of Intelligent Data Processing for Geological Models / V. Taranchuk // Computer Algebra Systems in Teaching and Research. –2019. –Volume VIII. –P. 147–158.
- Taranchuk, V. New computer technologies, analysis and interpretation of geodata / V. Taranchuk // MATEC Web of Conferences IPICSE-2018. –V. 251, 04059. –P. 1–8.
- Taranchuk, V. B. Examples of the use of artificial neural networks in the analysis of geodata / V. B. Taranchuk // Open Semantic Technologies for Intelligent Systems : Research Papers Collection. –2019. –Issue 3. –P. 225–230.
- Таранчук, В. Б. Интеллектуальные вычисления, анализ, визуализация больших данных / В. Б. Таранчук // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня : сб. материалов V Междунар. науч.-практ. конф. В 2 ч. Ч. 1. –Минск : БГУИР, –2019. –С. 337–346.
- Chichurin, A. Computer simulation of two chemostat models for one nutrient resource / A. Chichurin, H. Shvychkina // Mathematical Biosciences, 278, 2016, –P. 30–36.
- Orlov, V. Mathematical modeling of emergency situations at objects of production and gas transportation / V. Orlov, E. Detina, O. Kovalchuk // MATEC Web of Conferences IPICSE-2018. –V. 251, 04059.

АЛГОРИТМЫ МОДЕЛИРОВАНИЯ ДИНАМИКИ ТВЕРДОГО ТЕЛА

Турец Е. А., Кукин Д. П.

Факультет информационных технологий и управления, кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: egorturets@gmail.com, kukin@bsuir.by

Расчетами внешней баллистики с учетом сопротивления воздуха ученые занимаются уже около 100 лет. Рынок в развитии внешней баллистики в начале 20-го века был обусловлен одновременным прогрессом ствольного оружия и вычислительной математики, аэродинамики и физики обтекания тел. Вместе с тем требования предъявляемые системам расчета постоянно увеличиваются. Алгоритмы моделирования динамики твердых тел используются при научных исследованиях, военных и гражданских разработках, физических движениях и других отраслях.

ВВЕДЕНИЕ

Внешняя баллистика – одна из сфер применения алгоритмов моделирования динамики твердого тела. Она занимается решением четырех основных задач.

1. Основная или прямая задача состоит в расчете траекторий движения снарядов по заранее известным данным. Для ее решения необходимо, прежде всего, правильно определить, какие силы действуют на снаряд в полете, и знать, какова будет их величина в каждый момент времени. Затем следует составить дифференциальные уравнения движения снаряда с учетом всех действующих сил. В результате решения этих уравнений получаются все характеристики движения: скорость, ускорение, время полета и координаты центра масс, по которым может быть построена траектория;
2. Обратная задача состоит в определении проектных баллистических характеристик движения по заданным тактико-техническим данным ракетной или артиллерийской системы. Вторая задача непосредственно связана с баллистическим проектированием системы, важным этапом которого является отыскание оптимальных режимов движения и траекторий полета;
3. Расчет стабилизации снарядов различного назначения и определение условий их управляемости;
4. Изучение факторов, влияющих на рассеивание траекторий снарядов, и рассмотрение способов уменьшения рассеивания и повышения точности стрельбы. [1]

Математическая модель внешней баллистики, в которой снаряд представлен в виде материальной точки, используется уже давно. Полигонные исследования показали, что переходить к более сложным моделям, в которых снаряд представлен в виде твердого тела, нецелесообразно. В модели, в которой снаряд представлен в виде материальной точки, для учета движения снаряда

вокруг центра масс используется параметр формы для разных дальностей стрельбы.

I. СУЩЕСТВУЮЩИЕ РЕШЕНИЯ И СФЕРЫ ПРИМЕНЕНИЯ

Модель внешней баллистики снаряда используется в двух видах расчетов:

- Расчет траектории при известном угле бросания;
- расчет таблиц стрельбы при известных дальностях.

Расчет таблиц стрельбы более сложный и более востребованный в практике вид расчетов.

Таблицы стрельбы – сборники параметров, характеризующих стрельбу из определенного образца оружия, содержащие данные о прицеливании, траектории и результативности стрельбы. Их используются для расчета установок прицела полевой артиллерии, проектировании прицельных приспособлений стрелкового оружия, расчета поправок и прогнозирования результативности стрельбы.

Таблицы рассчитывают для нормальных метеорологических и баллистических условий стрельбы. При этом кривизна Земли и ее вращение не учитываются, ускорение силы тяжести принимается постоянным. [2]

При расчете баллистики патрона широкое распространение получил баллистический коэффициент, который отражает интегральное значение степени замедления и снижения траектории эталонного снаряда.

Баллистический коэффициент определяет относительно некоторого стандартного снаряда. На данный момент существует несколько стандартов, каждому из которых соответствует строго определенная форма и размер снаряда. Баллистический коэффициент зависит от формы патрона, его веса и калибра. На силу сопротивления воздуха влияет скорость пули, но поскольку она не учитывается в баллистическом коэффициенте необходимо использовать балли-

стический коэффициент того стандарта, который больше всего похож на тестируемую пулю.

На текущий момент разработано и используется множество баллистических калькуляторов и баллистических движков. Существуют программные продукты для самых распространенных операционных систем. Кроме того, разработаны мощные баллистические калькуляторы для мобильных устройств. Они в основном предназначены для помощи при калибровке и внесении поправок в прицельные приспособления и пользуются спросом у охотников, спортивных и гражданских стрелков

Одно из мощнейших приложений для операционных систем iOS и Android «Стрелок ПРО» разработано с применением самых последних знаний в области баллистики. [3] Также существуют приложения для настольных компьютеров, например, Dexadine Ballistic Explorer, ON TARGET, QuickLOAD и другие. Эти приложения позволяют выбирать характеристики множества патронов из имеющихся баз знаний, задавать собственные характеристики патрона, выбирать, по каким алгоритмам будет проводиться расчет и какие параметры должны использоваться. Некоторые программные продукты позволяют редактировать и загружать собственные формулы моделирования динамики тела, комбинировать несколько формул при расчете.

Безусловно, эти приложения являются очень мощными и полезными для охотников, стрелков любителей, исследователей и производителей стрелкового оружия и патронов. Однако не все приложения могут быть использованы для симуляции стрельбы танков и артиллерии на сверхдальние дистанции, запуска ракет, с изменяющейся скоростью и массой. Также они не могут быть использованы в качестве ядра игрового физического движка.

От современных систем моделирования динамики твердого тела недостаточно простого указания точки начала и конца движения. Необходимо, чтобы система могла показывать актуальные и корректные результаты моделирования в каждый момент времени. Промежуточные значения являются крайне необходимыми при предсказании поведения объекта, систем управления, промышленных разработках, исследованиях, аэрокосмическом моделировании и других сферах использования. Некоторые отрасли требуют отображения множества актуальных данных в режиме реального времени.

II. УРАВНЕНИЯ МОДЕЛИРОВАНИЯ ДИНАМИКИ ТВЕРДОГО ТЕЛА

Имеется множество уравнений и подходов моделирования динамики тела. Они могут значительно отличаться по точности, простоте, условиям использования. Например, для расчетов элементов траектории с учетом влияния воздушной среды используются такие методы как:

1. Численное интегрирование разностным методом. Позволяет рассчитать элементы траектории, заданной таблично, используя интерполирующую функцию [4];
2. Приближенные аналитические методы расчета траектории. Основаны на использовании зависимостей сопротивления воздуха, начальных условий, некоторых допущений и использования специальных табличных значений или функций;
3. Подobie траектории и табличные методы решения.
Для учета сопротивления воздуха используются такие законы как:
 1. Квадратичный закон, предложенный Ньютоном. Справедлив для скоростей 250 м/с;
 2. Степенной закон Маиевского – Забудского (1982-1985). Диапазон скоростей до 1000 м/с был разбит на 7 участков, каждому из которых соответствует свое значение;
 3. Закон Сиаиччи (1896 г.), в котором итальянский баллистик предложил аналитическую зависимость в диапазоне скоростей от 0 до 1200 м/с;
 4. Закон 1943 г. получен в Артиллерийской академии им. Ф. Э. Дзержинского на основании многочисленных опытов для диапазона скоростей до 2000 м/с и сведенный в таблицы;
 5. Закон 1958 г., разработан для оперенного эталонного снаряда. Также представлена в виде таблиц для скоростей до 2000 м/с.

ЗАКЛЮЧЕНИЕ

На основании изложенного материала авторская коллегия ставит перед собой цель разработать современную высокопроизводительную систему моделирования динамики твердого тела. Используемые модели и алгоритмы должны предоставлять возможность получать точные данные описывающие состояние тела и его положение в пространстве в каждый момент времени. Результаты расчетов должны быть доступны в режиме реального времени. Производительность системы должна позволять провести несколько параллельных расчетов различных тел с отображением результатов в различном виде в реальном времени.

СПИСОК ЛИТЕРАТУРЫ

1. Гуськов, А. В. Внешняя баллистика : учеб. пособие / А. В. Гуськов, К. Е. Милевский, А. В. Сотенко. – Новосибирск: Изд-во НГТУ, 2010. – 188 с.
2. Степанов, А. А., Лебединец А. Н. Расчеты внешней баллистики в исследованиях эффективности стрельбы / А. А. Степанов // Инженерный вестник. – 2015. – №9. – С. 537-542.
3. Стрелок Про [Электронный ресурс]. – Режим доступа: <http://www.borisov.mobi/>. –Дата доступа: 06.10.2019
4. Дмитриевский, А. А. Внешняя баллистика: Учебник для студентов вузов. – 4-е изд., перераб. и доп. / А. А. Дмитриевский, Л. Н. Лысенко – М. : Машиностроение, 2005. – 608 с.

ПРОГНОЗИРОВАНИЕ ПОГОДЫ С ПОМОЩЬЮ ЦЕПЕЙ МАРКОВА

Чистова К. О., Гуринович А. Б.

Факультет информационных технологий и управления, кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: lublubulbu@gmail.com, gurinovich@bsuir.by

В настоящее время цепи Маркова широко используются при гидрометеорологических исследованиях. В основном их применяют для качественного анализа гидрометеорологических явлений, однако представляет интерес использования цепей Маркова для прогнозирования вероятностей состояний погоды. В данной работе я использую цепь Маркова первого порядка для прогнозирования вероятностей состояний погоды на апрель.

ВВЕДЕНИЕ

Смену состояний погоды можно рассматривать как Марковский процесс, т.к. состояния являются случайными (все процессы, имеющие развитие во времени, являются стохастическими/случайными), но в них существует влияние предыдущих состояний на последующие. Имеется дискретное количество состояний погоды в зависимости от наличия, отсутствия осадков и степени облачности: облачно, малооблачно, пасмурно, солнечно, идёт дождь, идёт град, идёт гроза. На основании данных из дневника погоды [1] для апреля за последние десять лет была получена квадратная матрица вероятностей переходов P (см. табл. 1).

I. ФУНДАМЕНТАЛЬНАЯ МАТРИЦА

При степени n , стремящейся к бесконечности, матрица в данной степени, умноженная на матрицу P , останется равной матрице P^n , т.е. $P^n * P = P^n = A$, где A – предельное состояние матрицы переходов вероятностей. В матрице все элементы в каждом столбце будут равняться между собой с заданной точностью (см. табл. 2). Предельная матрица была получена на двенадцатой итерации, это означает, что на двенадцатый день и последующие дни состояние погоды будет не будет зависеть от состояния системы в предыдущие дни при заданной точности равной 0.0001. Строки предельной матрицы равны между собой и представляют собой вектор предельных вероятностей a . Элемент a_j равняется вероятности появления состояния j в апреле, с предельной точностью совпадает с вероятностью появления состояния j в апреле, полученной на основе данных из дневника погоды для апреля.

II. СРЕДНЕЕ ВРЕМЯ ПРЕБЫВАНИЯ ПРОЦЕССА ДО ДОСТИЖЕНИЯ ВЫБРАННОГО СОСТОЯНИЯ

Для определения среднего время пребывания процесса до достижения состояния j в каждом из остальных состояний необходимо данное состояние сделать поглощающим. Для поглощающего состояния вероятность p_{jj} равна едини-

це, все остальные элементы строки для состояния j соответственно равны нулю. Т.к. состояние j можно получить из всех остальных состояний, то данная цепь будет являться поглощающей цепью Маркова. Например, матрица N (см. табл. 3), выражающая среднее время пребывания процесса до достижения состояния *Солнечно* в любом из состояний, определяется по формуле $N = (J - Q)^{-1}$, где J – единичная матрица, Q – матрица переходов между непоглощающими состояниями. [2]

III. СРЕДНЕЕ ВРЕМЯ ДОСТИЖЕНИЯ

Важной характеристикой является продолжительность времени, затраченного на прохождение из состояния i в состояние j , т.е. так называемое время первого достижения. Матрица средних времен достижения (см. табл. 4) определяется по формуле $M = (J - Z + E * Z_d)$, где E – матрица, все элементы которой равны единице, Z_d – диагональная матрица, образованная из матрицы A , D – диагональная матрица, у которой элемент $d_{jj} = 1/a_j$, J – единичная матрица, матрица Z определяется по формуле $Z = (J - P + A)^{-1}$. Из матрицы M можно получить следующую информацию: элемент m_{ij} указывает количество дней, через которые можем ожидать состояние j после состояния i . [3] Для независимых событий выполняется равенство $M = E * D$. Для независимых событий возможно определить статистические показатели, однако для них невозможно определить время совершения событий с достаточной точностью из-за отсутствия зависимости между состояниями. Из полученных результатов была составлена матрица $E * D$ (см. табл. 5).

ЗАКЛЮЧЕНИЕ

Для событий, рассматриваемых в данной работе, равенство $M = E * D$ не выполняется, что подтверждает наличие зависимости между состояниями, однако для зависимых событий исключена возможность использования статистического анализа для определения парамет-

ров процесса. Следовательно, время совершения перехода к погодному состоянию может быть определено только с некоторой долей вероятности, однако цепи Маркова высших порядков дают хороший результат.

2. Поглощающие цепи Маркова [Электронный ресурс] – Режим доступа : <https://studwood.ru/1806897/matematika-himiya-fizika/pogloschayuschie-markovskie-tsepi>. – Дата доступа : 22.04.2019.

СПИСОК ЛИТЕРАТУРЫ

1. Дневник погоды [Электронный ресурс] – Режим доступа: <http://rp5.by/Дневник-погоды-в-Минске>. – Дата доступа: 15.03.2019.

3. Применение теории однородных марковских цепей для прогнозирования сроков наступления событий [Электронный ресурс] – Режим доступа : <https://pandia.ru/text/78/165/56375.php>. – Дата доступа : 22.04.2019.

Таблица 1 – Квадратная матрица вероятностей переходов

Сегодня/завтра	Облачно	Малооблачно	Пасмурно	Солнечно	Дождь	Град	Гроза
Облачно	0.2405	0.1899	0.2500	0.0759	0.2278	0.0127	0.0253
Малооблачно	0.3333	0.2167	0.1667	0.0667	0.1833	0.0167	0.0167
Пасмурно	0.2413	0.1207	0.2241	0.0344	0.3445	0.0172	0.0172
Солнечно	0.1538	0.3462	0.0769	0.2308	0.1154	0.0385	0.0385
Дождь	0.1909	0.2000	0.1727	0.0545	0.4455	0.0182	0.0091
Град	0.1250	0.1250	0.1250	0.1250	0.2500	0.1250	0.1250
Гроза	0.1250	0.2500	0.1250	0.1250	0.1250	0.1250	0.1250

Таблица 2 – Фундаментальная матрица

Сегодня/завтра	Облачно	Малооблачно	Пасмурно	Солнечно	Дождь	Град	Гроза
Облачно	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Малооблачно	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Пасмурно	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Солнечно	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Дождь	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Град	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034
Гроза	0.2444	0.1734	0.1917	0.0626	0.3213	0.0031	0.0034

Таблица 3 – Матрица N

Сегодня/завтра	Облачно	Малооблачно	Пасмурно	Дождь	Град	Гроза
Облачно	5.3226	3.6129	4.0000	6.1613	0.4193	0.4193
Малооблачно	4.3548	4.5806	3.8710	6.0000	0.4193	0.4193
Пасмурно	4.4194	3.6452	5.0645	6.4516	0.4194	0.4194
Дождь	4.3226	3.7419	4.0323	7.6129	1.5161	0.4194
Град	3.7097	3.1290	3.4194	5.4839	0.4194	0.5161
Гроза	3.1613	2.6774	2.9032	4.5161	0.4561	1.4516

Таблица 4 – Матрица средних времен достижения

Сегодня/завтра	Облачно	Малооблачно	Пасмурно	Солнечно	Дождь	Град	Гроза
Облачно	4.0913	5.9697	5.2649	20.0863	4.0512	324.6791	297.0136
Малооблачно	3.6009	6.6208	5.5812	20.2430	4.3064	324.9269	293.6618
Пасмурно	4.1352	6.6714	5.2161	20.2925	3.4618	324.0795	294.0654
Солнечно	4.3437	4.0662	6.4759	15.9677	4.9481	325.5704	294.2216
Дождь	4.4392	6.7265	5.6119	20.5290	3.1124	320.6137	294.2952
Град	5.4428	7.7256	6.6152	21.5333	1.0024	320.6167	295.0524
Гроза	4.8925	1.5274	6.6903	21.5713	1.5800	323.2010	294.6376

Таблица 5 – Матрица $E * D$

Сегодня/завтра	Облачно	Малооблачно	Пасмурно	Солнечно	Дождь	Град	Гроза
Облачно	4.0913	4.0913	4.0913	4.0913	4.0913	4.0913	4.0913
Малооблачно	5.7662	5.7662	5.7662	5.7662	5.7662	5.7662	5.7662
Пасмурно	5.2161	5.2161	5.2161	5.2161	5.2161	5.2161	5.2161
Солнечно	15.9676	15.9676	15.9676	15.9676	15.9676	15.9676	15.9676
Дождь	3.1124	3.1124	3.1124	3.1124	3.1124	3.1124	3.1124
Град	320.6156	320.6156	320.6156	320.6156	320.6156	320.6156	320.6156
Гроза	294.6376	294.6376	294.6376	294.6376	294.6376	294.6376	294.6376

Секция «Системы обработки
информации»

Section «Systems of information
processing»

СНИЖЕНИЕ УРОВНЯ СПЕКЛ-ШУМА В ЦИФРОВЫХ ИЗОБРАЖЕНИЯХ НА ОСНОВЕ 2-D НЕРАЗДЕЛИМОГО Q-ПУБФ С ИСПОЛЬЗОВАНИЕМ ПОРОГОВОЙ НЕЙРОННОЙ СЕТИ

Аврамов В. В., Рыбенков Е. В., Петровский Н. А.

Кафедра электронных вычислительных средств,

Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: avramov.vladislav@gmail.com, {rybenkov, nick.petrovsky}@bsuir.by

В работе рассматриваются методы адаптивного шумоподавления на основе пороговых нейронных сетей в вейвлет-домене. Предлагается пороговая нейронная сеть (TNN) с 2D неразделимым параунитарным банком фильтров на основе алгебры кватернионов. Высокие характеристики многополосного (2-D NS Q-PUBF) (структура '64in-64out', $CD_{2D} = 17.15$ дБ, банк фильтров прототипа 8×24 Q-PUBF), формирующего базис для TNN, обеспечивают сравнимые результаты шумоподавления по сравнению с подходами, основанными на двухканальном вейвлет-преобразовании для спекл-шума.

ВВЕДЕНИЕ

Шумы и артефакты в методах визуализации влияют на качество изображения и, как следствие, на их интерпретацию человеком, а также на точность принятия решений методов диагностики в случае медицинской визуализации. Следовательно, редактирование шумов в изображении становится главным требованием для многих практических приложений. В общем случае задача редактирования шумов заключается в получении восстановленной копии полезного сигнала x из его искаженной версии $y = x + \xi$, где ξ , например, аддитивный Гауссовый белый шум или спекл-шум. Таким образом, целью обработки является получение на выходе системы редактирования шумов f восстановленного сигнала $\hat{x} = f(y)$ как можно более близкого к полезному сигналу x :

$$\min_{x, \hat{x} \in \mathbb{R}} E(x, \hat{x}) = \min_{x, \hat{x} \in \mathbb{R}} \|x - \hat{x}\|^2. \quad (1)$$

I. ПОРОГОВОЕ РЕДАКТИРОВАНИЕ ШУМА

При оценке сигнала шума ξ может и должна использоваться любая доступная априорная информация о распределении энергии сигнала и шума, которая может быть получена путем перевода исходного сигнала в другое пространство, на основе линейного ортогонального преобразования T , как показано в [1] (см. рис. 1). Применение такого преобразования для искаженной версии сигнала позволяет изменить исходное распределение сигнала, что обеспечивает важную информацию для осуществления разделения сигнала и шума. Таким образом, полезный сигнал может быть извлечен из его искаженной версии путем обработки области преобразования некоторой пороговой функцией, сохраняющей энергию коэффициентов преобразования $T(x + \xi)$ попадающих в область распределения полезного сигнала, и подавляющей энергию шумовых ко-

эффициентов, не попавших в данную область. На этой концепции и основываются пороговые методы редактирования шумов.

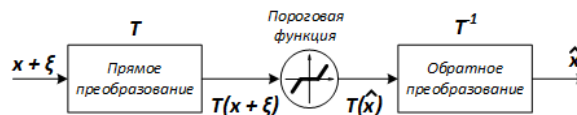


Рис. 1 – Схема порогового редактирования шумов

Помимо выбора конкретной пороговой функции применительно к решаемой задаче, выбор оптимального порогового значения также играет решающую роль в процессе шумоподавления.

II. ПОРОГОВАЯ НЕЙРОННАЯ СЕТЬ

Концепция, лежащая в основе пороговой нейронной сети (thresholding neural network (TNN)) [2], существенно отличается от классических нейронных сетей. Обучение данного типа нейронных сетей направлено на подстройку параметров пороговой функции активации, с целью достижения оптимального порога усечения, в отличие от классических нейронных сетей, где обучение заключается в подстройке весовых коэффициентов.

Фиксированная структура классических "жесткой" и "мягкой" пороговых функций и их производных, их зависимость от порогового значения, а иногда и отсутствие дифференцируемости, уменьшают их функциональность и гибкость при решении оптимизационной задачи обозначенной в выражении 1. Поэтому, в [2, 3, 4] были предложены разновидности дифференцируемых версий пороговых функций. Например, функция, показанная в [3], а также применяемая в [4], помимо значения порога t также имеет 2 параметра m и k отвечающих за форму функ-

ции:

$$f(x, t, m, k) = \begin{cases} x + (k-1)t - \frac{kt^m}{2x^{m-1}}, & x > t \\ \frac{k|x|^{m+(2-k)/k}}{2t^{m+2(1-k)/k}} \text{sign}(x), & |x| \leq t \\ x - (k-1)t + \frac{k(-t)^m}{2x^{m-1}}, & x < -t \end{cases}$$

Параметр k определяет асимптоту функции. Другими словами, если $k = 1$, то функция стремится к "жесткому" порогу, а когда $k \rightarrow 0$ соответственно к "мягкому". Поэтому, настраивая параметр k можно определить форму пороговой функции.

III. СИСТЕМА РЕДАКТИРОВАНИЯ ШУМА

Система редактирования шума состоит из пороговой нейронной сети, 2-D неразделимых банков фильтров анализа и синтеза в алгебре кватернионов (2-D NS Q-PUBF) [5]. Поток данных системы показан на рис. 2. Прямые и обратные преобразования реализованы на структуре "64in-64out" для $N = 2$, ядром которого 8-канальный Q-ПУБФ. Блок изображения с шумом размером 8×8 пикселей является входом системы, а блок восстановленного изображения размером 8×8 формируется на выходе обратного преобразования.

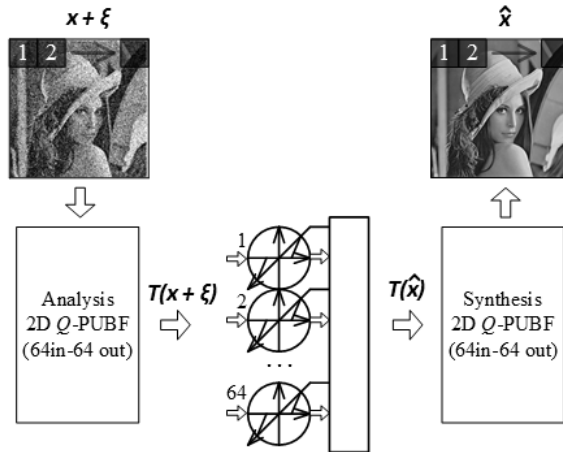


Рис. 2 – Схема пороговой обработки

Для оценки работоспособности предлагаемого подхода, в качестве тестовых данных использовались стандартные изображения *Lena*, *Barbara*, *Boat*.

Все изображения были искажены гауссовым шумом с отклонением σ равными 10, 20 и 30, результаты оценки качества реконструкции сигнала по метрике PSNR представлены в [6].

Особый интерес представляет задача редактирования спекл-шума, важным фактором является эффективность системы редактирования шума как в однородных (низкочастотных) областях изображения, так и в высокочастотных областях изображения, т.е. сохранение граничной информации.

Для анализа производительности предлагаемой системы редактирования шумов в применительно к задаче снижения уровня спекл-шума

использовались MSE , S/MSE , β и ρ (таблица 1), как это предложено в [7], где два последних параметра определены как критерии сохранения границ – β и мера подавления шума соответственно – ρ .

Результаты полученные на основе предлагаемого подхода в сравнении с известными подходами к редактированию шума на изображениях являются сравнимыми, т.о. рассмотренная схема может быть перспективна для редактирования шумов в медицинских и космических изображениях.

Таблица 1 – Характеристики системы шумоподавления для спекл-шума в сравнении с известным подходом

	Исходн. Изобр. + Шум	TNN Nasri [3]	Пред-ложен. подход	σ	Исходн. Изобр.
MSE	1622.5	134.33	216.83	0.1	<i>Lena</i>
S/MSE	10.339	19.92	19.12		
β	0.1176	0.240	0.1402		
ρ	0.7542	0.9716	0.9520		
MSE	3053.5	245.99	319.21	0.2	
S/MSE	7.5867	15.84	17.44		
β	0.0822	0.1162	0.0947		
ρ	0.6247	0.95	0.9305		
MSE	790.79	186.68	223.82	0.05	<i>Barbara</i>
S/MSE	13.261	19.320	18.77		
β	0.4148	0.6530	0.6222		
ρ	0.8863	0.9680	0.9618		
MSE	166.65	94.97	83.80	0.01	
S/MSE	20.025	22.48	23.04		
β	0.7109	0.8040	0.8323		
ρ	0.9731	0.9845	0.9859		

1. Krim, H. On Denoising and Best Signal Representation / H. Krim, et al. // IEEE Transactions on Information Theory. – 1999. – Vol. 45. – P. 2225–2238.
2. Zhang X.-P. Nonlinear Adaptive Noise Suppression Based On Wavelet Transform / X.-P. Zhang, M. Desai // Proceedings of the ICASSP. – 1998. – Vol. 3. – P. 1589–1592.
3. Nasri, M. Image denoising in the wavelet domain using a new adaptive thresholding function / M. Nasri, H. Nezamabadi-pour // Elsevier Journal of Neurocomputing. – 2009. – Vol. 72. – P. 1012–1025.
4. Bhutada, G. G. Image enhancement by wavelet-based thresholding neural network with adaptive learning rate / G. G. Bhutada, R. S. Anand, S. C. Saxena // IET Image Processing. – 2011. – Vol. 5, № 7. – P. 573–582.
5. Petrovsky, N. A. 2-D non-separable integer implementation of paraunitary filter bank based on the quaternionic multiplier block-lifting structure / N. A. Petrovsky, E. V. Rybenkov // 2019 27th European Signal Processing Conference (EUSIPCO) IEEE. – 2019.
6. Avramov, V. V. Image Enhancement by 2-D non-Separable Quaternionic Filter Bank-based Thresholding Neural Network / V. V. Avramov, E. V. Rybenkov, N. A. Petrovsky // PRIP'2019 Pattern Recognition and Information Processing. – 2019. – P. 207–212.
7. Sattar, F. Image enhancement based on a nonlinear multiscale method / F. Sattar, et al. // IEEE Transactions on Image Processing. – 1997. – Vol. 6, № 6. – P. 888–895.

ВЫБОР ТИПА ИНТЕРВЕНЦИИ НЕСТАЦИОНАРНОГО ВРЕМЕННОГО РЯДА ПО ЗАДАНЫМ ШАБЛОНАМ

Адерейко А. Д., Жук Е. Е.

Кафедра математического моделирования и анализа данных, Белорусский государственный университет
Минск, Республика Беларусь

E-mail: adereyko@yandex.ru, zhukee@mail.ru

Исследуемая проблема заключается в выборе шаблона, ближайшего к реализации нестационарного временного ряда, характеризуемого наличием интервенции.

ВВЕДЕНИЕ

В различных прикладных областях, в том числе для экономики и финансов, важно спрогнозировать не само значение цены, а ее поведение, то есть тенденцию, определяемую трендами (особенно во времена резкого повышения или понижения цены).

Производится исследование над выбором трендовой модели (шаблона) [2, 3], ближайшей к реализации $X = \{x_t\}_{t=1}^T$ длительности T нестационарного временного ряда (ВР):

$$x_t = f(t) + u_t, \quad (1)$$

где $f(\cdot)$ – реальный тренд, а случайные величины $u_{t=1}^T$ имеют смысл ошибок наблюдений и удовлетворяют условиям:

- имеют нулевые математические ожидания;
- имеют одинаковые ограниченные дисперсии;

Пусть трендовые модели $\{\Omega_l\}_{l \in S}$ задаются своими типовыми (базовыми) трендами [2, 3] $\{f_l(\cdot)\}_{l \in S}$, ($S = \{1, \dots, L\}$ – множество номеров этих моделей, $L \geq 2$), причем

$$\sum_{t=1}^T f_l(t) = 0, l \in S.$$

Получим семейства сдвига:

$$f_l^h = f_l(t) + h, t = \overline{1, T}, h \in R, l \in S,$$

где h – параметр сдвига.

Для принятия решений будем использовать решающее правило (РП) по методу наименьших квадратов (МНК) [3]. В качестве меры эффективности вычислим риск (вероятность ошибочно определить ближайший к реализации шаблон).

1. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ И ПОСТАНОВКА ЗАДАЧИ

По реализации $X = \{x_t\}_{t=1}^T$ нестационарного ВР длительности T необходимо решить, к какой трендовой модели из $\{\Omega_l\}_{l \in S}$ она «ближе». Заранее нужно определить понятие «близость», а также предложить критерий эффективности принимаемых решений [1].

По сути шаблон и является трендом из (1). Примеров шаблонов существует много, но необходимо выбрать наиболее «близкий» из них, для поиска которого предложено использовать МНК.

Примеры наиболее часто встречающихся шаблонов:

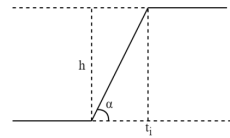


Рис. 1 – Зависит от α, h ($f_{\alpha, h}$)

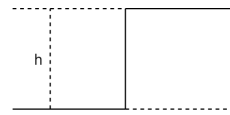


Рис. 2 – Зависит от α, h ($f_{\alpha, h}$)

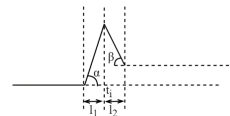


Рис. 3 – Зависит от α, β, l_1, l_2 , ($f_{\alpha, \beta, l_1, l_2}$)

Для простоты вычислений можно рассмотреть только один из нескольких промежутков, содержащих интервенцию.

Пусть реализация содержит N промежутков с интервенцией, далее будет вестись работа с любым из них (например, i -м). Каждый из этих промежутков имеет точку экстремума, обозначим её t^i , тогда рассматриваемый промежуток:

$$\Delta T = [t^i - m\Delta t, t^i + n\Delta t], (m, n \geq 0)$$

разделен на $m + n$ отрезков длины Δt (всего $m + n + 1$ точка).

Пусть имеется реализация $X = \{x_t\}_{t=t^i-m\Delta t}^{t^i+n\Delta t}$ длительности $|\Delta T| = \Delta t(n + m + 1)$. нестационарного временного ряда. Отсчеты $x_t \in R, t = t^i - m\Delta t, t^i + n\Delta t, (m, n \geq 0)$ с учетом ошибок расположены возле своего тренда (см(1)).

Для (1) характерно, что

$$\{u_t\}_{t=t^i-m\Delta t}^{t^i+n\Delta t}$$

несут в себе смысл ошибок наблюдений и обладают следующими свойствами:

$$\begin{aligned}
& E \{u_t\} = 0, \\
& D \{u_t\} = E \{u_t^2\} = \sigma^2 < +\infty, \\
& E \{u_t u_l\} = 0, \\
& \forall t, l = \overline{t = t^i - m\Delta t, t^i + n\Delta t}, \\
& l \neq t, m, n \geq 0
\end{aligned}$$

II. РАЗБОР ОДНОГО ИЗ ШАБЛОНОВ

Рассмотрим только один из шаблонов. Работа будет вестись с шаблоном, изображенным на (рис. 3). Запишем для него аналитическое представление:

$$f_{\alpha, \beta, l_1, l_2}(t) = \begin{cases} 0, & \text{если } t < t^i - m\Delta t, \\ (t - t^i m\Delta t)tg\alpha, & \\ \text{если } t^i - m\Delta t \leq t \leq t^i - m\Delta t + l_1, & \\ -(t - t^i m\Delta t - l_1)tg\beta + l_1tg\alpha, & \\ \text{если } t^i - m\Delta t + l_1 \leq t \leq t^i + n\Delta t, & \\ l_1tg\alpha - l_2tg\beta, & \text{если } t^i + n\Delta t < t. \end{cases}$$

Данный шаблон определяет модели $\Omega_{\alpha, \beta, l_1, l_2}$.

Модель отцентрируется относительно нулевого уровня с помощью параметра сдвига h ($h \in R$):

$$\begin{aligned}
f_{\alpha, \beta, l_1, l_2}^h(t) &= f_{\alpha, \beta, l_1, l_2}(t) + h, \\
t &= \overline{t^i - m\Delta t, t^i + n\Delta t}.
\end{aligned}$$

Таким образом, $f_{\alpha, \beta, l_1, l_2}$ не зависит от абсолютных значений (h) тренда.

III. ОСНОВНЫЕ РЕЗУЛЬТАТЫ

Обозначим $p = \{\alpha, \beta, l_1, l_2\}$ – наборы четырех параметров, тогда

$$\begin{aligned}
S &= \{p : \alpha \in [-\pi, \pi], \beta \in [-\pi, \pi], \\
& l_1 \in [0, +\infty), l_2 \in [0, +\infty)\}
\end{aligned}$$

Воспользуемся МНК для построения РП по реализации $X = \{x_t\}_{t=t^i-m\Delta t}^{t^i+n\Delta t}$ длительности $|\Delta T| = \Delta t(n+m+1)$ [1, 3]. РП $d = d(X) \in S$ относит реализацию к той модели из Ω_p , к базовому тренду (шаблону) которой она ближе. Решающее правило имеет следующий вид:

$$\begin{aligned}
d(X) &= \arg \min_p \min_{h \in R} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} (x_t - f_p^h(t))^2 = \\
&= \arg \min_p \min_{h \in R} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} (x_t - f_p(t) - h)^2 = \\
&= \arg \min_p \min_{h \in R} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} (x_t - f_p(t) - \bar{x}_T)^2,
\end{aligned}$$

где $\bar{x}_T = \frac{1}{\Delta T} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} x_t$ – арифметическое (выборочное) среднее [3] значение отсчетов из ре-

ализации $X = \{x_t\}_{t=t^i-m\Delta t}^{t^i+n\Delta t}$, которое является решением экстремальной задачи

$$\sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} (x_t - f_p - h)^2 \rightarrow \min_{h \in R}.$$

С другой стороны необходимо определить множество D° (по аналогии с [1]) номеров тех моделей из Ω_p , к которым тренд $f(\cdot)$ ближе:

$$D^\circ = \{p : \rho_*(f, f_p)\} = \min_{p \in S} \rho_*(f, f_p),$$

где

$$\rho_*^2(f, f_p) = \min_{h \in R} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} (f(t) - \bar{f}_T - f_p)^2,$$

$\bar{f}_T = \frac{1}{\Delta T} \sum_{t=t^i-m\Delta t}^{t^i+n\Delta t} f(t)$ имеет смысл значения параметра сдвига, «приводящего» реальный тренд $f(\cdot)$ к нулевому уровню, а $\rho_*(f, f_p)$ – евклидово расстояние между трендом $f(\cdot)$ реализации X и базовым трендом $f_p(\cdot)$, определяющим модель Ω_p ($p \in S$).

В качестве меры эффективности используется риск [1] $r_T = P\{d(X) \notin D^\circ\}$ [1], имеющий смысл вероятности не отнести при помощи РП реализацию X к ближайшей трендовой модели. Если $D^\circ = \{d^\circ\}$, существует только одна ближайшая к тренду $f(\cdot)$ реализации X модель из $\{\Omega_p\}$ и

$$r_T = P\{d(X) \neq d^\circ\},$$

где

$$d^\circ = \arg \min_p \rho_*(f, f_p)$$

Чем меньше r_T , тем эффективнее принимаемые при помощи РП решения.

Вычислим риск РП (как и в [1]) $d = d(X)$:

– истинный номер ближайшего базового тренда.

Таким образом, был найден ближайший к реализации базовый тренд для одного промежутка (i -го). После применения данного алгоритма на N промежутках сформируется оценка множества $\{d_1^\circ, d_2^\circ, \dots, d_N^\circ\}$. И необходимый номер ближайшего базового тренда будет оценен d^* . Поставленная задача решена.

СПИСОК ЛИТЕРАТУРЫ

1. Жук, Е. Е. Статистическое отнесение реализаций нестационарных временных рядов к заданным трендовым моделям // Вест. Нац. акад. наук Беларуси, Сер. физ – мат. наук, – 2017. – № 2. – С. 52-59.
2. Андерсон, Т. Статистический анализ временных рядов: пер. с англ. / Т. Андерсон. – М: Мир, 1976. – 759 с.
3. Харин, Ю. С. Математическая и прикладная статистика / Ю. С. Харин, Е. Е. Жук. – Минск: БГУ, 2005. – 276 с.

ПРОБЛЕМА ПРОМЕЖУТОЧНОЙ СИНХРОНИЗАЦИИ ДАННЫХ В МНОГОСТУПЕНЧАТОМ ПРОЦЕССЕ РАЗРАБОТКИ СОВРЕМЕННОГО РАДИОЭЛЕКТРОННОГО ИЗДЕЛИЯ

Андрухович А. К., Герман Ю. О., Сманцер М. С.

Кафедра программного обеспечения информационных технологий, кафедра информационных технологий автоматизированных систем, кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: anastasia.andruhovich@gmail.com, julgerman@gmail.com, drowmar@gmail.com

В статье рассмотрена проблема промежуточной синхронизации данных в процессе разработки современного радиоэлектронного изделия. Перечислены основные этапы конструирования конечного продукта, а также возникающие в процессе разработки ошибки переноса данных из различных сред проектирования. Проведен анализ появления ошибок и предоставлено решение проблемы промежуточной стыковки этапов разработки изделия. Обосновано создание программного продукта для распознавания, коррекции ошибок и интерактивной модификации данных для последующего этапа разработки радиоэлектронного изделия.

ВВЕДЕНИЕ

Процесс разработки современного радиоэлектронного изделия представляет собой сложный многоступенчатый алгоритм. В нем можно выделить два основных этапа:

- этап разработки электронных модулей;
- этап проектирования механических деталей.

Важнейшими составляющими конструкции аппаратной части являются электронные модули (печатные платы), обеспечивающие функционирование устройства, и различные механические детали, задающие их оптимальное расположение в корпусе изделия. Электронные модули разрабатываются в специализированных программных пакетах (например, «PCAD», «Altium Designer», «Mentor Graphics»). Механическая часть изделия разрабатывается в системах объемного проектирования (например, «Autodesk Inventor», «Solid Works», «Creo Parametric»). Работа в обоих видах систем проектирования обычно ведется параллельно двумя и более специалистами.

Важнейшим условием корректной разработки изделия является необходимость жесткой синхронизации геометрии и атрибутивной информации печатной платы в обеих системах проектирования после каждого изменения конструкции. Для многократной (несколько десятков или даже сотен раз) передачи данных между ними часто используются файлы формата «IDF» версии 3.0 [1].

Практический опыт показывает, что для повышения гибкости взаимодействия очень удобно иметь возможность перед импортом в систему объемного проектирования выполнить визуальный контроль и, при необходимости, не нарушающую синхронизацию интерактивную коррекцию импортируемых данных.

I. ФОРМАТ ДАННЫХ

Данные формата «IDF» представлены двумя файлами, чаще всего имеющими расширения «.emn» и «.emp». Файл с расширением «.emn» описывает геометрию печатной платы, содержит месторасположение радиоэлектронных элементов. Пример файла с расширением «.emn» представлен на рисунке 1.

```
.HEADER
BOARD_FILE 3.0 "Expedition PCB" 2017/11/16.11:00:40 1
"Rapidsum.pcb" MM
.END_HEADER
.BOARD_OUTLINE UNKNOWN
L:500000
0 58.6000000 19.6500000 0.0000000
0 58.6000000 32.3000000 0.0000000
0 53.6000000 37.3000000 0.0000000
0 5.0000000 37.3000000 0.0000000
0 0.0000000 32.3000000 0.0000000
0 0.0000000 5.0000000 0.0000000
0 5.0000000 0.0000000 0.0000000
0 53.6000000 0.0000000 0.0000000
0 58.6000000 5.0000000 0.0000000
0 58.6000000 19.6500000 0.0000000
.END_BOARD_OUTLINE
```

Рис. 1 – Пример файла с расширением «.emn»

Файл с расширением «.emp» хранит в себе информацию о контуре геометрии каждого радиоэлектронного элемента [2]. Пример файла с расширением «.emp» представлен на рисунке 2.

```
.HEADER
LIBRARY_FILE 3.0 "Expedition PCB" 2017/11/16.11:00:40
.END_HEADER
.ELECTRICAL
"RES160X45H" "RKT3B1TTD512J" MM 0.4500000
0 -0.8500000 -0.4500000 0.0000000
0 0.8500000 -0.4500000 0.0000000
0 0.8500000 0.4500000 0.0000000
0 -0.8500000 0.4500000 0.0000000
0 -0.8500000 -0.4500000 0.0000000
.END_ELECTRICAL
.ELECTRICAL
"RES160X45H" "RKT3B1TTD305J" MM 0.4500000
0 -0.8500000 -0.4500000 0.0000000
0 0.8500000 -0.4500000 0.0000000
0 0.8500000 0.4500000 0.0000000
0 -0.8500000 0.4500000 0.0000000
0 -0.8500000 -0.4500000 0.0000000
.END_ELECTRICAL
```

Рис. 2 – Пример файла с расширением «.emp»

II. ПОСТАНОВКА ЗАДАЧИ ПРОМЕЖУТОЧНОЙ СИНХРОНИЗАЦИИ ДАННЫХ

Опыт показывает, что наибольшие трудности возникают при передаче результатов разработки электронного модуля в систему разработки объемного проектирования в первый раз. При этом в системе объемного проектирования могут отсутствовать 3D-модели радиоэлектронных элементов, и тогда они создаются «на лету» по

данным из файла «.emp» [3]. Будучи неотвалированными, данные о геометрии радиоэлектронных элементов и печатной платы могут содержать незамкнутые контуры и накладывающиеся друг на друга линии. При количестве строк в импортируемых файлах, достигающем нескольких тысяч, крайне сложно выявить эти ошибки вручную.

Кроме того, положение и ориентация геометрии печатной платы относительно начала координат могут быть неудобными для дальнейшей разработки конструкции изделия в системе объемного проектирования. Отсюда вытекает требование наличия возможности поворота и смещения геометрии электронного модуля относительно начала координат. Для оценки такой необходимости инженеру достаточно предварительно увидеть графическое изображение печатной платы. Пример графического изображения печатной платы представлен на рисунке 3.

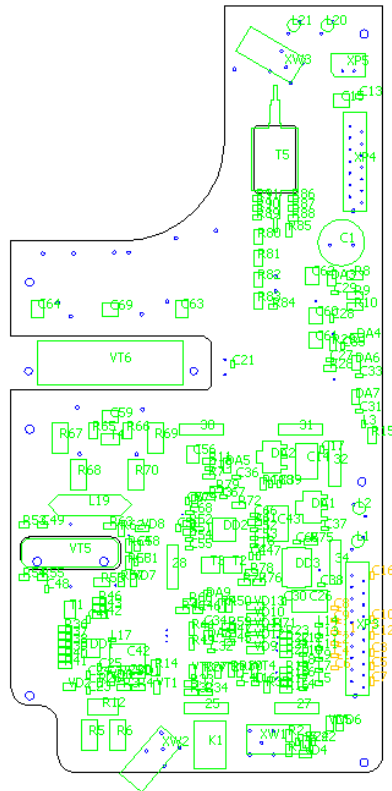


Рис. 3 – Пример графического изображения печатной платы

III. РЕШЕНИЕ ПРОБЛЕМЫ СИНХРОНИЗАЦИИ ДАННЫХ

Таким образом можно сделать вывод о целесообразности создания системы, предоставля-

ющей соответствующий функционал для визуализации и интерактивной коррекции импортируемых данных. При этом автоматическая модификация данных должна гарантировать целостность импортируемой геометрии. В случае ручной коррекции данных ответственность за их целостность пользователь принимает на себя.

Программное обеспечение должно предоставлять реализацию вышеописанного функционала, который должен в себя включать:

- чтение и запись данных в файл в формате «IDF» версии 3.0;
- графическое отображение печатной платы;
- возможность поворота на заданный угол и смещения на заданное расстояние относительно начала координат всей конструкции печатной платы;
- интерактивная коррекция ошибок в исходном файле.

Создание системы для промежуточной синхронизации данных в процессе разработки радиоэлектронного изделия позволит облегчить решение проблемы стыковки этапов проектирования изделия и значительно сэкономит время поиска и коррекции невалидных данных.

ЗАКЛЮЧЕНИЕ

В статье рассмотрена проблема промежуточной синхронизации данных формата «IDF» версии 3.0 в процессе разработки современного радиоэлектронного изделия. Проанализированы основные этапы конструирования и возникающие в процессе разработки ошибки переноса данных из различных сред проектирования. Предложено решение проблемы промежуточной стыковки этапов разработки изделия. Обосновано создание программного продукта для распознавания, коррекции ошибок и интерактивной модификации данных для последующего этапа разработки радиоэлектронного изделия.

СПИСОК ЛИТЕРАТУРЫ

1. Норенков И. П. Основы автоматизированного проектирования: Учебник для вузов. М.: Издательство МГТУ им. Н. Э. Баумана, – 2009. – с. 232.
2. Collaborating Between Altium Designer and SOLIDWORKS [Электронный ресурс] / Altium Designer and SOLIDWORKS info –Режим доступа: <http://techdocs.altium.com/display/ADON> – Дата доступа: 19.09.2019.
3. Гончаренко А. М. Сквозное проектирование сборок на печатных платах с применением систем Altium Designer и Solid Works // Наука и образование: науч. издание МГТУ им. Н. Э. Баумана. – 2015. – № 12. – DOI: 10.7463/1215.0828517

ИСПОЛЬЗОВАНИЕ БЕССЕРВЕРНЫХ ВЫЧИСЛЕНИЙ ДЛЯ РЕШЕНИЯ ЗАДАЧ ОБАРАБОТКИ ИНФОРМАЦИИ

Богдан А. А.

Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: alexeybogdan@icloud.com

В данной работе будет дано понятие бессерверным вычислениям, будут рассмотрены их достоинства и недостатки. Приведены примеры существующих облачных решений, позволяющих разрабатывать приложения, используя бессерверные вычисления. Проанализирована разница стоимости равных по мощности бессерверных и серверных вычислений на облачном сервисе Amazon Web Services.

ВВЕДЕНИЕ

Задача обработки информации - это, в большинстве случаев, высоконагруженный процесс, который требует высокие мощности и соответственно дорогостоящее оборудование. В некоторых случаях процесс обработки информации выполняется по определенному расписанию либо сторонний сервис является триггером его запуска. Это значит что дорогостоящее оборудование большую часть времени работает вхолостую.

Напротив, для такого рода приложений можно использовать подход, когда не нужно покупать дорогостоящее оборудование, а использовать бессерверные вычисления. При их использовании, стоимость зависит от того, сколько раз процесс обработки информации был запущен.

В данной работе будут рассмотрены бессерверные вычисления, которые позволят сократить издержки в несколько раз.

I. ПОНЯТИЕ БЕССЕРВЕРНЫХ ВЫЧИСЛЕНИЙ, ИХ ДОСТОИНСТВА И НЕДОСТАТКИ

Бессерверные вычисления (англ. serverless computing) — модель облачных вычислений, в которой платформа динамично руководит выделением машинных ресурсов. Иногда бессерверные вычисления также называют «Функция как услуга» (англ. Function as a Service, FaaS), потому что единицей кода является функция, которая выполняется платформой. По сути, для выполнения каждого запроса создается отдельный контейнер, который уничтожается после выполнения.

Конечно, бессерверные вычисления требуют аппаратного обеспечения. Это название используется потому, что пользователям такой платформы не надо иметь дело с настройкой серверов для запуска кода: все серверные настройки и планирование вычислительных ресурсов скрыты от пользователей и управляются платформой. Бессерверный код может быть частью приложений, построенных на традиционной архитектуре, например, на микросервисах.

Разрабатывая приложения, используя бессерверные вычисления, отсутствует необходи-

мость в обслуживании инфраструктуры, поскольку разработчики могут арендовать большую часть ресурсов, необходимых при разработке ПО. Более того, бессерверная архитектура имеет возможность масштабирования в соответствии с рабочей нагрузкой приложения. Разработчикам больше не нужно приобретать дополнительную инфраструктуру для обслуживания непредвиденного роста. Также, одним из достоинств является то, что нет необходимости писать сценарии для развертывания приложения на сервер. Для развертывания бессерверного приложения разработчикам просто нужно скомпилировать свой код, заархивировать его и загрузить на бессерверную платформу.

Бессерверные вычисления имеют и свои недостатки. Одним из них является ограничение ресурсов. Это необходимо для того, чтобы платформа могла справляться со скачками нагрузок и противостоять внешним атакам злоумышленников. Также существует риск небезопасной работы, так как разработчики запускают свои функции на общей платформе. Таким образом, при загрузке вредоносного кода возможны негативные последствия для работы других приложений.

Главным достоинством бессерверных вычислений является то, что это модель «плати, как ты используешь». Вам просто нужно заплатить за время выполнения вашей функции — продолжительность выполнения вашего кода и количество срабатываний. Сравнивая с традиционным подходом использования серверов, вам не нужно нести расходы за неиспользуемые и «незанятые» ресурсы.

II. ПРОВАЙДЕРЫ БЕССЕРВЕРНЫХ ВЫЧИСЛЕНИЙ

Приведем примеры провайдеров бессерверных вычислений:

- Amazon Lambda Functions;
- IBM Cloud Functions;
- Microsoft Azure Functions;
- Google Cloud Functions.

Более подробно рассмотрим Amazon Lambda Functions. AWS Lambda - это управляемая событиями бессерверная вычислительная служба, которая позволяет запускать код без подготовки или управления серверами и может расширять другие службы AWS с помощью настраиваемой логики. Лямбда может автоматически запускаться в ответ на несколько событий, таких как HTTP-запросы через Amazon API Gateway, изменения данных в хранилище Amazon S3 или в базе данных Amazon DynamoDB, добавление нового элемента в очередь Amazon SQS. AWS Lambda запускает код в высокодоступной вычислительной инфраструктуре и выполняет все функции администрирования базовой платформы, включая обслуживание серверов и операционных систем, выделение ресурсов и автоматическое масштабирование, мониторинг кода и ведение журнала.

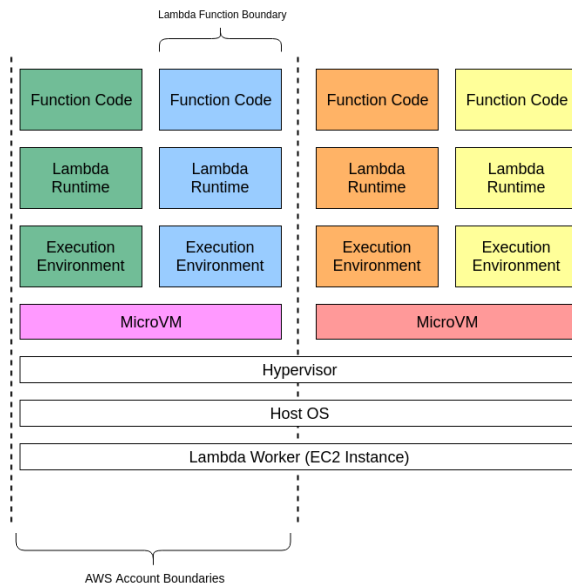


Рис. 1 – Принцип работы AWS Lambda

Рассмотрим принципы работы AWS Lambda. Каждая функция выполняется в одной или нескольких выделенных средах выполнения, которые используются в течение всего срока действия функции, а затем уничтожаются. Каждая среда выполнения содержит один параллельный вызов, но может повторно использоваться в нескольких последовательных вызовах одной и той же функции. Среда выполнения работают на аппаратных виртуальных машинах (microVM). MicroVM предназначен только для одной учетной записи AWS, но может быть повторно использован средами выполнения других функций учетной записи. MicroVM упакованы на аппаратной платформе, принадлежащей

AWS и управляемой ей (Lambda Workers). Среда выполнения никогда не передаются между функциями, а MicroVM никогда не распределяются между учетными записями AWS (см. рис 1).

III. СРАВНЕНИЕ СТОИМОСТИ СЕРВЕРНЫХ И БЕССЕРВЕРНЫХ ВЫЧИСЛЕНИЙ

Ниже приведен пример оформления простой таблицы. Рассмотрим стоимости AWS EC2 (сервис, предоставляющий пользователям облачные сервера на операционных система Linux, Windows) и AWS Lambda для следующей задачи: необходимо анализировать покупки пользователей и определить популярную категорию товара в зависимости от региона. Время выполнения такой функции примем равным 10 секунд. В таблице 1 вы можете увидеть сравнительную характеристику стоимости равных по мощности бессерверных и серверных вычислений за месяц использования при различных условиях выполнения задачи.

Таблица 1 – Сравнительная характеристика

Кол-во запусков в час	СТОИМОСТИ	
	AWS Lambda, USD	AWS EC2, USD
1	0	14.976
6	0	14.976
10	0	14.976
20	0	14.976
30	0.59	14.976
60	7.75	14.976
120	22.2	14.976

Как вы можете заметить, если функция обработки информации выполняется несколько раз в час, то стоимость такой AWS Lambda функции будет бесплатна или стоить значительно дешевле, чем выделенный сервер. Выделенный сервер имеет смысл использовать, если средняя нагрузка на него достигает более чем тридцать процентов.

СПИСОК ЛИТЕРАТУРЫ

1. Moving to Serverless with AWS Lambda [Electronic resource] / Paul Johnston. – O'Reilly. – Mode of access: <https://learning.oreilly.com/learning-paths/learning-path-moving/9781492054887>. – Date of access: 01.09.2019.
2. AWS Lambda Developer Guide [Electronic resource] / Amazon. – Amazon Web Services. – Mode of access: <https://docs.aws.amazon.com/lambda/latest/>. – Date of access: 01.02.2018.
3. In Depth AWS Lambda Overview [Electronic resource] / Georgi Velinov. – Medium. – Mode of access: <https://medium.com/faun/in-depth-aws-lambda-overview-1eeb4580696b>. – Date of access: 14.04.2019.

КЛАССИФИКАЦИЯ И ОБРАБОТКА ЭЛЕКТРОННЫХ СООБЩЕНИЙ

Богдан А. А., Лапицкая Н. В.

Кафедра программного обеспечения информационных технологий,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: alexeybogdan@icloud.com, lapan@bsuir.by

В данной статье обсуждаются методы и подходы для решения проблемы классификации сообщений с дальнейшим распределением по папкам и также их дальнейшей обработкой без участия пользователя.

ВВЕДЕНИЕ

Постоянно увеличивающийся поток входной информации, который является отличительной чертой настоящего времени, требует решения задач ее классификации даже на бытовом уровне. Скорости обработки входных данных привычной человеку уже не хватает, поэтому необходимо создавать технические решения. Пример такого канала поступления информации является электронная почта. Пользователь сталкивается с большим потоком электронных сообщений, которые они не в состоянии обработать самостоятельно. Более того, большинство сообщений - это реклама или спам рассылка, в то время как некоторые из сообщений могут быть очень важны для пользователя и нельзя допустить чтобы они были оставлены без внимания.

Сегодня, пользователи вручную создают папки и группируют свои сообщения с их помощью. Но ручная группировка может быть длительным процессом, если пользователь получает их в большом количестве. Для борьбы со спамом на почтовые сервера устанавливаются брандмауэры. Данный подход позволяет заблокировать только IP и DNS адреса. Для того чтобы обойти защиту брандмауэра, достаточно изменить IP или DNS адрес, с которого отправляется спам рассылка.

В качестве решения проблемы будут рассмотрены методы распределения по заданному условию и автоматического распределения сообщений по папкам, то есть их классификации. Также будут рассмотрен процесс автоматической обработки сообщений. Под автоматической обработкой понимается выполнение какого действия над сообщениями в зависимости от его классификации. Данные решения значительно уменьшат время обработки электронных сообщений пользователей.

I. МЕТОД РАСПРЕДЕЛЕНИЯ СООБЩЕНИЙ ПО УСЛОВИЮ

Рассмотрим простейший способ классификации электронных сообщений с технической точки зрения. Метод распределения сообщений по условию работает следующим образом: поль-

зователь создает папку в электронном ящике и создает набор условий для данной папки. Если новое сообщение удовлетворяет всем условиям данной папки, то сообщение автоматически перемещается в данную папку.

Примерами условий могут быть:

- адрес отправителя;
- дата отправки сообщения;
- размер сообщения;
- наличие прикрепленных файлов в сообщении;
- наличие тега в письме.

На рисунке 1 вы можете увидеть пример работы метода распределения сообщений по условию.

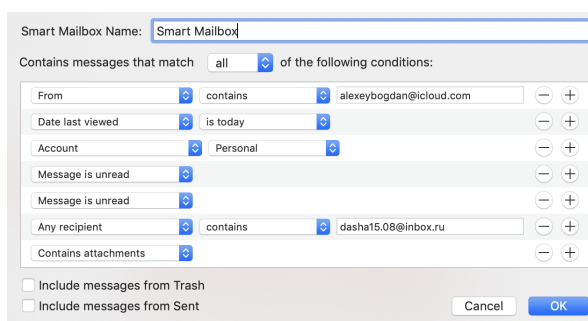


Рис. 1 – Распределение сообщений по условию в приложении Apple Mail

II. АВТОМАТИЧЕСКОЕ РАСПРЕДЕЛЕНИЕ ПИСЕМ

В случае автоматического группирования писем, алгоритм будет основан на классификации. В качестве классов будут выступать существующие папки. При синхронизации нового сообщения, сначала будет применяться метод распределения сообщений по условию. Если новое сообщение подойдет одной из папок по выражению, то оно будет перемещено в папку. Если нет, то будет применяться алгоритм автоматического группирования писем. При автоматическом распределении необходим алгоритм, который сам определит, что некоторые сообщения являются похожими и их необходимо переместить в одну папку.

В качестве метода автоматического распределения сообщений можно использовать метод максимина (Рис. 2). Он предназначен для разделения объектов на кластеры, причем количество кластеров заранее неизвестно. Оно определяется автоматически в процессе разбиения объектов. Принцип работы метода следующий. Выбирается один из объектов и назначается прототипом первого кластера. Находится объект, наиболее удаленный от выбранного и назначается прототипом второго кластера. Все объекты распределяются по двум кластерам. Каждый объект относится к кластеру, представленному ближайшим прототипом. Затем в каждом из кластеров находится объект, наиболее удаленный от своего прототипа. Если расстояние между этим объектом и прототипом кластера оказывается значительным (превышающим некоторую предельную величину), то объект становится новым прототипом, т.е. образуется новый кластер. После этого распределение объектов по кластерам выполняется заново. Процесс продолжается, пока не будет получено такое разбиение на кластеры, при котором расстояние от каждого объекта до прототипа кластера не будет превышать заданную предельную величину.

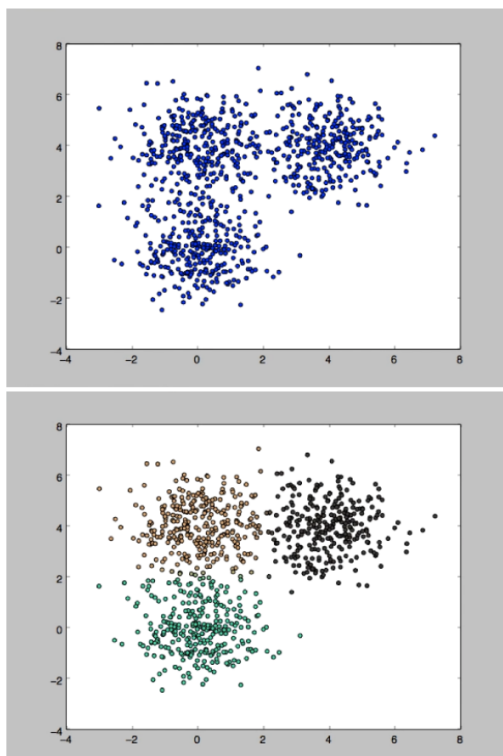


Рис. 2 – Пример работы метода максимина

III. АВТОМАТИЧЕСКАЯ ОБРАБОТКА СООБЩЕНИЙ

Под автоматической обработкой сообщений понимается механизм, который будет выполнять определенные действия на сообщении автома-

тически. Данный метод работает следующим образом: пользователь добавляет обработчики для определенной папки и при попадании нового сообщения в папку автоматические обработчики начинают выполняться. Автоматические обработки могут быть следующими:

- удаление сообщения;
- отправка push нотификации на смартфон;
- sms оповещение;
- ответ на письмо с предопределенным текстом письма.

Рассмотрим следующий пример. Вам, как пользователю, хотелось бы избавиться от постоянных писем со спамом и рекламой. Тогда вы можете сделать следующее: создать папку, в которую будут попадать письма такого рода, и добавить действие удалить письмо. Таким образом вы избавитесь от таких писем. Рассмотрим следующий вариант. Вы, как сотрудник компании, должны в срочном порядке отвечать на все письма своего начальника. В таком случае вы можете создать папку, в которую будут сохраняться все письма вашего начальника, и добавить sms оповещение. В таком случае вы не упустите из виду важные сообщения.

IV. ЗАКЛЮЧЕНИЕ

В работе представлены три подхода к решению задачи повышения скорости обработки входных электронных сообщений. Предложены подходы обработки сообщений без участия пользователя. В дальнейшем будет проведен сравнительный анализ эффективности предложенных методов в зависимости от персональных характеристик пользователя.

СПИСОК ЛИТЕРАТУРЫ

1. An Introduction to Data Science [Electronic resource] / Saint Petersburg State University. – Coursera, 2019. – Mode of access: <https://www.coursera.org/lecture/vvedeniye-v-nauku-o-dannykh/algoritm-k-means-hOVUY>. – Date of access: 25.08.2019.
2. Types of classification algorithms in Machine Learning [Electronic resource] / Mandy Sinada. – Medium, 2017. – Mode of access: <https://medium.com/@Mandysidana/machine-learning-types-of-classification-9497bd4f2e14>. – Date of access: 28.02.2017.
3. How to Organize Your Email with Smart Mailboxes in Apple Mail [Electronic resource] / Matt Clain. – HowToGeek, 2016. – Mode of access: <https://www.howtogeek.com/252635/how-to-organize-your-email-with-smart-mailboxes-in-apple-mail>. – Date of access: 07.03.2017.
4. How do email servers detect spam? [Electronic resource] / Joy Larkin. – Quora, 2016. – Mode of access: <https://www.quora.com/How-do-email-servers-detect-spam>. – Date of access: 10.01.2016.
5. Обучение без учителя: 4 метода кластеризации данных на Python [Электронный ресурс] / Библиотека программиста. – Proglib, 2019. – Метод доступа: <https://proglib.io/p/unsupervised-ml-with-python/>. – Дата доступа: 24.05.2018.

МОДИФИЦИРОВАННЫЙ АЛГОРИТМ МУРАВЬИНОЙ КОЛОНИИ ДЛЯ ПОИСКА ПУТИ В НАГРУЖЕННОЙ СЕТИ

Борисов Д. В., Кива В. С.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: divlboris@gmail.com, knijaz@gmail.com

Сформулирована задача поиска оптимального маршрута в сети, важной особенностью которой является тот факт, что каждый узел обладает нагрузженностью. Эта нагрузженность зависит от момента времени, в который происходит прохождение узла. Для решения данной задачи была произведена модификация алгоритма муравьиной колонии.

ВВЕДЕНИЕ

К алгоритмам муравьиной колонии относятся алгоритмы, появившиеся в результате исследования и обработки данных, полученных при наблюдении за реальными муравьями. Поведение муравьев было формализовано и использовано для решения сложных оптимизационных задач. На сегодняшний день эти алгоритмы являются конкурентноспособными и для некоторых задач показывают наилучшие результаты.

Муравьи являются социальными насекомыми, решающими проблемы сообщества. Это позволяет им выполнять такую работу, которая не могла бы быть выполнена отдельными муравьями. Основой алгоритма муравьиной колонии является моделирование поведения муравьев, связанного с их способностью находить кратчайший путь до пищи и отсутствие централизованного управления. А также со способностью находить новый путь, если старый по каким-либо причинам оказался недоступен. Муравьи взаимодействуют друг с другом путем прямой передачи информации или с помощью стигмержи. Стигмержи — это разнесенный во времени тип взаимодействия, когда один субъект взаимодействия изменяет некоторую часть окружающей среды, а остальные используют информацию об ее состоянии позже, когда находятся в ее окрестности[1]. Стигмержи осуществляется посредством феромонов. Феромоны это секрет специальных желез муравья, который он оставляет на земле в процессе своего перемещения. Муравьи могут воспринимать этот секрет на протяжении нескольких суток. Феромоновый след становится своего рода памятью колонии.

Нахождение муравьями кратчайшего пути начинается с выхода в случайном направлении за едой. Если на пути встречается препятствие, муравьи с равной вероятностью начнут его обходить. Достигнув еды позади препятствия, муравьи берут ее часть и отправляются в обратный путь. Столкнувшись с упомянутым препятствием они таким же образом с равной вероят-

ностью начнут его обходить. В процессе своего перемещения муравьи оставляют феромоновый след. Из-за того, что при своем движении муравьи ориентируются на концентрацию феромона, следующие муравьи выберут путь с наибольшим его значением. Наибольшее значение будет на самом коротком пути, так как по нему успеет пройти большее число муравьев. Таким образом муравьи продолжают увеличивать концентрацию феромонов на коротком пути до тех пор, пока еда не будет перенесена или путь по какой-либо причине перестанет быть доступен.

Из-за того, что в основе алгоритма муравьиной колонии лежит поиск оптимального пути, он подходит для решения задач, которые могут быть представлены в виде графов.

1. ОБОБЩЕННЫЙ АЛГОРИТМ

Обобщенный алгоритм муравьиной колонии выглядит следующим образом:

1. Создаются муравьи для поиска решения. Муравьи в зависимости от условия задачи помещаются в разные узлы или в какой-то конкретный узел. Также на этом этапе наносится феромоновый слой для того, чтобы в начале работы вероятности перехода были ненулевыми.
2. Начинается поиск решения. Для каждого муравья оценивается вероятность перехода из одной вершины в другую, которая основана на величине феромонового следа и расстоянии до узла. Больше муравьев будет перемещаться по более короткому пути к источнику пищи и откладывать больше феромонов, чем по более длинным путям.
3. Запускается процесс обновления феромонов. Каждое значение феромона уменьшается, имитируя испарения, и увеличивается, моделируя отложение феромонов муравьями на следе.
4. Выполняется проверка, по результатам которой или продолжается выполнение, или делается вывод о том, что решение найдено[2].

Работа алгоритма также зависит от следующих параметров:

- коэффициента влияния феромона на направление;
- коэффициента влияния соседнего узла;
- коэффициента испарения феромона;
- коэффициента увеличения феромона.

II. Достоинства

- Алгоритмы муравьиной оптимизации адаптивны по своей природе и позволяют адаптироваться к меняющимся условиям динамических приложений.
- Алгоритм имеет преимущество распределенных вычислений. Также можно доказать его сходимость.
- В анализе сетей реальных размерностей, чтобы проверить возможность разработки метаэвристического алгоритма, который позволяет вычислять потоки в сети быстрее, чем при использовании традиционных алгоритмов.
- Позволяет динамически перенаправлять потоки через кратчайший путь, если один узел сломан. Большинство других алгоритмов вместо этого предполагают, что сеть является статической[3].

Таблица 1 – Сравнение эвристических методов оптимизации маршрутов грузовиков

Число заказчиков	Муравьиный алгоритм	Генетический алгоритм	табу-поиск
200	6460.98	6460.98	6697.53
255	586.87	596.89	593.35
300	1007.07	1018.74	1016.83
399	927.27	933.74	936.04
420	1834.79	1846.55	1915.83

III. Недостатки

- В стандартных алгоритмах муравьиной колонии феромоновый путь и эвристические значения зависят от масштабов задачи.
- Теоретический анализ сложен.
- Сходимость гарантирована, но время сходимости неопределено.
- Есть склонность к попаданию в локальное оптимальное решение.

IV. Модификация

Для использования алгоритма муравьиной колонии при решении поставленной задачи необ-

ходимо его модифицировать. При оценке вероятности перехода к очередному узлу необходимо учитывать возможную задержку на обслуживание в этом узле. Для оценки задержки на узлах, каждому из узлов добавляется словарь. Этот словарь состоит из промежутков времени, где каждому промежутку ставится в соответствие расчетное значение времени, которое предстоит провести на узле прежде чем можно будет считать, что узел достигнут. Каждый муравей хранит информацию о своем текущем времени в пути, для того чтобы можно было однозначно оценить в какой из промежутков он попал во время посещения узла. Для поставленной задачи веса ребер отражают время в пути между каждой парой узлов. При этом время пути на ребре может быть или взято из открытых источников, содержащих информацию о загруженности дорог, или может быть рассчитано на основе известного расстояния и средней скорости.

ЗАКЛЮЧЕНИЕ

Алгоритм муравьиной колонии является хорошим выбором для задач поиска оптимального пути в графах. При этом его эффективность растет вместе с ростом количества узлов в графе. Результат решения задач напрямую зависит от параметров, от алгоритма выбора следующего узла муравьем, от правил добавления и испарения феромона. Особенностью же предложенной модификации алгоритма муравьиной колонии является возможность работать с графами, в которых нагруженность узлов меняется с течением времени. Алгоритм позволяет найти путь, при котором обход будет произведен за минимальное время. Это позволяет использовать его с высокой эффективностью в транспортных перевозках и логистике.

СПИСОК ЛИТЕРАТУРЫ

1. Штовба С.Д. Муравьиные алгоритмы // Exponenta Pro. Математика в приложениях, 2003, с.70–75.
2. Профессиональный информационно-аналитический ресурс, посвященный машинному обучению, распознаванию образов и интеллектуальному анализу данных. [Электронный ресурс] / Муравьиные алгоритмы. – Режим доступа: <http://www.machinelearning.ru/>. – Дата доступа: 07.10.2019.
3. O.Deepa, A.Senthilkumar, Swarm intelligence from natural to artificial systems: ant colony optimization. // International Journal on Applications of Graph Theory in Wireless Ad hoc Networks and Sensor Networks. – 2016. – Vol. 8, № 1. – P. 13.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ МОНИТОРИНГА ДИСТАНЦИОННОЙ ЗАЩИТЫ: ПОДХОД К ФИКСАЦИИ УТЯЖЕЛЕНИЯ РЕЖИМА

Бухаров Д. С., Гусев Р. А.

Служба автоматизированных систем диспетчерского управления, Филиал АО «СО ЕЭС» Иркутское РДУ
Иркутск, Российская Федерация
E-mail: bukharovds@gmail.com

Рассматривается подход к фиксации приближения точки нагрузки к области срабатывания дистанционной защиты. Представленный подход реализован в виде программного обеспечения, используемого для оперативного оповещения диспетчерского персонала об ухудшении электроэнергетического режима.

ВВЕДЕНИЕ

Ключевая проблема в мониторинге нахождения точек нагрузки (ТН) относительно области срабатывания (ОС) дистанционной защиты (ДЗ) – отсутствие визуального отображения ТН, ОС ДЗ и необходимой телеметрической информации (ТМ) с энергообъектов (ЭО), на которой установлена ДЗ, защищающая некоторую линию электропередачи.

Осуществление контроля режима по величине перетоков активной P и реактивной Q мощности и напряжению U не позволяет диспетчеру в полной мере определить необходимые управляющие воздействия на силовое оборудование для предупреждения возможного ложного срабатывания ДЗ.

Программное обеспечение (ПО), обеспечивающее прием необходимой ТМ, моделирование работы ДЗ по заданным уставкам и визуализацию в реальном времени текущей ситуации на энергообъекте, позволяет диспетчеру оперативно оценить текущую ситуацию и уменьшить время принятия решения на изменение режима работы энергосистемы.

I. ПО МОНИТОРИНГА ДЗ

В разрабатываемом ПО реализованы следующие функции:

- Прием данных ТМ из оперативно-информационного комплекса (ОИК) (реализован посредством стандартной динамической библиотеки ОИК OICDAS.dll [1]).
- Вычисление ТН на основе ТМ.
- Фиксация приближения ТН к ОС ДЗ.
- Фиксация схожести изменения текущей нагрузки с нагрузкой в режиме, который привел к работе ДЗ.
- Отображение информации в графическом виде (реализовано посредством графической библиотеки ZedGraph.dll [2]).

Необходимыми параметрами для мониторинга ДЗ являются мощности P, Q и напряжение U , которые преобразуются в активное R и реактивное X сопротивления в разрабатываемом ПО (рис. 1). Эти параметры могут не приниматься в диспетчерском центре (ДЦ) в силу отсутствия

ЭО устройств телемеханики, способных передать значения P, Q и U в ДЦ. Данная проблема решается только путем модернизации системы сбора и передачи информации ЭО.

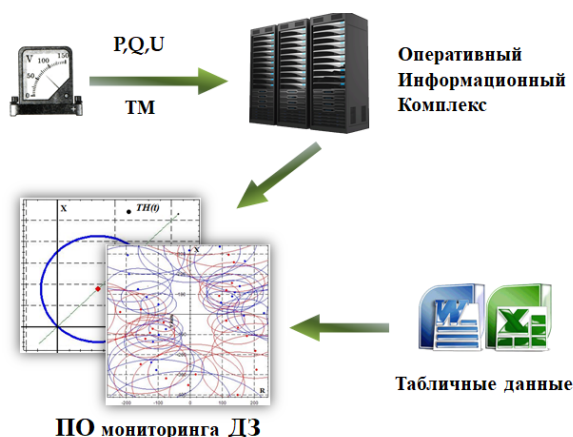


Рис. 1 – ПО мониторинга ДЗ

Каждый параметр ТМ в ОИК подвергается достоверности по описателю качества, времени поступления, физическим пределам, проверяется физическая доступность сервера телемеханики (ring) и устанавливается сессия с его ПО.

Так, например, при выходе (необходимого для мониторинга ДЗ) параметра ТМ за физические пределы (т.е. наблюдается выброс ТМ из-за сбоя системы телемеханики на ЭО) выполняется «замораживание» значения этого параметра до момента восстановления его достоверности. При «замораживании» ПО работает по последнему достоверному значению.

II. ВЫЧИСЛЕНИЕ ТОЧКИ НАГРУЗКИ

Для визуализации точки нагрузки $TН(t) = f\{P(t), Q(t), U(t)\}$ необходимо вычислить активное $R(t)$ и реактивное $X(t)$ сопротивление [3]:

$$R(t) = \frac{3 \cdot P(t) \cdot U^2(t)}{P^2(t) + Q^2(t)}, X(t) = \frac{3 \cdot Q(t) \cdot U^2(t)}{P^2(t) + Q^2(t)}.$$

Расчетная точка $TН(t)$ необходима для определения принадлежности ТН к ОС ДЗ.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ МОНИТОРИНГА ДИСТАНЦИОННОЙ ЗАЩИТЫ: АЛГОРИТМ ВЫЯВЛЕНИЯ СХОЖЕГО РЕЖИМА

Бухаров Д. С., Гусев Р. А.

Служба автоматизированных систем диспетчерского управления, Филиал АО «СО ЕЭС» Иркутское РДУ
Иркутск, Российская Федерация
E-mail: bukharovds@gmail.com

Рассматривается алгоритм фиксации схожести поведения текущей нагрузки с нагрузкой в утяжеленном режиме, повлекшем ложное срабатывание дистанционной защиты. Оперативное выявление схожего режима необходимо для предоставления диспетчеру вспомогательной информации и предотвращения ошибочных действий при изменении режима работы энергосистемы.

ВВЕДЕНИЕ

Наличие в диспетчерском центре необходимой телеметрической информации (ТМ) с присоединений, защищаемых дистанционной защитой (ДЗ), позволяет рассчитать координаты точки нагрузки (ТН) в величинах активного R и реактивного X сопротивления и обеспечить диспетчерский персонал средствами мониторинга утяжеления электроэнергетического режима.

Фиксация приближения ТН к области срабатывания (ОС) ДЗ – основа для мониторинга ДЗ. Такой мониторинг предназначен для общего контроля нагрузочной характеристики и своевременного оповещения диспетчерского персонала о потенциальной работе ДЗ.

При изменении режима работы энергосистемы могут быть предприняты действия, которые впоследствии приведут к работе ДЗ. Информация о таких действиях и их последствиях должна храниться и предоставляться диспетчеру в форме справочных данных.

На основе ретроспективных данных (зафиксированных при режиме, который привел к работе ДЗ) и принимаемой ТМ возможно сопоставить текущее поведение ТН с ранее зафиксированными и проинформировать диспетчерский персонал о возможных последствиях при ведении нормального или аварийного режима.

Для решения данной задачи разработан алгоритм, позволяющий проанализировать текущее поведение нагрузки с нагрузкой при известном неблагоприятном режиме. Алгоритм реализован в рамках программного обеспечения (ПО) для мониторинга ДЗ.

I. ПРИНЦИП ФИКСАЦИИ СХОЖЕСТИ

ДЗ срабатывает при попадании ТН в ОС D (рис. 1, ОС в форме окружности). До сближения с D ТН преодолевает некоторый «предавварийный путь» S .

Линия S – ломанная, соединяющая последовательно точки TH_i , $i = 1, n$. Точка $TH_n \in D$ – точка нагрузки, при которой выполнялась работа ДЗ.

Оповещение о сходстве поведения текущей нагрузки $TH(t)$ с нагрузкой в предаварийном режиме выполняется тогда, когда фиксируется траектория изменения нагрузки S_{TH} близкая к предаварийной S .

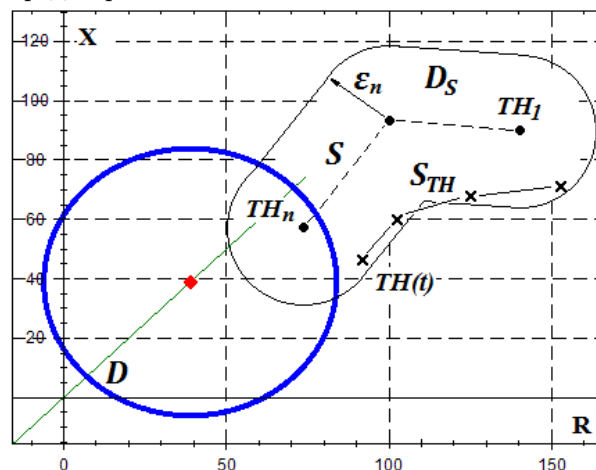


Рис. 1 – Принцип фиксации схожести

Близость S_{TH} к S определяется по вхождению $TH(t)$ в область D_S (рис. 1), охватывающую траекторию S ее эквидистантой [1] на расстоянии радиуса ϵ_n области отстройки от

$$Z_{TH_n} = \sqrt{R_{TH_n}^2 + X_{TH_n}^2},$$

где $\epsilon_n = \mu \cdot Z_{TH_n}$, μ – коэффициент отстройки от нагрузки, $TH_n \in D_S \cap D$.

II. ПОСТРОЕНИЕ ЭКВИДИСТАНТЫ ПРЕДАВАРИЙНОГО ПУТИ

Пример эквидистанты (пунктирная линия) отрезка AB приведен на рис. 2. Отрезок AB опоясывается прямоугольником (зеленый цвет) длиной AB и высотой сторон $2 \cdot \epsilon_n$

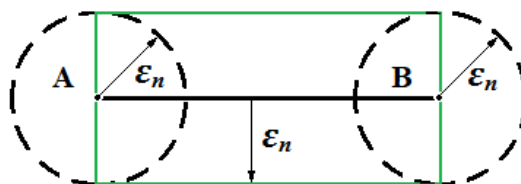


Рис. 2 – Построение эквидистанты

В концах отрезка AB строятся окружности (черная пунктирная линия) радиуса ε_n с центрами в точках A и B .

Предаварийный путь S содержит $n - 1$ отрезков пути, соответственно строится $n - 1$ прямоугольник и n окружностей. Выполняется проверка вхождения $TH(t)$ внутрь каждого прямоугольника и окружности.

Величина n определяется исходя из факта приближения ТН к ОС ДЗ D (рис. 3). Точка TH_1 – первая точка перед входом в желтую область предупредительной сигнализации.

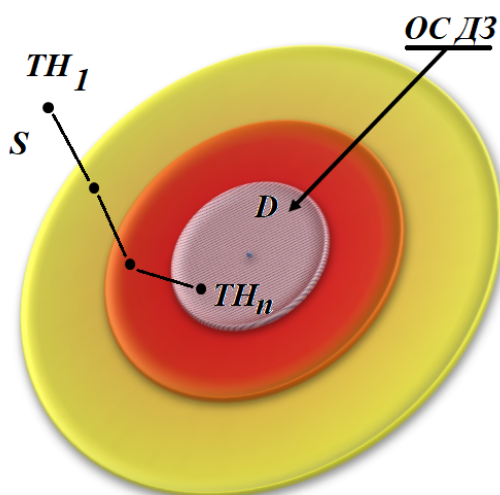


Рис. 3 – Определение предаварийного пути

Проверка вхождения $TH(t)$ внутрь каждого прямоугольника требует применение тригонометрических функций, что с вычислительной точки зрения – трудоемко.

Для снижения нагрузки на центральный процессор используется замена прямоугольников совокупностью окружностей (рис. 4).

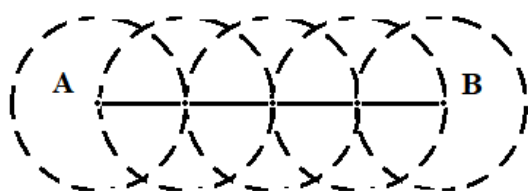


Рис. 4 – Приближенное построение эквидистанты

Отрезок разбивается на равные участки, конец каждого небольшого участка – центр окружности.

III. АЛГОРИТМ

В общем виде алгоритм фиксации схожести S_{TH} и S содержит в себе следующие вычислительные операции:

1. Проверка вхождения точки $TH(t)$ в часть эквидистанты D_S .
2. Фиксация элемента D_S , в который вошла точка $TH(t)$.
3. Определение направления изменения ТН. Это необходимо для исключения дребезга $TH(t)$ возле области D_S и ложного срабатывания сигнализации.
4. Вычисление процентного соотношения количества точек S_{TH} вошедших в область D_S . Если более 40% точек S_{TH} оказались внутри D_S , то срабатывает сигнализация о схожести S_{TH} и S .

Условие активации механизма фиксации схожести – вхождение $TH(t)$ в область D_S и приближение к области D на расстояние желтой области сигнализации приближении ТН к ОС ДЗ (рис. 3).

В оперативной памяти ПО для мониторинга ДЗ хранится последовательность окружностей от TH_1 к TH_n , которая определяет направление изменения предаварийной нагрузки.

Выполнение операции № 4 происходит при вхождении $TH(t)$ в красную область сигнализации о приближении ТН к ОС ДЗ.

IV. ЗАКЛЮЧЕНИЕ

Механизм фиксации схожести поведения текущей нагрузки с нагрузкой в утяжеленном режиме, повлекшем ложное срабатывание ДЗ, позволяет заблаговременно:

- Выявить схожесть режимов.
- Привлечь внимание диспетчера и отразить текущее состояние нагрузочной характеристики на комплексной плоскости.
- Предоставить диспетчеру справочную информацию по стабилизации режима.

Дальнейшее направление модернизации разрабатываемого ПО – передача информации в оперативно-информационный комплекс следующей информации:

- Факт вхождение желтую, красную области и ОС ДЗ(рис. 3).
- Период времени для создания импульс-архива данных.
- Поведение ТН и управляющих воздействиях.

Данная информация необходима для выработки указаний для диспетчеров и проведения противоаварийных тренировок.

СПИСОК ЛИТЕРАТУРЫ

1. Арнольд, В. И. Особенности каустик и волновых фронтов / В. И. Арнольд // М.: ФАЗИС, 1996. – 334 с.

DECISION MAKING BASED ON RELIEF ALGORITHM

German O. V., German J. O., Nasr S.

Information technologies in automatized systems department, The Belarusian State university of informatics
and radioelectronics

Minsk, Republic of belarus

E-mail: {german o, german j.,nasr}ovgerman@tut.by, juliagerman@tut.by, sara.nasrh@gmail.com

An approach is considered to select the best candidate for a job on the basis of curriculum vitae (CV). There are some important criteria to be taken into account: age, education, participating in real projects, availability of published papers or research activities, possessing modern programming languages and technologies and so on. Each criteria has an unknown weight to be used in estimating the integral choice function. From this, a method is required to build such a function and the author's approach to solve the problem is described.

INTRODUCTION

One of the interesting applied problems is automatic CVs processing. The final goal is to select the best candidate for some position(s) from those who sent his/her e-mail. Typically, CV contains a block of private and professional data of the candidate including age, education, participating in real projects, availability of published papers or research activities, possessing modern programming languages and technologies and so on. The problem of selection of the best candidate requires to estimate the integral choice function usually represented in the next form

$$C = \sum w_i \cdot uf_i$$

Here, w_i stands for the weight of criterion i , and uf_i represents the value of utility function of criterion i . The main problem is connected to evaluation of the weighting coefficients w_i . Clearly, the specificity of this process is its extreme indefiniteness. To cope with this indefiniteness we propose to use well-known RELIEF method [1]. The rest of this short report contains the details.

I. THE RELIEF BASED APPROACH

Let us remind the basic principles of RELIEF. The starting point is gathering data based on the accumulated CVs for the past time periods. For each CV it is known the final result, namely, was the applicant taken to the desired position or not. Basing on these information the features of each applicant should be estimated directly from the CV text. Of course, it is a separate problem which requires detail attention which moves it out of the frames of this report. To estimate the feature A weight we select any CVs - CVx from the corpus and evaluate the next expression

$$W[A] = W[A] - (\text{dif}(A,R,H) - \text{dif}(A,R,M))/m$$

Here, $W[A]$ stands for initial value of the weighting coefficient of the feature A , m represents the number of features (that is, the number of criteria). Value of $\text{dif}(A,R,H)$ taken with minus sign defines the difference between the value of the feature A of CVx and the nearest feature A value from the set of declined candidates. Value of $\text{dif}(A,R,M)$ taken with plus sign defines the

difference between the value of the feature A of CVx and the nearest feature A value from the set of accepted candidates. The expression re-evaluation is performed for all CV samples from corpus. To find differences of the feature A values between samples i_1 and i_2 one uses the following expression

$$\text{dif}[A,i_1,i_2] = \text{abs}(\text{value}(A,i_1) - \text{value}(A,i_2)) / (\text{max}(A) - \text{min}(A))$$

As a final result one gets the weighting coefficients $W[A]$ for each feature A . These coefficients now can be used in integral choice function. The integral choice function is used for each sample of CV in the corpus. Judging by its values all samples can be divided into three clusters: (1) those, which be accepted; (2) those, which be declined and (3) those in the "grey"zone (may be or may be not accepted). The grey zone is of evident interest. We suggest to use a kind of k -mean technique for sample falling into the "grey"zone. Accordingly to these technique one finds the nearest values of the integral choice function, corresponding to the samples from "grey"zone with respect to the value of integral choice function of the newly incoming CV. Then one define how many from these k -mean values correspond to accepted CVs, and how many to declined CVs. Then it remains to define which number is greater and classify the newly incoming CV respectively.

II. THE CV-AUTOMATIZATION PROCESSING

The entire process is divided into four stages [2]. The first stage is to transform Word document or a pdf-file with CV to a plain text. This can be done with a Tika system which extracts a raw text and deletes unnecessary control information such as colors, fonts, and the like. The next step is to get keywords of the text. The idea is to consider practically all text words as keywords due to not big size of CV. It is required that each sentence in CV contains at minimum one keyword. Besides, this simplifies the algorithm as it not requires to define each keyword score and test that each sentence is covered at least by one keyword. The prepositions, conjunctions, pronouns, auxiliary and modal verbs (such as can, have, may etc.) are excluded. The keywords are selected in such a way that the similar

words are identified as the same. For instance, *programming* and *programmer* are considered as one keyword. By thus, each keyword labels one or more sentences. To realize the keywords selection we use the Dice metrics (measure) given by the formula

$$P = 2 * |X \cap Y| / (|X| + |Y|),$$

where $|X|$ ($|Y|$) stands for the set X (Y) size. For example, for $X = programming$ and $Y = programmer$ one has $P = 2 * 8 / (11 + 10) = 0.8$. We adopt the rule accordingly to which two words are considered alike if the Dice measure is 0.5 or greater. From this, in the example above the words *programming* and *programmer* are considered a one keyword. It should be noticed here that fuzzy sequential text searching methods represent evident practical interest especially if they are applied on servers with very large amount of CVs to be processed. The next stage consists of building two directories. Each directory represents a collection of the pairs $\langle key, value \rangle$. The first directory contains the pairs of items and each pair represents a record with key standing for a keyword and a value representing the set of sentences numbers labelled with the keyword key. The second dictionary consists of the pairs representing the sentences numbers and their texts. Now let us explain how to extract an answer to the question represented by a set of words (some of words are keywords and some not (these latter words do not belong to CV text)). For each keyword k_i in the question the set of sentences numbers $N_i = n_{i1}, n_{i2}, \dots, n_{iz}$ is defined. Then for all N_i the most frequently encountered number(s) n_w is (are) defined. This number n_w defines a sentence to be displayed as an answer. If there are more than one candidate to be an answer then all candidates are selected. It may be the case when CV contains quite a big blocks of a semantically connected information. This situation is somewhat wider than a typical one. Each semantically united block may consist of one or more sentences. The block specifies some concept, that is, it may be considered as a semantical whole. Practically, we associate such blocks with paragraphs (indentations) or the text segments separated from the others with empty lines or by spaces. As before, we should define the keywords for the sentences: $k_1, k_2, k_3, \dots, k_z$. We introduce the third dictionary with the keys representing the sentences numbers and values standing for the text blocks. Thus, we got the chain: keyword \rightarrow sentence(s) \rightarrow text block(s). The searching procedure remains the same. The keywords from the query are used to find the sentences numbers and the corresponding text blocks. The text block which is referred to by the majority of keywords is then selected. The

problem of blocks definition has a special scientific meaning. The simplest case consists in dividing the text by paragraphs separated with indentions or empty lines. Evidently, each block should contain the connected notions (keywords). One says that related keywords form a cluster. So, one needs to build a number of clusters, combining different but semantically related keywords. The first problem with clusters is their number. Obviously, one needs to apply the existing clustering approaches. Then, after the necessary text block is selected, the corresponding criteria associated with this query should be evaluated. The list of queries is predefined. Each query is linked to some criteria. For example, let the query be "participation in real projects and programs". The answer "Not found" is considered as lowest level of the criteria. The value of criteria is estimated accordingly to the number of projects and programs an applicant was engaged in. Other criteria may require even a 0-1 answer, as for instance "are you married?". The list of criteria consists of the next ones: (1) speciality in programming/mathematics/informatics; (2) participation in real projects/programs; (2) ability and experience in writing programs in c, java, python etc. languages; (3) family status (married/unmarried); (4) possessing printed publications (papers,books); (5) speciality by education; (6) possessing certificates in programming/system adminestering; (6) knowing foreign languages; (7) studying at additional professional cources. For these criteria a list of predefined questions is made. The questions are processed in the way shortly outlined above. As a result, the corresponding criteria are evaluated with the help of decision making technique and an integral applicant grade is made to be used in final selection.

III. CONCLUSION

The specificity of the approach suggested in this report consists in usage of RELIEF technique to define the coefficients of the integral choice function. By means of the defined choice function we divide the learning set of CVs into 3 clusters with one of them called "grey"zone. Basing on the ideas of this report we intend to realize CV-processing system in practice.

1. Gao, L. Research and application of data mining feature selection based on Relief algorithm / L. Gao, T. Li, L. Yao, F. Wen // Journal of Software.2014. vol.9, 2, p.p.515-522.
2. German, Yu. Information extraction method from resume / Yu. O. German, O. V. German, S. Nasr // Proceedings of BSTU Scientific journal. (Minsk, Belarus).2019, 1(218), p.p.64-69.

СИСТЕМА СОВМЕСТНОЙ РАБОТЫ С РЕЗУЛЬТАТАМИ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Гончарик И. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: amsterget@gmail.com, gurinovich@bsuir.by

В статье исследуются проблемы актуальности применения автоматизированного тестирования и вопросы, которые возникают в ходе использования данного подхода к тестированию программных продуктов. Производится описание системы, предлагающей инструменты для их решения.

ВВЕДЕНИЕ

Автоматизированное тестирование программного обеспечения стало неотъемлемой частью процесса разработки ПО. Это является одним из мировых трендов для компаний, занимающихся ИТ-инженерией. Использование автоматизированного тестирования позволяет оптимизировать процесс разработки программного продукта основные цели данного подхода представлены на рисунке 1.



Рис. 1 – Цели автоматизированного тестирования

Преимущества такого подхода к тестированию программных продуктов бесспорны, однако, существует ряд недостатков. Главная проблема заключается в отсутствии централизованного управления результатами автотестов. Это влечёт за собой невозможность получения информации о состоянии автоматизации в целом, например, менеджерами проекта и заинтересованными лицами.

I. АКТУАЛЬНОСТЬ СИСТЕМЫ

Компании, использующие автоматизированное тестирование в процессе разработки ПО, вынуждены мириться с недостатками данного подхода. Проблемы связаны, в основном, со структуризацией полученных результатов автотестов, приведением результатов тестирования различных частей программного продукта к одному виду и организацией простого доступа к этим данным любому члену команды.

Автоматизация тестирования предполагает написание программного кода, тестирующего работу другого кода. В общем случае, различные части программного продукта реализуются с использованием разных технологий и языков программирования, что влечёт за собой и многообразие платформ для их тестирования. Таким образом получение актуальных данных о состоянии автоматизации и, следовательно, работоспособности функционала ПО, является задачей, требующей непосредственного контакта управляющего звена с группами тестировщиков-автоматизаторов. Это создаёт дополнительные временные затраты на обсуждение и визуализацию результатов тестирования.

В связи с этим указанные недостатки автоматизированного тестирования дают толчок к созданию инструмента агрегации и визуализации результатов автотестов, а также совместной работы с ними.

II. ПОСТАНОВКА ЗАДАЧИ

Системе организации совместной работы с результатами автоматизированного тестирования необходима поддержка следующих возможностей:

- Универсальность. Представляет собой возможность системы работать с результатами автотестов независимо от платформы тестирования.
- Визуализация данных. Система включает в себя возможность графического отображения результатов тестирования в виде графиков и диаграмм.
- Совместный доступ. Предполагает собой возможность организации совместной работы с результатами тестирования в системе с поддержкой разграничения прав доступа к данным.
- Гибкость. Включает в себя возможность продукта по интеграции со сторонними сервисами, включая существующие системы отслеживания дефектов и сервера электронной почты.

Архитектура системы представляет собой сетевую клиент-серверную архитектуру. Совокупность взаимодействующих между собой посредством сетевых протоколов сервисов серверная часть. Сторона клиента представлена группой посредников, так называемых «агентов», являющихся связующим звеном между платформами для тестирования и серверной частью приложения. Для визуализации данных и упрощения работы с ними также реализуется сервис пользовательского интерфейса. Взаимодействие между «агентами» и серверной частью системы осуществляется через API (Application programming interface). Структурная схема системы представлена на рисунке 2.

Для реализации серверной части системы используется объектно-ориентированный язык программирования Java совместно со Spring Framework, система управления базами данных PostgreSQL. Все используемые технологии являются программным обеспечением с открытыми исходным кодом. Это даёт доступ к большому количеству инструментов и сообществу, что способствует успешному процессу разработки.

Поддержка необходимой платформы тестирования требует реализации соответствующего «агента», «понимающего» язык платформы и «умеющего» работать с API сервера.

В работе показано, что автоматизируя тестирование работоспособности на соответствие требованиям запроецированного функционала программного продукта, можно существенно уменьшить риски получения некачественного продукта и оптимизировать затраты на разработку. При переводе описания тест-кейсов в код, тестировщик-автоматизатор получает задокументированную часть функционала системы и возможность в любой момент проверить это соответствие. Для получения полного представления о готовности программного продукта, необходимо систематизировать результаты тестирования всего приложения и произвести их визуализацию. Система совместной работы с результатами автоматизированного тестирования призвана помочь в решении данных задач.

СПИСОК ЛИТЕРАТУРЫ

1. Автоматизированное тестирование программного обеспечения / Э. Дастин, Дж. Рэшка, Д. Пол // Издательство: Лори, 2003. – 580 с.
2. Experiences of Test Automation: Case Studies of Software Test Automation / D. Graham, M. Fewster // Addison-Wesley, 2012. – 607 с.
3. Problems with Test Automation and Modern QA [Electronic resource] / A. Ghahrai. – 2018. – Mode of access: <https://www.testingexcellence.com/problems-test-automation-modern-qa/>.

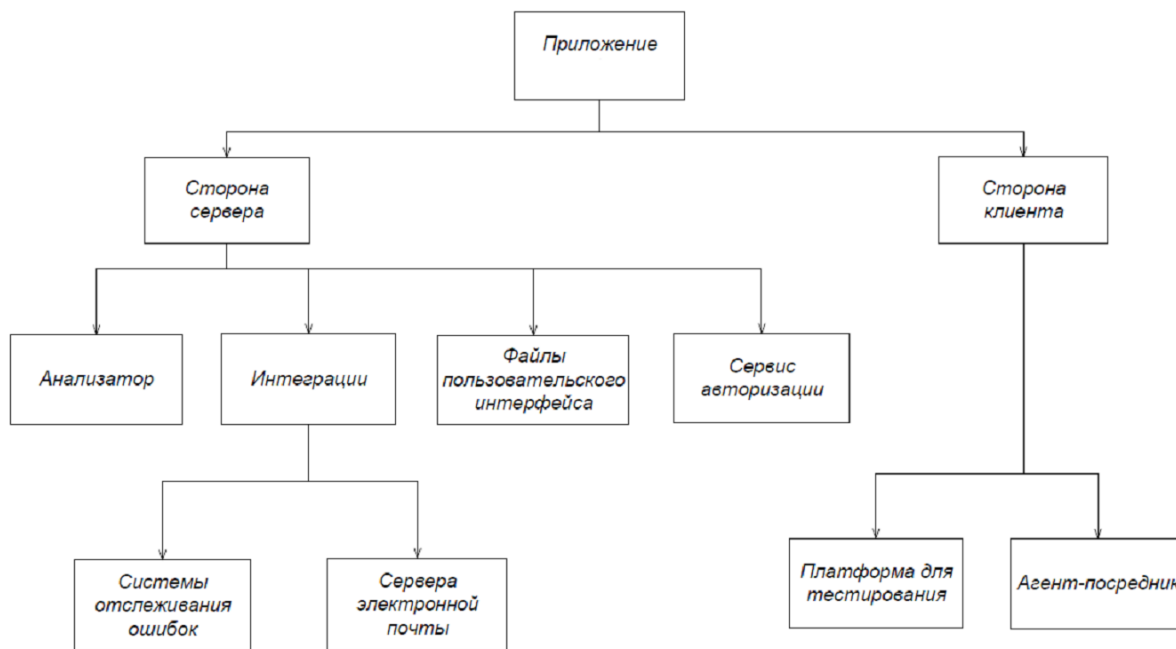


Рис. 2 – Структурная схема системы

ПРИНЦИПЫ ИДЕМПОТЕНТНОСТИ И КОНВЕРГЕНТНОСТИ В ОБЛАСТИ МИГРАЦИИ ДАННЫХ

Данильчик В. В., Прытков В. А.

Кафедра электронных вычислительных машин, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: danilchican@mail.ru, prytkov@bsuir.by

Миграция схем данных является неотъемлемой задачей при переходе от одной структуры схем данных к другой. Сложность миграции данных заключается в наличии большой разницы между текущей структурой схемы данных и новой, а также достаточно большую сложность составляет наличие существующих зависимостей между объектами. В данной статье рассмотрены принципы, которые помогают определить алгоритм действий для миграции схемы данных и ее сущностей.

Актуальность анализа способов миграции схем данных обусловлена необходимостью постоянной поддержки структуры базы данных и ее последующей оптимизации при выходе новой версии программного продукта. В настоящее время процессу миграции данных часто сопутствуют проблемы их дублирования и возникновения ошибок, связанных с изменением структуры данных.

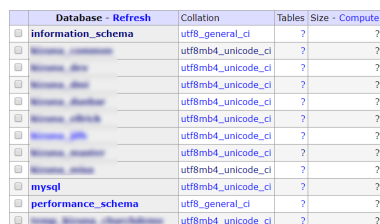
У большинства подходов есть общий принцип: им необходимо основание — некоторое эталонное состояние базы данных (БД), от которого можно отталкиваться. Что из себя представляет основание — это дампы структуры базы данных для версии, которая принята за базовую. Имея основание, впоследствии всегда можно будет создать БД с нуля. Чтобы получить БД со структурой самой последней версии, нужно применить к этой базе данных все миграции, созданные в процессе разработки.

При разработке программного продукта и, соответственно, написании кода разработчики пользуются системами контроля версий, такими как Git или Subversion. Данные системы позволяют вести совместный процесс разработки несколькими сотрудниками, хранить только разницу между версиями кода, отслеживать и, при необходимости, возвращаться к конкретной версии исходного кода. В то же время применение систем контроля версий при работе со структурами схем данных довольно ограничено.

Также, одной из причин миграции схем данных является наличие каких-либо аномалий либо увеличение нагрузки при анализе работы программного продукта с базой данных, т.к. быстрота работы БД напрямую зависит от ее структуры и выполненной оптимизации. В этом случае при миграции схемы данных, могут быть добавлены новые индексы на определенные столбцы таблиц.

Для того, чтобы повысить производительность системы, нужно проанализировать производительность базы данных. Для этого существует Схема Производительности (см. рис.1). Она фокусируется прежде всего на данных о

производительности [1], следит за развитием событий сервера. «Событие» - что-либо, что сервер делает, который занимает время. Собранные события сохранены в таблицах в базе данных. Эти таблицы могут быть запрошены, используя «SELECT» операторы как другие таблицы.



Database - Refresh	Collation	Tables	Size	Compute
information_schema	utf8_general_ci	?	?	?
mysql	utf8mb4_unicode_ci	?	?	?
performance_schema	utf8mb4_unicode_ci	?	?	?
sys	utf8mb4_unicode_ci	?	?	?
test	utf8mb4_unicode_ci	?	?	?
test2	utf8mb4_unicode_ci	?	?	?
test3	utf8mb4_unicode_ci	?	?	?
test4	utf8mb4_unicode_ci	?	?	?
test5	utf8mb4_unicode_ci	?	?	?
test6	utf8mb4_unicode_ci	?	?	?
test7	utf8mb4_unicode_ci	?	?	?
test8	utf8mb4_unicode_ci	?	?	?
test9	utf8mb4_unicode_ci	?	?	?
test10	utf8mb4_unicode_ci	?	?	?
test11	utf8mb4_unicode_ci	?	?	?
test12	utf8mb4_unicode_ci	?	?	?
test13	utf8mb4_unicode_ci	?	?	?
test14	utf8mb4_unicode_ci	?	?	?
test15	utf8mb4_unicode_ci	?	?	?
test16	utf8mb4_unicode_ci	?	?	?
test17	utf8mb4_unicode_ci	?	?	?
test18	utf8mb4_unicode_ci	?	?	?
test19	utf8mb4_unicode_ci	?	?	?
test20	utf8mb4_unicode_ci	?	?	?
test21	utf8mb4_unicode_ci	?	?	?
test22	utf8mb4_unicode_ci	?	?	?
test23	utf8mb4_unicode_ci	?	?	?
test24	utf8mb4_unicode_ci	?	?	?
test25	utf8mb4_unicode_ci	?	?	?
test26	utf8mb4_unicode_ci	?	?	?
test27	utf8mb4_unicode_ci	?	?	?
test28	utf8mb4_unicode_ci	?	?	?
test29	utf8mb4_unicode_ci	?	?	?
test30	utf8mb4_unicode_ci	?	?	?
test31	utf8mb4_unicode_ci	?	?	?
test32	utf8mb4_unicode_ci	?	?	?
test33	utf8mb4_unicode_ci	?	?	?
test34	utf8mb4_unicode_ci	?	?	?
test35	utf8mb4_unicode_ci	?	?	?
test36	utf8mb4_unicode_ci	?	?	?
test37	utf8mb4_unicode_ci	?	?	?
test38	utf8mb4_unicode_ci	?	?	?
test39	utf8mb4_unicode_ci	?	?	?
test40	utf8mb4_unicode_ci	?	?	?
test41	utf8mb4_unicode_ci	?	?	?
test42	utf8mb4_unicode_ci	?	?	?
test43	utf8mb4_unicode_ci	?	?	?
test44	utf8mb4_unicode_ci	?	?	?
test45	utf8mb4_unicode_ci	?	?	?
test46	utf8mb4_unicode_ci	?	?	?
test47	utf8mb4_unicode_ci	?	?	?
test48	utf8mb4_unicode_ci	?	?	?
test49	utf8mb4_unicode_ci	?	?	?
test50	utf8mb4_unicode_ci	?	?	?

Рис. 1 – Схема Производительности

Схема Производительности предназначена, чтобы обеспечить доступ к полезной информации о выполнении сервера, оказывая минимальное влияние на производительность сервера. Реализация следует за этими целями проекта. Контроль сервера происходит непрерывно и незаметно с очень небольшими издержками. Активирование Схемы Производительности не делает сервер неприменимым

В результате в процессе развития программного продукта отследить изменения структуры данных весьма проблематично ввиду того, что DDL, язык определения схем данных, не предусматривает документирование изменения структуры на каждом шаге [2].

Решением данной задачи является создание в каталоге проекта пронумерованных скриптов модификации структуры базы данных при этом, не прибегая к использованию дополнительных инструментов. В таком случае все изменения будут накапливаться в проекте при условии соблюдения заранее определенных правил участниками проекта.

Одним из наиболее современных и практичных способов контроля структуры схем данных является использование дополнительного инструмента Liquibase [3]. Основной задачей системы является контроль номера последнего выполненного скрипта изменения структуры дан-

ных и обеспечение единовременного линейного выполнения скриптов по изменению структуры данных. Однако, использование инструментов, в основе которых лежит принцип накопления change-скриптов (см. рис.2), не позволяет решить задачу так же легко, как написать программный код приложения. Главным минусом данного подхода является необходимость хранения огромного количества неактуальных change-скриптов, что затрудняет получение представления о фактически используемой структуре схемы данных. У liquibase есть еще один существенный недостаток — автоматическое возвращение изменений работает лишь только в том случае, если миграция описана в виде файла XML. Однако, Liquibase позволяет писать миграции и на SQL, но с одной ремаркой — автоматическое возвращение теперь не работает и секцию, которая выполняется при отмене миграции нужно писать вручную.

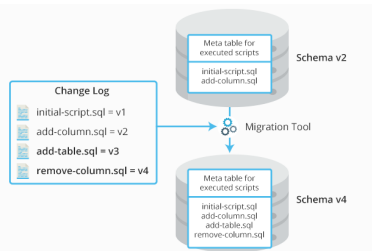


Рис. 2 – Change-скрипты

Время от времени, при обновлении версии базы данных, нужно обновлять не только структуру БД, но и хранящиеся в ней данные. В качестве примера можно привести перенос данных из таблицы со старой структурой в новые таблицы — в целях нормализации. Поскольку данные уже существуют и используются, недостаточно просто создать новые таблицы и удалить старые, нужно еще и перенести имеющиеся данные.

Есть несколько решений этой задачи:

- хранить изменения данных путем накопления миграционных скриптов;
- нигде не хранить запросы-изменения данных, а в процессе изменения структуры схемы данных подготовить данные для импорта, которые будут соответствовать новой схеме.

Для решения поставленных задач используются такие принципы, как конвергентность и идемпотентность. Соблюдение приведенных принципов в разработанном модуле миграции

позволяет наиболее эффективно контролировать состояние схемы данных и не допускать повторного выполнения скрипта миграции данных при его успешном запуске ранее.

Наиболее оптимальным решением задачи миграции схем данных и является следование данным принципам: идемпотентности и конвергентности [4]. Отличительной особенностью такого подхода является то, что скрипт, который удовлетворяет условиям принципов, описывает состояние, к которому объект должен быть приведен, а не действия, которые требуется произвести над ним.

Идемпотентность обеспечивает отсутствие изменений объекта при выполнении скрипта повторно в случае, если скрипт был выполнен успешно при первом его запуске. То есть основная идея этого подхода — написание миграционных файлов таким образом, чтобы их можно было выполнить больше одного раза. При первой попытке выполнить любую из SQL-команд, изменения будут применены; при всех последующих попытках ничего не произойдет.

Соблюдение данного принципа позволяет избежать дублирования и искажения данных. Однако, в случае невыполнения скрипта или неудачного завершения его выполнения, система, при повторном выполнении данного скрипта, будет пытаться прийти к тому состоянию, которое было задано в качестве желаемого. Такое поведение системы обусловлено принципом конвергентности, который направлен на приведение системы в нужное состояние при каждом следующем запуске одного и того же скрипта без выполнения операций, успешно выполненных ранее. Т.е. изменения, не выполненные при одной попытке, система будет пытаться выполнить еще раз при повторном запуске.

1. Performance Schema Quick Start [Электронный ресурс] - Режим доступа: <https://dev.mysql.com/doc/refman/5.6/en/performance-schema-quick-start.html>
2. Клеппман. М. Высоконагруженные приложения. Программирование, масштабирование, поддержка/М. Клеппман // СПб: Питер, 2018.
3. Версионирование структуры БД с помощью Liquibase [Электронный ресурс] - Режим доступа: <http://easy-code.ru/lesson/database-versioning-liquibase>
4. Database Migration: What It Is and How to Do It [Электронный ресурс] - Режим доступа: <https://rollout.io/blog/database-migration>

КЛАССИФИКАЦИЯ ДЕФЕКТОВ ПОДШИПНИКА С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

Денисов В. А., Федоров П. А.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: {q1munys, pavl.fedorov}@gmail.com

Вибрационный анализ является жизненно важной задачей в области контроля состояния и диагностики неисправностей различного промышленного оборудования, таких как индукционные машины. В данной статье рассматривается использование методов интеллектуального анализа данных для классификации неисправностей оборудования.

ВВЕДЕНИЕ

Вибрационный анализ - это процесс поиска аномалий и отслеживания изменений по установленной вибрационной сигнатуре системы. Вибрация любого объекта характеризуется изменениями амплитуды, интенсивности и частоты. Они могут соотноситься с некоторыми физическими явлениями, что позволяет использовать данные о вибрации для получения представления о состоянии оборудования.

Вибрационный анализ в основном сфокусирован на частотном анализе. Задача классификации типа возникающих отказов машины по временному ряду является нетривиальной, поэтому временные сигналы (смещение, скорость или ускорение) преобразуются с использованием преобразования Фурье в сумму синусоидальных волн разных частот, где каждый представляет частотную составляющую. Частотный анализ используется для обнаружения неисправностей оборудования, таких как дисбаланс, перекос, изогнутый вал, механическая ослабленность, неисправность подшипника, неисправность коробки передач и т. д.

В индукционной машине подшипники часто подвергаются воздействию несбалансированной силы, создаваемой в машине. Это приводит к различным типам неисправностей в подшипниках и одновременно к отказу вращающегося механизма. А неисправности подшипников приводят к различным проблемам, которые являются причиной неправильного функционирования вращающегося оборудования.

Дефекты в оборудовании можно представить в виде простого гармонического движения, которое выражается в изменении амплитуды вибрационного сигнала. Техническое обслуживание такого механизма требует мониторинга в течение некоторого промежутка времени. Различные параметры влияют на возникновение неисправностей в оборудовании, которые могут быть связаны с циклической нагрузкой станка, ошибкой при установке подшипников качения во вращающемся оборудовании между валами, и расход смазки вращательных элементов оборудования. Все эти упомянутые параметры приво-

дят к выходу из строя подшипника. Факторы, который приводят к дефекту подшипника, можно разделить на две категории: распределенные и локализованные дефекты. Эти дефекты классифицируются по частотным характеристикам в терминах спектра колебаний.

МЕТОДЫ ОБНАРУЖЕНИЯ НЕИСПРАВНОСТЕЙ

Методы обнаружения неисправностей можно разделить на три группы: основанные на сигналах, основанные на знаниях и основанные на моделях методы. Методы обнаружения неисправностей на основе сигналов состоит в анализе компонентов спектра измеряемых сигналов. Методы, основанные на моделях, требуют точной модели системы, для имитации реального поведения процесса, поэтому доля использования данного метода составляет около 10% обнаружения неисправностей. Методы, основанные на знаниях, используют методы, такие как нечеткие системы и нейронные сети для выявления неисправностей в оборудовании.

Подшипниковые дефекты можно разделить на точечные дефекты и шероховатости. Точечные дефекты обычно создаются в автономном режиме в лаборатории или мастерской, например, путем сверления отверстия в любой части подшипника. Шероховатость чаще всего возникает при работе в оперативном режиме из-за деградации поверхностей подшипников. Состояние подшипника можно контролировать с помощью вибрации оборудования. Это связано с тем, что повреждения подшипников, будь то точечные дефекты или общая шероховатость, как правило, приводят к последовательным и периодическим импульсным составляющим вибрации машины, вызванным пропуском шарикоподшипника через дефектные точки. Период этих членов можно рассчитать, зная скорость вращения, положение разломов и размеры подшипников. Простейшим методом анализа частотной области, используемым для обнаружения повреждений подшипников, является быстрое преобразование Фурье (БПФ). Воздействие вибрации, вызванной повреждением подшипника, имеет относительно низкую энергию и часто сопровож-

дается шумом и вибрацией высокой энергии. Поэтому трудно идентифицировать дефект подшипника в спектрах, используя обычные методы. Чтобы преодолеть проблемы БПФ, были разработаны различные передовые методы обработки сигналов для обнаружения неисправностей подшипников, такие как кратковременное преобразование Фурье (STFT), вейвлет-преобразование (WT), вейвлет-пакетное преобразование (WPT) и векторный подход Парка.

В 1991 г. нейронная сеть была впервые предложена для диагностики неисправностей однофазных двигателей. Нейронная сеть была использована для диагностики различных типов внешних неисправностей в трехфазных индукционных двигателях. Была разработана нейронная сеть с обратным распространением и обратной связью, предназначенная для обнаружения несущих, сломанных стержней, обмоток статора и несбалансированных повреждений ротора. Хотя эти исследования достигли некоторого успеха, для обучения нейронной сети использовались особенности как во временной, так и в частотной областях со сложным расчетом. Некоторые из них использовали WPT для предварительной обработки вибрационных сигналов перед тренировкой сети.

Для обнаружения неисправностей используется трехэтапный алгоритм. На первом этапе нейронная сеть оценивает скорость вращения вала, а динамическая рекуррентная нейронная сеть используется в качестве пилотной модели для оценки остаточного значения текущего сигнала. Заменяемый результат обрабатывается с помощью WPT для обнаружения механических неисправностей. Однако такие методы требуют много вычислительных операций из-за использования WPT. В последние годы стали активно применяться методы распознавания на основе ANN.

В предлагаемом способе, на первом этапе исправные компоненты вибрационного сигнала оцениваются интеллектуальным фильтром, который разработан на основе нейронной сети. Отметим, что эта сеть обучается при нормальных условиях. Поэтому, когда измеренный вибрационный сигнал проходит через этот фильтр, компоненты, которые существующие в исправ-

ном состоянии удаляются, а выходные данные фильтра содержат компоненты, относящиеся к неисправным условиям. На следующем этапе характеристики отфильтрованного по промежутку времени сигнала извлекаются для использования в качестве входных данных классификатора для того, чтобы различать локализацию отказов. Следует отметить, что этот подход не требует каких-либо параметров подшипников для расчета частот дефектов, так как он использует только функции временной области для обнаружения неисправностей.

ЗАКЛЮЧЕНИЕ

В данной статье рассматривается алгоритм обнаружения и классификации дефектов, состоящий из двух этапов. На первом этапе фильтр нейронной сети пропускает компонент несоответствующего дефекта дискретизированного сигнала. Этот фильтр называется удалением не несущего компонента неисправности (RNFC). На следующем этапе функции по временному промежутку извлекаются из выходного сигнала фильтра RNFC и применяются к другой нейронной сети в форме прототипов для обнаружения неисправностей с использованием метода распознавания антипаттерна. Предложенный метод может применяться для выявления других типов неисправностей индукционных двигателей. Например, предлагаемый фильтр RNFC может быть использован для обнаружения поломки ротора или повреждения статора.

СПИСОК ЛИТЕРАТУРЫ

1. Solution to a general forced nonlinear oscillations problem. Multiple time scales harmonic balance. / J. J. Wu and L. C. Chien // *Journal of Sound and Vibration*. – 2015. – P. 185(2), 247-264
2. Incremental harmonic balance method with multiple time scales for aperiodic vibration of nonlinear systems. *Journal of applied Mechanics* / S. L. Lau, Y. K. Cheung, S. Y. Wu // *ASME* – 2014. – P. 50(4), 871-876
3. A modified Lindstedt-Poincare method for a strongly nonlinear two-degree-of-freedom system. / S. H. Chen, Y. K. Cheung // *Journal of Sound and Vibration*. – 2015. P 193(4), 751-762
4. Математические основы машинного обучения и прогнозирования / Вьюгин В. В. // ДМК Пресс – 2018 – С. 55-106

ЛОКАЛЬНО-ЧУВСТВИТЕЛЬНОЕ ХЕШИРОВАНИЕ ДЛЯ СЕГМЕНТАЦИИ РЫНКА

Дрозд П. С.

Факультет радиофизики и компьютерных технологий Белорусского государственного университета
Минск, Республика Беларусь
E-mail: drozdps@gmail.com

В этой работе предлагается новый метод сегментации рынка путем кластеризации транзакционных данных клиентов, пользователей или покупателей в любой компании. Данный способ основан на алгоритме локально-чувствительного хеширования, который увеличивает эффективность и быстродействие последующей иерархической кластеризации в несколько раз. Показана высокая устойчивость подхода, алгоритм реализован в системе с использованием фреймворка Apache Spark.

ВВЕДЕНИЕ

В теории маркетинга существует понятие сегментации рынка, которое подразумевает разбиение клиентов или потенциальных клиентов на различные осмысленные группы. Решение этой задачи позволяет предприятию оптимальнее таргетировать своё предложение на определённую группу (или сегмент) потребителей. В рамках данной работы исследуются методы сегментации рынка с помощью кластерного анализа. Использование кластеризации позволяет создать и проверить гипотезы о существовании в исследуемой совокупности потребителей однородных групп. Для того чтобы добиться улучшения устойчивости и производительности кластеризации, был использован оригинальный подход на базе комбинации алгоритмов Bisecting Kmeans и локально-чувствительного хеширования. Для проверки предложенного подхода мы использовали открытый набор данных «Ta-Feng» компании ACM RecSys, который содержит информацию о покупках различных товаров, совершённых более чем 32 тысячами уникальных клиентов. Всего в наборе имеется 817741 запись, каждая из которых описывает совершённую покупателем транзакцию с помощью 9 характеристик (дата проведения платежа, тип товара, сумма транзакции и т.д.).

I. ЛОКАЛЬНО-ЧУВСТВИТЕЛЬНОЕ ХЕШИРОВАНИЕ

Итак, после предварительных очистки, стандартизации и визуализации данных, было предложено использовать алгоритм локально-чувствительного хеширования (LSH - от англ. Locality-Sensitive Hashing), который традиционно применяется для решения задачи поиска ближайшего соседа[2]. Этот алгоритм позволяет быстро найти «похожие» точки в многомерном пространстве признаков, причем признаки могут быть представлены в различных типах шкал, а записи данных - иметь «выбросы» и/или пропуски. Специфика LSH заключается в том, что этот алгоритм оперирует со специальным набором «плохих» хеш-функций, которые, в

отличии от обычных хеш-функций, должны генерировать коллизии на схожих образах. Таким образом, соседние точки скорее всего попадут в одну хеш-корзину. Пусть $\mathbf{x} = (x_1, x_2, \dots, x_p)^T$ - точка в исходном p -мерном пространстве признаков, которая в рассматриваемом случае описывает одну транзакцию, совершённую одним клиентом/пользователем/покупателем. Зафиксируем максимальное значение всех координат этой точки: $C = \sup\{x_1, x_2, \dots, x_p\} + \epsilon, \epsilon \geq 0$. После этого преобразуем \mathbf{x} в новый вектор размерности Cp по следующему правилу:

$$\begin{cases} \mathbf{v}(\mathbf{x}) = \Psi_C(x_1)\Psi_C(x_2)\dots\Psi_C(x_p) \\ \Psi_C(x_i) = \underbrace{11\dots1}_{x_i}\underbrace{00\dots00}_{C-x_i}, \forall i \leq p \end{cases}$$

Функция $\Psi_C(x_i)$ переводит значение i -ой компоненты \mathbf{x} в последовательность из x_i единиц, за которыми следуют $C - x_i$ нулей. Например, если $\mathbf{x} = (3, 4)^T$ и $C = 5$, то $\mathbf{v}(\mathbf{x}) = (1110011110)^T$. Хеш-функция в алгоритме LSH вычисляет своё значение для вектора \mathbf{x} путём конкатенации k битов (параметр локально-чувствительного хеширования) из $\mathbf{v}(\mathbf{x})$, порядковые индексы которых содержатся в предварительно сгенерированном множестве Υ , содержащем k случайных целочисленных элементов из $\{1, 2, 3, \dots, Cp\}$. На практике генерируется l множеств $\{\Upsilon_1, \Upsilon_2, \dots, \Upsilon_l\}$ и соответственно l хеш-функций, каждая из которых вычисляет своё значение для каждого вектора исходного набора.

II. BIsecting K-MEANS

На предыдущем этапе (LSH) мы выбрали хеш-функции таким образом, чтобы их можно было определить в метрическом пространстве, так как для любого алгоритма иерархической кластеризации необходимо вычислять попарные расстояния между объектами. После этапа LSH все данные оказались разбиты на домены, объединяющие схожие транзакции, и каждому домену соответствовало какое-то значение хеш-функции, то есть если изначально нам необходимо было обрабатывать сотни тысяч объектов в исходном многомерном пространстве признаков, то уже после LSH можно работать с

тысячами или сотнями значений коллизионных хеш-функций. Кроме того, в некоторых случаях размерность значений хеш-функций можно было снизить по сравнению с размерностью исходных данных [2]. Для непосредственного кластерного анализа этих значений был применён алгоритм Bisecting K-means - высокоэффективная иерархическая версия K-means. Этот алгоритм является дивизионным, так как предполагает, что все точки изначально принадлежат одному глобальному кластеру. На каждой итерации он делит текущий кластер на два дочерних с помощью обычного K-means с фиксированным параметром $k = 2$. В некоторых случаях [3], Bisecting K-means на порядки производительнее K-means.

III. СИСТЕМА ДЛЯ СЕГМЕНТАЦИИ РЫНКА

На рисунке 2 представлена use-case диаграмма системы сегментации рынка, которая основана на предложенном методе. Для распараллеливания вычислительных задач система использует фреймворк Apache Spark.



Рис. 1 – Функциональная схема системы

IV. РЕЗУЛЬТАТЫ

После обработки системой набора из Ta-Feng было получено 6 осмысленных кластеров покупателей, которые представляли различные сегменты рынка (пенсионеры, VIP-клиенты и т.д.), а также также представлены ориентировочные советы (marketing insights) по оптимизации маркетинга в отношении этих групп (например, для максимизации прибыли следует предоставлять скидку пенсионерам в выходные и предпраздничные дни, так как средняя корзина в этом случае становится больше).

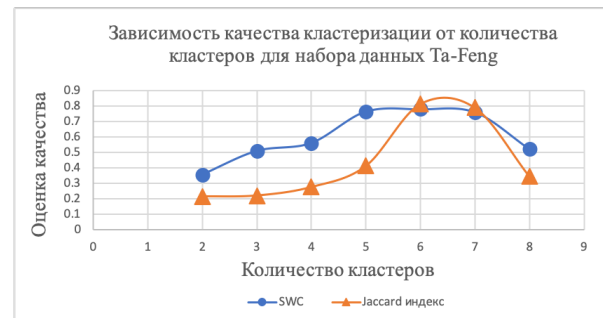


Рис. 2 – Анализ качества сегментации Ta-Feng

Ниже представлена итоговая схема работы предложенного подхода (коллизионное хеширование в LSH и последующая иерархическая кластеризация).

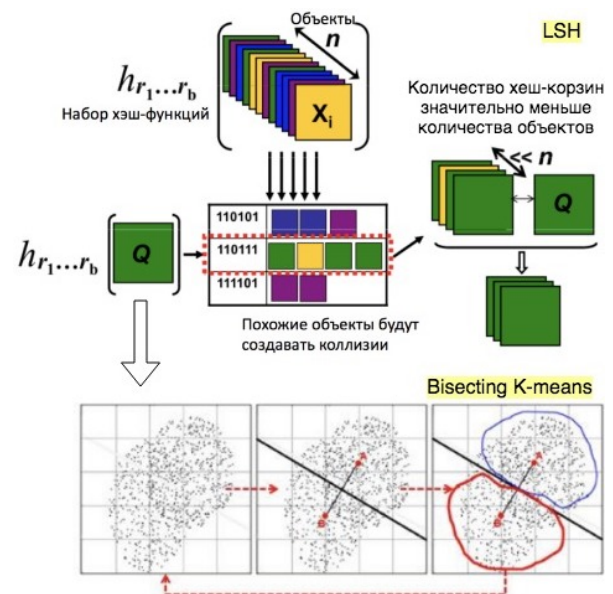


Рис. 3 – Схема предложенного метода

ЗАКЛЮЧЕНИЕ

Применение алгоритма локально-чувствительного хеширования непосредственно перед иерархической кластеризацией позволяет значительно снизить количество обрабатываемых объектов, а также (в некоторых случаях) снизить размерность пространства признаков. Разработана облачная система, демонстрирующая производительность и устойчивость описанного метода.

СПИСОК ЛИТЕРАТУРЫ

1. Drozd P. Locality-Sensitive Hashing for Customer Segmentation / P. Drozd // Open Readings 2019 abstract book / ed. E. Skliutas. – Vilnius, 2019. – P.103.
2. Koga, H. Hierarchical Clustering Algorithm Using Locality-Sensitive Hashing / H. Koga, T. Ishibashi, T. Watanabe // Discovery Science, 7 th International Conference, Padova, 2-5 oct. 2004 / ed. E. Suzuki. – Padova, 2004. – P.114-128
3. Fern, X.Z., Clustering ensembles for high dimensional data clustering / X.Z. Fern, C.E. Brodley // In Proc. International Conference on Machine Learning / ed. T. Fawcett. – Washington DC, 2003. – P.178-185

АЛГОРИТМ КОРРЕКЦИИ ПРОЕКЦИИ ОДНОСТРОЧНОГО НОМЕРА

Заерко Д. В., Липницкий В. А.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: zaerko1991@gmail.com, valipnitski@yandex.by

Дано описание алгоритма фильтрации от помех однострочных символьных строк, основанного на использовании вертикальной проекции строки. Впервые предложена идея фильтрации проекции строки на этапе ее построения и построение модельного множества проекции для фильтрации. Приведен модельный пример фильтрации проекции.

I. ВЕРТИКАЛЬНАЯ ПРОЕКЦИЯ СИМВОЛЬНОЙ СТРОКИ

Технологии распознавания символьных строк не новы и давно нашли повсеместное применение во всех сферах человеческой жизнедеятельности: от автоматического контроля затранSPORTными потоками до сканирования паспортов в различных учреждениях. Хотя и сфер применения технологий распознавания множество, их объединяет небольшой набор одних и тех же действовавших базовых алгоритмов и подходов. Процесс распознавания – длительный и многоступенчатый. Изменение на каком-либо шаге может улучшить или наоборот – ухудшить результат работы в целом всего процесса. Особенно важен первоначальный этап анализа символьной строки – при очистке её от шумов и искажений.

Распознавание номерных знаков транспортных средств относится к проблематике распознавания изображений, где основной задачей, является максимальное приближение изображаемого номера к реальному качественному снимку. Технические средства, зачастую используют алгоритмы распознавания, несовершенные в плане безошибочности распознавания номерных знаков. Сегментация и коррекция символов являются одним из фундаментальных этапов распознавания номерных знаков, так как остальные этапы всецело зависят от полученных на них результатов.

Несомненно, что этап очистки от шумов на этапе анализа строки был бы очень полезен. Однако в научной литературе слабо представлены алгоритмы очистки от помех на этапе сегментации. Наиболее часто используемые способы сегментации номеров: методы вертикальной и горизонтальной проекции и сегментация на их основе.

II. АЛГОРИТМ ШУМОФИЛЬТРАЦИИ ВЕРТИКАЛЬНОЙ ПРОЕКЦИИ

Пусть монохромное изображение определено дискретной бинарной функцией $f(x, y)$ т.е. некоторой функцией распределения $f(x, y)$ яркости или цвета на двумерной плоскости; здесь x и y – декартовы координаты точек плоскости

изображения [1]. Вертикальная проекция на ось Ox , представляет собой сумму интенсивностей пикселей по столбцам изображения. Тогда вертикальная проекция p_{Ox} функции f в точке x – это сумма всех закрашенных пикселей в столбце x этого изображения: $p_{Ox}(x) = \sum_{j=0}^{h-1} f(x, j)$; $p_{Oy}(y) = \sum_{i=0}^{w-1} f(i, y)$; где w, h – это размеры растрового изображения с анализируемой строкой. Выделяются два основных этапа: обнаружение шума и коррекция проекции по шаблону. Ключевую роль играет конечное модельное множество проекции p_{mod} . Модельная проекция символа – множество значений $p_{mod}(x_{m,0}, p_{mod}(x_{m,1}), \dots, p_{mod}(x_{m+1,z(m,m+1)})$ функции проекции $p_{mod}(x)$, содержащее $z(m, m+1) \in N$ целых значений проекций символа на сегменте $[x_m, x_{m+1}), m \in [0, w-1]$. В связи с тем, что точное число значений проекции $z(m, m+1)$ на сегменте $[x_m, x_{m+1}), m \in [0, w-1]$ зависит от ширины символа, далее будем использовать ее среднюю величину \bar{z} [2]. Это способствует совпадению количества точек в модельной и анализируемой проекциях на сегменте. Скелетная проекция это – множество значений $p_{frame}(x_{m,0}, p_{frame}(x_{m,1}), \dots, p_{frame}(x_{m+1,z(m,m+1)})$ функции проекции $p_{frame}(x)$ слева и справа ограниченных границами символа $[x_m, x_{m+1}), m \in [0, w-1]$. Скелетная проекция это – проекция символа без помех, ее строят на основе искусственного шаблона. Модельная проекция содержит оценку допустимого «искажения» проекции символа $\sum_{m=0}^{w-1} \epsilon_{mod}(p_{mod}(x_m)) = \text{sum}(\epsilon_{mod}) \geq 0$. В свою очередь, $\epsilon_{mod}(p_{mod}(x_m))$ – это допустимое искажение проекции для столбца x_m . Здесь $\epsilon_{mod}(p_{mod}(x_{m,0}, \dots, x_{m+1,\bar{z}}))$ – это поточечная разность между множеством значений проекции модельного символа $p_{mod}(x_{m,0}, \dots, x_{m+1,\bar{z}})$ и множеством значений скелетной проекции этого же символа $p_{frame}(x_{m,0}, \dots, x_{m+1,\bar{z}})$, где \bar{z} – среднее число значений проекции на фрагменте

$$\epsilon_{mod}(p_{mod}(x_m)) = \epsilon_{mod}(p_{mod}(x_{m,0}) - p_{frame}(x_{m,0}), \dots, p_{mod}(x_{m+1,\bar{z}}) - p_{frame}(x_{m+1,\bar{z}}))$$

При наличии наборов A и N модельных проекций для символов латинского алфавита и арабских цифр можно провести замену зашум-

ленного сегмента проекции на сегментскелетной проекций. Замена возможна, при условии, что анализируемый фрагмент проекции α может быть оценен исходя их имеющихся множеств модельных проекций, т.е. $\alpha \in [A \cup N]$. Оценка $\alpha \in [A \cup N]$ предполагает то, что поточечная сумма проекций для анализируемого сегмента меньше или равна наибольшей из всехпоточечных сумм модельных проекций. Опишем шаги предлагаемого алгоритма шумофильтрации.

1. Если для сегмента проекции $[x_m, x_{m+1}]$, $m \in [0, w - 1]$ выполняется условие $\sum_{m=0}^{w-1} \sum_{z=0}^{\bar{z}} p_{real}(x_{m,z} - p_{mod}(x_m, z)) = sum(\epsilon_{real}) \leq \epsilon_{mod}$, то шаг 2.
2. Замена сегмента $[x_m, x_{m+1}]$, $m \in [0, w - 1]$ проекции на скелетную $p_{frame}(x_{m,0}), \dots, p_{frame}(x_{m+1}, \bar{z})$. Если $m \leq w - 1$ то шаг 3, иначе шаг 5.
3. переход к следующему сегменту проекции: шаг 1 при $m = m + 1$
4. Если $\alpha \in [A \cup N]$, то переход к следующему множеству модельных проекций для заданного алфавита (символы латинского алфавита и арабские цифры) и шаг 1, иначе сегмент проекции не анализируем $\alpha \notin [A \cup N]$ и следует пропустить его, т.е. шаг 3.
5. Завершение алгоритма.

Рассмотрим применение алгоритма фильтрации проекции зашумленной символьной строки, представленной в виде автомобильного номерного знака на растровом изображении (рисунок 1 справа).



Рис. 1 – Монохромное, изображение исходной и зашумленной символьной строки автомобильного номера

Для наглядности приведем графики вертикальной проекций номерного знака без искусственно зашумленного тестового символа (рисунок 2) и с зашумленным фрагментом проекции (рисунок 3).

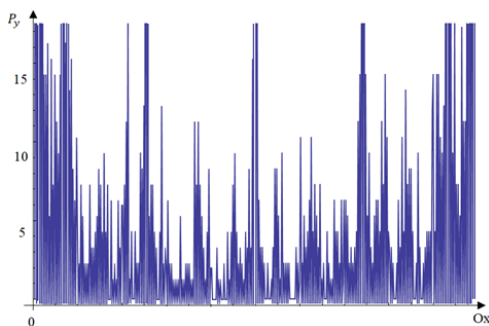


Рис. 2 – Проекция символьной строки без зашумления

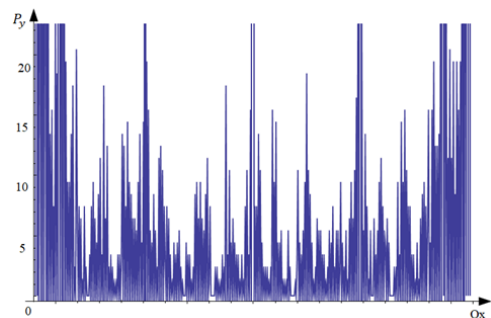


Рис. 3 – Проекция символьной строки с зашумленным фрагментом

После применения алгоритма шумофильтрации зашумленный фрагмент проекции был заменен на фрагмент скелетной проекции (рисунок 4).

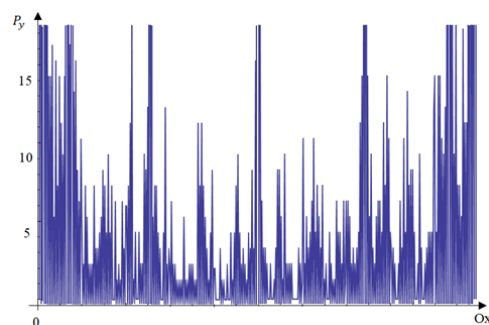


Рис. 4 – Проекция строки после применения алгоритма фильтрации для зашумлённого номера

Выбор фрагмента скелетной проекции обусловлен ее привязкой к набору модельной проекции. Проекция на рисунке 2 и на рисунке 4 совпадают почти полностью. Это означает, что, во-первых, успешно найден и заменён зашумленный фрагмент проекций, во-вторых, проведена замена на фрагмент, близкий к исходной не зашумленной проекции.

Сегментация символьной строки [3], дополненная алгоритмом шумофильтрации, представляет собой усовершенствованный процесс распознавания растровых изображений. Основная проблема алгоритма – наличие необходимых баз – словарей модельной и скелетной проекции для символов строки. Решение этой задачи требует дополнительного изучения и будет описано в дальнейшем.

СПИСОК ЛИТЕРАТУРЫ

1. Yungang Zhang and Changshui Zhang. A new algorithm for character segmentation of license plate. // IEEE Intelligent Vehicles Symposium, 2003. – P. 106–109.
2. Заерко, Д. В., Липницкий, В.А. Вычисление среднего расстояния между пиками вертикальной проекции / Д. В. Заерко, В.А. Липницкий. // Системный анализ и прикладная информатика. - 2019. – № 4. (в печати).
3. Ondrej Martinsky. Algorithmic and Mathematical Principles of Automatic Number Plate Recognition Systems. // B.Sc.Thesis, Brno University of Technology, Faculty of Information Technology, Department of Intelligent Systems, 2007.

СРАВНЕНИЕ АЛГОРИТМОВ КЛАСТЕРИЗАЦИИ В ЗАДАЧЕ ОБРАБОТКИ GPS ДАННЫХ

Иванин Н. С., Романов А. А.

Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {nikivnik, antekromanov}@gmail.com

В статье рассматривается возможность применения различных алгоритмов кластеризации для обработки GPS траекторий. Основной задачей при обработке GPS траекторий является выделение таких кластеров, которые относятся к одному дорожному сегменту. Для решения этой задачи рассматриваются алгоритмы single linkage clustering, k-means, DBSCAN. Для каждого алгоритма производится анализ его преимуществ и недостатков, а также проверяется эффективность на реальных GPS траекториях.

ВВЕДЕНИЕ

В настоящее время наблюдается значительное увеличение количества транспортных средств, оснащенных GPS датчиками, которые позволяют определять их местоположение и записывать пройденные ими траектории. Зачастую существует необходимость извлечения информации из коллекции записанных траекторий, например, дороги по которой двигалось транспортное средство или статистики скорости.

Для извлечения информации из нескольких GPS траекторий необходимо объединить эти траектории в группы. Основной проблемой при объединении траекторий является неточность измерений GPS датчика. Согласно [1] точность GPS датчиков в современных смартфонах составляет 4,9 метра. Для решения проблемы погрешности измерений GPS датчика возможно применить различные алгоритмы кластеризации.

I. АЛГОРИТМ SINGLE LINKAGE CLUSTERING

Авторы работы [2] используют алгоритм single linkage clustering (SLC), описанный в [3]. Этот метод строит иерархию кластеров в зависимости от расстояния между траекториями. На первом шаге рассчитываются расстояния между траекториями, причем каждая из траекторий образует свой собственный кластер. Затем кластеры, имеющие наименьшее расстояние объединяются в один кластер. Эта процедура повторяется до тех пор, пока не выполняются заданные условия, такие как количество кластеров либо расстояние между элементами кластера.

К достоинствам алгоритма SLC можно отнести отсутствие необходимости задания числа кластеров, а также возможность образования кластеров не эллиптической формы. К недостаткам можно отнести чувствительность к данным с большой погрешностью измерений и выбросам, сложности при разделении кластеров большого размера, сложности при выделении кластеров разного размера.

II. АЛГОРИТМ K-MEANS

В работе [4] предлагается использовать алгоритм k-means [5] с ограничениями. Целью этого алгоритма является нахождение k кластеров, расстояния между точками в которых минимально. В ходе своей работы этот алгоритм минимизирует сумму квадратов расстояний между точками и центром кластера.

Авторы [4] вводят дополнительные ограничения, накладываемые на точки:

- сильная связь. Точки обязаны находится в одном кластере;
- пустая связь. Точки не должны находится в одном кластере.

К достоинствам алгоритма k-means можно отнести довольно высокую скорость работы. Однако у алгоритма есть несколько недостатков. Первый из них – это необходимость задания точного числа кластеров. Также на первом этапе своей работы алгоритм выбирает центры кластеров случайным образом, что приводит к выделению разных кластеров при последовательных запусках алгоритма.

III. АЛГОРИТМ DBSCAN

В работе [6] для кластеризации GPS данных предлагается использовать алгоритм DBSCAN [7]. Этот алгоритм, в отличие от k-means не предполагает, что у кластера есть заранее известная форма. С точки зрения алгоритма DBSCAN кластер – это соединенные регионы, имеющие высокую плотность точек.

У алгоритма DBSCAN есть несколько важных преимуществ. Во-первых, для этого алгоритма не требуется начальное число кластеров. Также точки, имеющие значительные погрешности измерений и находящиеся в стороне от остальных точек, принимаются как шумовые и не добавляются в кластеры. Алгоритм DBSCAN показывает довольно хорошие результаты при поиске кластеров произвольной формы. Основным недостатком алгоритма является снижение

качества выделения кластеров из областей точек с изменяющейся плотностью.

IV. ПРОВЕРКА АЛГОРИТМОВ КЛАСТЕРИЗАЦИИ НА РЕАЛЬНЫХ ДАННЫХ

Для эксперимента были получены 20 GPS траекторий, записанных велосипедистами в г. Минске и проходящими через одинаковые дорожные участки. В ходе эксперимента использовались реализации алгоритмов SLC, k-means, DBSCAN из библиотеки scikit-learn[8].

При кластеризации важным является выбор метрики расстояния. Наиболее точные результаты для географических координат дает использование расстояния Евклида и Хаверсайн. По сравнению с расстоянием Хаверсайн расстояние Евклида на небольших расстояниях между точками дает маленькую погрешность, например на расстоянии 100 км между точками погрешность составляет 2 метра.

Для алгоритма SLC были выбраны следующие параметры: количество кластеров было выбрано равным 12, в качестве метрики расстояния между точками использовалось расстояние Хаверсайн, для объединения кластеров использовался критерий single. Результаты работы алгоритма SLC представлены на рисунке 1.



Рис. 1 – Результаты работы алгоритма SLC

Для алгоритма k-means были заданы следующие параметры: количество кластеров было принято равным 12, для выбора начальных кластеров использовался алгоритм k-means++, максимальное число итераций было установлено равным 300, для формирования кластеров использовался комбинированный алгоритм. Результаты кластеризации представлены на рисунке 2.

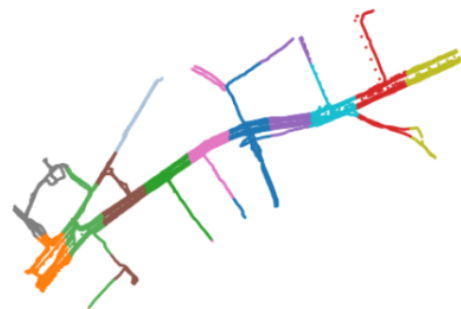


Рис. 2 – Работа алгоритма k-means

Для алгоритма DBSCAN были заданы параметры eps, min_samples, metric, algorithm. Параметр eps был выбран равным четырем средним расстояниям между точками и равнялся 7. Значение параметра min_samples было принято равным 5, что позволило улучшить кластеризацию точек, относящихся к линейным участкам дорог. Параметр algorithm определяет алгоритм поиска ближайшего соседа. Выбранный алгоритм BallTree при своей работе использует бинарное дерево, представляющие собой иерархию кластеров с точками. Точки, находящиеся в листьях дерева считаются близкими. Результаты работы DBSCAN представлены на рисунке 3.

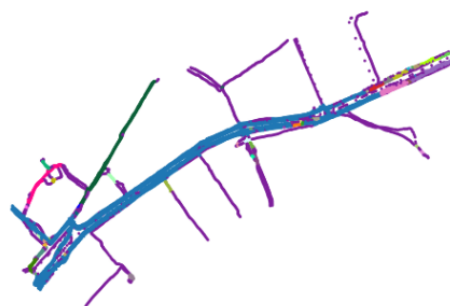


Рис. 3 – Работа алгоритма DBSCAN

Как видно из рисунков наилучшие результаты показал алгоритм DBSCAN. Он позволил выделить точки, относящиеся к одному дорожному сегменту(участок дороги без поворотов), что позволяет использовать этот алгоритм в задачах сопоставления карт и разбиения траектории на отдельные участки. Алгоритм k-means может быть использован в задаче разделения дорожной сети на равные участки.

СПИСОК ЛИТЕРАТУРЫ

1. GPS Accuracy [Электронный ресурс] / Режим доступа: <https://www.gps.gov/systems/gps> – Дата доступа: 09.10.2019.
2. Crowdatlas: self updating maps for cloud and personal use / Y. Wang [et al.] // Proc. of 11th inter. conf. mobile systems, applications and services. – 2013. – P. 27–40.
3. Sibson, R. Slink: An optimally efficient algorithm for the single-link cluster method / R. Sibson // The Computer Journal. – 1973. – Vol. 6, № 1. – P. 30–34.
4. Constrained k-means clustering with background knowledge / K. Wagsta [et al.] // Proc. of the 18 International Conference on Machine Learning(ICML 2001). – 2001. – P. 577–584.
5. MacQueen, J. Some methods for classification and analysis of multivariate observations / J. MacQueen // Proc. of the 5 Symposium on Math, Statistics, and Probability. – 1967. – Vol. 1. – P. 281–296.
6. Boeing, G. Clustering to Reduce Spatial Data Set Size / G. Boeing // SSRN Electronic Journal. 10.2139/ssrn.3145515. – 2018. – 7 p.
7. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise / M. Ester [et al.] // Proc. of the 2nd International Conference on Knowledge Discovery and Data Mining. – 1996. – P. 226–231.
8. scikit-learn: Machine Learning in Python [Электронный ресурс] / Режим доступа: <https://scikit-learn.org> – Дата доступа: 09.10.2019.

ПРИМЕНЕНИЕ ЯЗЫКА ПРОГРАММИРОВАНИЯ SCALA В РЕШЕНИИ ЗАДАЧ ОБРАБОТКИ ДАННЫХ

Иванюк А. И., Клапатов И. А., Чибисов И. В.

Кафедра программного обеспечения информационных технологий, кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: ivanyuk42@gmail.com

В рамках данного исследования проведен анализ и предпринята попытка использования языка программирования Scala для обработки больших объемов данных с целью получения определённых аналитических данных. Проведено сравнение технических средств для обработки больших данных на языке Scala с уже зарекомендовавшими себя технологиями Hadoop и MapReduce. Исследованы возможности применения фреймворка Apache Spark как платформы для обработки данных.

ВВЕДЕНИЕ

В последнее десятилетие в связи с развитием горизонтально масштабируемых систем и эффективных хранилищ структурированных и неструктурированных данных значительно увеличилась популярность систем для обработки и анализа больших объемов данных. Развившиеся технические средства позволяли организовывать обработку огромных массивов данных, в некоторых случаях покрывающих весь мировой объем данных конкретной предметной области [1].

Одновременно с этим свойства, специфичные для обработки данных, такие как неизменяемость данных, применение операций функторов и редукции свойственны так же и функциональному программированию. Язык программирования Scala является типичным современным представителем функционального языка программирования общего назначения, который работает на платформе JVM. Эта особенность позволяет достичь достаточной гибкости при использовании языка и обеспечивает доступ к большому числу библиотек. Использование данного языка также возможно для решения задач обработки больших объемов данных, что обусловлено функциональным подходом и гибкостью языка.

I. БОЛЬШИЕ ДАННЫЕ

Традиционно для определения принадлежности задачи к области больших данных используют набор признаков называемых «три V». Эти признаки включают следующие элементы.

1. Объем (volume) – физически большой объем обрабатываемых данных;
2. скорость (velocity) – необходимость высокой скорости обработки информации, которая обусловлена наличием современных технологических средств, а также скорость прироста объема информации;
3. многообразие (variety) – возможность обработки разнородной информации, структурированной или неструктурированной [1].

Обработка данных, которым свойственны перечисленные признаки, требует специализи-

рованных средств. Такие средства включают NoSQL системы управления базами данных, средства распределенной параллельной обработки данных, специализированные алгоритмы (например, MapReduce) и библиотеки (например, входящие в пакет Hadoop). NoSQL системы позволяют хранить неструктурированные данные, свойственные для больших данных. В то же время комбинация из использования систем MapReduce и Hadoop позволяют преобразовывать эти данные. Рассмотрим эту типичную комбинацию зарекомендовавших себя решений.

II. HADOOP И MAPREDUCE

Hadoop представляет собой фреймворк и набор библиотек для создания и выполнения программ, работающих распределенно на большом количестве узлов. Он позволяет организовать распределенную и параллельную обработку данных, используя мощности большого количества машин одновременно.

Структурно Hadoop разделен на несколько составных частей.

- Hadoop Common – набор утилит для работы с другими составными частями, организации инфраструктуры и управления процессом обработки данных;
- HDFS – распределенная файловая система, которая специализирована на хранении файлов большого объема;
- YARN – модуль отвечающий за планирование задач внутри кластера и управления его ресурсами;
- Hadoop MapReduce – элемент фреймворка и шаблон, позволяющий реализацию MapReduce алгоритмов и их выполнение в среде Hadoop [2].

Алгоритмы MapReduce позволяют обрабатывать большие объемы данных параллельно на многих узлах, при этом данные должны представлять собой набор пар «ключ – значение». Работа приложения, разработанного с использованием данных алгоритмов, состоит из двух шагов: Map и Reduce. Данные шаги основаны на двух функциях высшего порядка, которые ис-

пользуются в функциональных языках программирования и носят соответствующие названия. На шаге Map происходит преобразование данных, какая-либо их предварительная обработка, при которой происходит заданное преобразование значений из пар «ключ – значение» входных данных. На шаге Reduce происходит свёртка значений, соответствующих одинаковому ключу [3]. При этом происходит последовательное выполнение заданной операции с накоплением результата.

III. SCALA

Язык программирования Scala представляет собой функциональный язык программирования с набором стандартных библиотек, которые, в том числе, включают функции работы с коллекциями. Семантическая близость операций Map и Reduce с операциями преобразования коллекций в функциональных языках программирования делает язык Scala хорошим кандидатом на реализацию соответствующих функций, пригодных к выполнению параллельно и распределенно.

Сама по себе стандартная библиотека языка Scala предоставляет возможности по работе с коллекциями данных только в рамках выполнения одного процесса на одном узле. Для выполнения распределенных вычислений на большом числе узлов необходимо использование специальных технических средств, которые выходят за пределы самого языка, но используют этот язык программирования как основу. Проект, использующий язык Scala для обработки больших объемов данных носит название Apache Spark.

IV. АРАШЕ SPARK

Apache Spark – это фреймворк, использующий инфраструктуру Hadoop для распределенной обработки больших объемов данных. Он является заменой MapReduce и обладает рядом преимуществ перед ним. По аналогии с MapReduce данный фреймворк позволяет проводить последовательные операции обработки и свертки, однако, если MapReduce ограничен выполнением двух шагов, то Spark позволяет комбинировать операции преобразования данных в последовательности [4]. При этом явным недостатком MapReduce является необходимость записи и чтения данных с диска после каждого выполнения пары операций Map и Reduce. В случае же использования Apache Spark обработка данных выполняется в основном в оперативной памяти, что значительно увеличивает скорость его работы. В случае со Spark основным замедляющим фактором является возникающая необходимость перераспределять данные между уз-

лами, которая возникает, например, при изменении ключа записи. Ключ записи используется для привязки данных к определенным узлам, что необходимо для быстрого выполнения операции свертки. В случае же его изменения, данные перераспределяются.

Для работы с данными Spark предоставляет две абстракции: трансформация и действие. Трансформации задают последовательность изменений изначального набора данных. Они включают в себя, например, такие операции, как map, filter или distinct. Применение трансформаций не вызывает реальных изменений данных, а лишь описывает будущие изменения. Для применения заданных трансформаций необходимо выполнить какое-либо действие. Действие – это операция, которая позволяет материализовать результат выполнения трансформаций. Примерами действий могут служить такие примеры, как saveAsTextFile (сохранение набора данных на диск), take (получение определенного количества элементов набора данных) или reduce (операция свертки).

Таким образом, Apache Spark позволяет добиться значительного выигрыша в скорости работы за счет активного использования оперативной памяти узла. Данное преимущество часто используется для реализации алгоритмов машинного обучения с использованием данного фреймворка. Напротив, MapReduce обладает значительно меньшей производительностью, но позволяет обрабатывать гораздо большие объемы данных. Если для Spark эксперты рекомендуют одновременную обработку не более чем 1-2 Тбайт данных, то Hadoop MapReduce может обрабатывать объемы вплоть до нескольких петабайт [5].

Также Apache Spark может быть использован в отрывке от инфраструктуры Hadoop и работать с различными хранилищами данных, например, с Apache Cassandra или Amazon S3 [5].

Перечисленные выше особенности позволяют использовать Apache Spark для решения широкого класса задач обработки больших объемов данных, в том числе задач машинного обучения и статистического анализа.

СПИСОК ЛИТЕРАТУРЫ

1. Силен, Д. Основы Data Science и Big Data. Python и наука о данных / Д. Силен, А. Мейсман, М. Али // Питер, 2018. – 336 с.
2. Уайт, Т. Hadoop. Подробное руководство / Т. Уайт // Питер, 2013. – 672 с.
3. Holmes, A. Hadoop in Practice / A. Holmes // Manning Publications, 2014. – 512 p.
4. Bengfort, B. Data Analytics with Hadoop / B. Bengfort, J. Kim // O'Reilly Media, 2016. – 288 p.
5. Frampton, M. Mastering Apache Spark / M. Frampton // Packt Publishing, 2015. – 320 p.

ДЕТЕКТИРОВАНИЕ НАЛИЧИЯ СКРЫТОЙ ИНФОРМАЦИИ В ЦИФРОВЫХ ИЗОБРАЖЕНИЯХ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Кадан А. М., Сазановец И. А.

Кафедра системного программирования и компьютерной безопасности, Гродненский государственный университет им. Янки Купалы

ООО «ИнтексСофт»

Гродно, Республика Беларусь

E-mail: kadan@mf.grsu.by, sazanovec_ia_13@mf.grsu.by

Для детектирования наличия скрытой информации в графических файлах рассматриваются методы на основе технологий машинного обучения. Детектирование ведется с использованием «слепых» методов – при отсутствии данных об исходном алгоритме, использованном для внедрения скрытой информации. Для формирования датасетов, используемых при обучении моделей детектирования, представлены методы на основе вейвлет-разложения. Приведены результаты испытания обученных моделей на тренировочных наборах данных.

ВВЕДЕНИЕ

Потребность скрыть информацию или поделиться ею, не привлекая внимания посторонних, существовала на протяжении всей истории общественных отношений. Научное направление, изучающее вопросы скрытой передачи информации, известно как стеганография. Наметившийся в последние годы интерес к стеганографическим методам связан в немалой степени с тем, что в отличие от криптографии, их использование практически не регулируется законодательством.

Современные методы стеганографии широко используют компьютерную технику для внедрения «стегоинформации» в другие цифровые данные, называемые «стегоконтейнерами», с качеством которых могут выступать цифровые изображения, аудио- или видео-данные, пакеты сетевого трафика и многое другое.

Одним из ключевых требований к стеганографическим алгоритмам является то, что внедрение информации не должно заметно изменять характеристики стегоконтейнера. Поэтому стеганографические алгоритмы часто эксплуатируют ограниченность биологических систем восприятия человека. Например, при сокрытии информации в изображениях, стегоалгоритмы изменяют интенсивность цветов так, чтобы, с одной стороны, этими изменениями внедрить стегоинформацию, а с другой, чтобы эти изменения не воспринялись органами зрения человека. В основе стегоалгоритмов по работе со звуком лежит тот же принцип – записываемая информация изменяет высокие частоты аудиосигнала, что вряд ли будет заметно при прослушивании [1].

I. СХЕМА И СРЕДСТВА СОКРЫТИЯ ИНФОРМАЦИИ

Алгоритмически, стеганография состоит из двух фаз: одна для сокрытия информации, дру-

гая для извлечения. На случай, если всё-таки обнаружится факт наличия скрытого сообщения, в большинстве стеганографических программ перед внедрением сообщения его вначале зашифровывают. Базовая модель стеганографии представлена на рисунке 1.

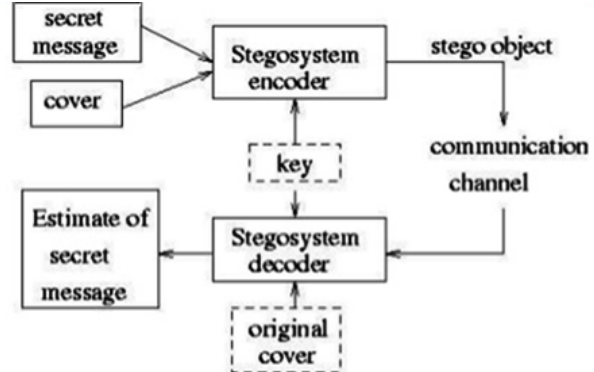


Рис. 1 – Базовая модель стеганографии

Актуальной задачей является детектирование наличия внедрённой информации при условии, что ничего не известно об исходном стегоалгоритме. В стегоанализе методы, решающие задачи такого типа, принято называть «слепыми».

II. ФОРМИРОВАНИЕ ИЗОБРАЖЕНИЙ, СОДЕРЖАЩИХ СКРЫТЫЕ ДАННЫЕ

Алгоритмы стеганографии формируют разные типы искажений исходного контейнера. Поэтому написать детерминированный алгоритм для детектирования наличия стегоинформации становится едва ли возможным. Именно в таких ситуациях прибегают к использованию методов машинного обучения. Для обучения моделей машинного обучения были выбраны 750 графических изображений из открытых наборов данных, предварительно уменьшенные до размеров от 640x480 до 1147x768 пикселей. Стегоинформация была внедрена с помощью трех программ:

- Steganography Software F5 (алгоритм f5) [2];
- StegHide (стеганография на основе теории графов) [3];
- OpenStego (RandomLSB, модифицированный алгоритм наименьшего значащего бита) [4].

III. ФОРМИРОВАНИЕ ФАЙЛОВ ПРИЗНАКОВ

Для использования моделей машинного обучения необходимо на основе исходных графических изображений сформировать датасет – файл признаков, который бы отражал характерные особенности «чистых» изображений и изображений, содержащих стегоинформацию. Датасет представлен в виде файла csv-формата. Каждая запись датасета содержит 84 признака, а также 85-й классификационный признак: «0» для «чистого» изображения и «1» в противном случае. В итоге был сформирован датасет из 3000 записей – 750 записей для «чистых» изображений, и по 750 записей для изображений со стегоинформацией, внедренной программами Steganography Software F5, StegHide и OpenStego соответственно.

Для построения по изображению вектора признаков, поскольку мы имеем дело с цифровыми изображениями, имеет смысл применять дискретные вейвлет-преобразования. Разложение будем производить по вейвлет-функциям Хаара, db2 и bior1.3.

Так как каждый цветовой канал в изображении представляется прямоугольной матрицей, то будем использовать дискретное вейвлет-преобразование, реализованное методом *wavedec2* модуля *PyWavelets* языка Python.

Метод *wavedec2* возвращает структуру вида $[cAn, (cNn, cVn, cDn), \dots (cN1, cV1, cD1)]$, где:

- cAn – аппроксимационный коэффициент разложения n -го уровня;
- cNn – горизонтальный коэффициент разложения n -го уровня;
- cVn – вертикальный коэффициент разложения n -го уровня;
- cDn – диагональный коэффициент разложения n -го уровня.

Признаки, характеризующие графическое изображение, будем формировать с использованием дискретных вейвлет-преобразований (с разложением не выше 3-го уровня) и статистических моментов 1-4 порядков.

IV. ПРИМЕНЕНИЕ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Задача слепого детектирования наличия стегоинформации в графическом изображении, рассматриваемая в данной работе, относится к задачам бинарной классификации и может быть решена в рамках технологии машинного обучения с учителем.

Был проведен эксперимент по применению для решения поставленной задачи следующих методов, с использованием их реализаций из библиотеки *scikit – learn* для языка Python:

- алгоритма К-ближайших соседей;
- наивного байесовского классификатора;
- дерева принятия решений;
- линейной регрессии;
- метода опорных векторов;
- нейронной сети прямого распространения.

Лучшие результаты удалось получить при использовании метода опорных векторов и нейронной сети прямого распространения с архитектурой «многослойный перцептрон» с двумя скрытыми слоями. В случае использования вейвлет-функций *db2* и *bior1.3* результаты оказались хуже, чем при использовании вейвлет-функции Хаара.

Также они хуже при использовании *minimax*-нормализации вместо стандартной.

Наиболее существенно влияющим на результат было значение признаков, связанных с моментами 3-го и 4-го порядков. Признаки, связанные с моментами 1-го и 2-го порядка не оказывали существенного влияния на результат. Также существенно не повлияли на результат значения признаков, вычисленных на горизонтальных коэффициентах вейвлет-разложения.

ЗАКЛЮЧЕНИЕ

Можно утверждать, что достаточно эффективными методами для решения задачи «слепого детектирования» являются метод опорных векторов с параметрами $C=1000$, $\text{gamma}=0.001$ и $\text{kernel}=rbf$ и многослойный перцептрон с двумя скрытыми слоями (90 и 20 нейронов).

В качестве эффективных признаков могут быть использованы коэффициенты асимметрии и эксцесса, вычисленные в частотной плоскости для аппроксимационного, вертикальных и диагональных коэффициентов двумерного трёхуровневого вейвлет-преобразования с использованием вейвлет-функции Хаара. С помощью обученных моделей можно с вероятностью, близкой к 0.7, предсказать, содержит ли графическое изображение скрытое сообщение или нет.

СПИСОК ЛИТЕРАТУРЫ

1. David Wheeler. Audio Steganography Using High Frequency Noise Introduction [Электронный ресурс] / RIT Scholar Works. 2012. – Режим доступа: <https://pdfs.semanticscholar.org/d547/3318c5c9171fe38abc550b89a15022d559cb.pdf>. – Дата доступа: 28.09.2019.
2. F5-steganography [Электронный ресурс] / The world's leading software development platform GitHub. – Режим доступа: <https://github.com/matthewgao/F5-steganography>. – Дата доступа: 28.04.2019.
3. Steghide [Электронный ресурс] / Sourceforge. – Режим доступа: <http://steghide.sourceforge.net/>. – Дата доступа: 28.04.2019.
4. OpenStego [Электронный ресурс] / OpenStego. – Режим доступа: <https://www.openstego.com/>. – Дата доступа: 28.04.2019.

СИСТЕМА ДЛЯ ПРОГНОЗИРОВАНИЯ ПОПУЛЯРНОСТИ ПУБЛИКАЦИЙ

Калоша А. Л., Хоронеко М. П., Медунецкий М. М.

Кафедра информатики, Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: andreikalosha@mail.ru

Цель данной работы заключается в создании системы для прогнозирования популярности публикаций. В данной системе используется нейронная сеть, которая обучена на наборе метрик, описывающих качество и популярность публикаций. В качестве набора метрик используется количество лайков, просмотров и репостов.

ВВЕДЕНИЕ

Объем информации, доступной в сети Интернет, растет с каждым годом. Причем большая часть этой информации представляет собой тексты на естественном языке. В зависимости от области знаний, информация может быть представлена в виде статей, комментариев или сообщений на публичном форуме. Информация в сети Интернет дублируется, уточняется и пополняется ежедневно. Нетрудно понять, что имеющиеся в данный момент доступные ресурсы всемирной сети представляют собой колоссальную базу знаний, представленных в форме, сложно поддающейся компьютерной обработке – в виде текста [1].

Как правило, изучить весь контент (текст) не представляется возможным даже в отдельных областях, поэтому приходится фильтровать получаемую информацию и выбирать лучшую.

Назначение разрабатываемой системы заключается в предсказании популярности статей через определенный промежуток времени. Статья считается популярной при высоком количестве лайков, репостов или просмотров. Данные метрики зависят от множества факторов, таких как название, авторов, время публикации и содержание статьи. Эти параметры наилучшим образом отражают популярность (качество) статьи. Правильно обученная нейронная сеть позволяет с высокой точностью предсказать значения метрик популярности для неопубликованного контента.

Для обучения нейронной сети была выбрана библиотека TensorFlow как один из лучших инструментов машинного обучения. TensorFlow – это библиотека программного обеспечения с открытым исходным кодом для численного расчета с использованием графиков потока данных [2].

Существует прямая зависимость между скоростью обучения нейронной сети и точностью предсказания. Для ускорения процесса обучения используется вычислительная мощность видеокарты, а именно технология CUDA. CUDA – это архитектура параллельных вычислений от

NVIDIA, позволяющая существенно увеличить вычислительную производительность благодаря использованию GPU (графических процессоров) [3].

Нейронная сеть – это громадный распределенный параллельный процессор, состоящий из элементарных единиц обработки информации, накапливающих экспериментальные знания и предоставляющих их для последующей обработки [4].

Нейронная сеть сходна с мозгом с двух точек зрения:

1. Знания поступают в нейронную сеть из окружающей среды и используются в процессе обучения;
2. Для накопления знаний применяются связи между нейронами, называемые синаптическими весами [4].

I. АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ

Нейронная сеть состоит из 4 слоев (входной, два промежуточных и выходной слой) (см. рис. 1). На промежуточных слоях используется функция активации LeakyReLU, на выходном слое применяется функция softmax. Между всеми слоями, кроме последнего, используется нормализация данных.

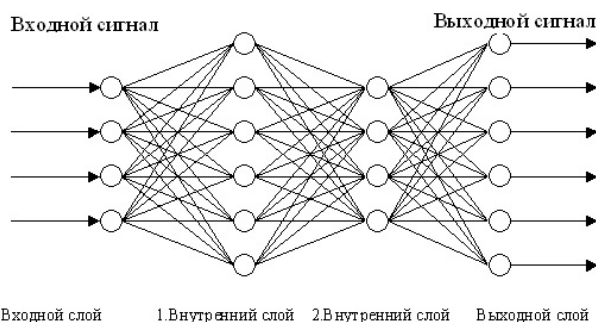


Рис. 1 – Архитектура нейронной сети

II. ОБУЧЕНИЕ НЕЙРОННОЙ СЕТИ

Для обучения нейронной сети необходимо большое количество статей и метаданных, таких как автор, дата создания, ключевые слова и другие.

Перед обучением данные делятся на 2 части: для тестирования и для обучения.

Опишем процедуру обучения нейронной сети. На вход нейронной сети подается матрица векторов MV , каждый вектор V которой содержит информацию о конкретном атрибуте публикации (например, авторе). Для формирования отдельного вектора V перед обучением необходимо получить словарь D всех значений атрибута публикации. Словарь D сортируется по убыванию и отбрасываются последние N значений, чтобы нейронная сеть не обучалась на редко встречающихся элементах, и тем самым не ухудшалась точность классификации. Указанная выше процедура выполняется для каждого атрибута. Для каждого автора публикации, производится поиск в словаре D , если данный автор найден, то под индексом найденного автора в вектор V ставится единица, иначе – ноль. Таким образом, заполняются все векторы матрицы MV [5].

Выходной вектор R описывает количество просмотров через заданный промежуток времени и состоит из единственного дробного числа, находящегося в диапазоне от нуля до единицы. Единица означает максимальное количество просмотров, в данном исследовании выбрано 50 миллионов [5].

Промежуток времени, на который нейронная сеть способна предсказать популярность публикации, является статическим и определяется до обучения нейронной сети. Т.е. что бы изменить этот параметр нужно обучить нейронную сеть заново. Для предсказания популярности публикации через несколько временных отрезков, например, неделя, месяц и год можно использовать два варианта:

1. Обучить несколько нейронных сетей;
2. Изменить архитектуру нейронной сети таким образом, что бы на выходном слое был вектор, содержащий значения популярности для нескольких временных интервалов.

У каждого из способов есть достоинства и недостатки и выбирать нужно, исходя из постановки задачи. Плюсом при использовании первого варианта, является простота реализации и тестирования приложения. Минусом является необходимость поддержания нескольких копий приложения, по одному на каждый из временной интервал.

Плюсом при использовании второго варианта является необходимость поддержания только одного экземпляра приложения вместо несколь-

ких, как в первом варианте. Минусом является сложность создания архитектуры, создания приложения и оценки результата, т.к. нейронная сеть может обучиться предсказывать некоторые временные участки лучше других, хотя в среднем результат будет оптимальным.

После обучения нейронной сети загружаются тестовые данные, и выполняется процедура тестирования. Далее на основании полученных векторов нейронная сеть предсказывает популярность статей через заданный промежуток времени. Данные обрабатываются и сохраняются в excel для анализа.

ЗАКЛЮЧЕНИЕ

Обучение сети производилось на более чем 100 000 текстов, что занимает от 4 до 16 часов, в зависимости от глубины обучения и точности результата. В результате нейронная сеть способна предсказать количество просмотров с точностью в 75%. Верным считается ответ, находящийся в диапазоне $\pm 200\,000$ просмотров от ответа. Максимальное количество просмотров при обучении составляло 48 миллионов. Коэффициент корреляции для массивов ответов и предсказанных значений составляет 0,3. Это означает, что между входными и выходными данными есть зависимость. Подобрать более точно входные данные или параметры нейронной сети, можно увеличить точность системы [5].

СПИСОК ЛИТЕРАТУРЫ

1. Степанов, П. А. Системы анализа текстов естественного языка / П. А. Степанов. – Тамбов: Издательство: Грамота, 2013. – С. 159–161.
2. Library for numerical computation using data flow graphs [Электронный ресурс] / Официальный сайт фреймворка «TensorFlow». – Режим доступа: <https://www.tensorflow.org>. – Дата доступа: 11.09.2019.
3. Параллельные вычисления CUDA [Электронный ресурс] / Официальная страница архитектуры «CUDA». – Режим доступа: <http://www.nvidia.ru/object/cuda-parallel-computing-ru.html>. – Дата доступа: 01.08.2019.
4. Хайкин, С. Нейронные сети: полный курс, 2-е издание / С. Хайкин. – М.: Издательский дом «Вильямс», 2006. – 1104 с.
5. Калоша, А. Л. Система предиктивного анализа для классификации документов текстовых коллекций / А. Л. Калоша, М. А. Медунецкий, М. П. Хорошко // BIG DATA Advanced Analytics: collection of materials of the fourth international scientific and practical conference, Minsk, Belarus, May 3 – 4, 2018 / editorial board: M. Batura [et al.]. – Minsk, BSUIR, 2018. – P. 467–468.

ТЕХНОЛОГИИ ДОСТУПА К ЦИФРОВЫМ УСТРОЙСТВАМ ДЛЯ ЛЮДЕЙ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ

Кармаз А. М., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: karmazet@mail.ru, gurinovich@bsuir.by

Современное развитие общества предполагает, что люди с ограниченными возможностями, не должны испытывать ограничений в развитии своих способностей и участия в жизни общества. Безбарьерная среда предполагает развитие технологий которые помогают снизить большинство этих ограничений. Используя компьютерные технологии для таких задач, как чтение и написание документов, общение с другими людьми и поиск информации в интернете, учащиеся и сотрудники с ограниченными возможностями могут самостоятельно выполнять более широкий спектр действий. Тем не менее, люди с ограниченными возможностями сталкиваются с различными препятствиями для использования компьютера. В работе исследуются и систематизируются подходы доступа к цифровым устройствам для людей с ограниченными возможностями.

ВВЕДЕНИЕ

По данным Всемирной организации здравоохранения (ВОЗ), примерно 15% населения мира живет с той или иной формой инвалидности. Инвалиды могут в равной степени участвовать в жизни общества и вносить существенный вклад в экономику, если имеются соответствующие инструменты.

Более того, информационные и коммуникационные технологии (ИКТ) могут стать мощным инструментом поддержки образования и интеграции для людей с ограниченными возможностями. Современные технологии могут позволить людям с ограниченными возможностями улучшить качество жизни. Внедрение таких технологий может позволить людям использовать все существующие цифровые устройства.

I. КОНЦЕПЦИЯ ДОСТУПНОСТИ

Доступность - это дизайн продуктов, устройств, услуг или сред для людей с ограниченными возможностями.

Доступность можно рассматривать как «возможность доступа» и выгоду от какой-либо системы или объекта. Концепция направлена на предоставление доступа людям с ограниченными возможностями или особым потребностям, или предоставление доступа посредством использования вспомогательных технологий; однако исследования и разработки в области доступности приносят пользу всем.

Для людей с ограниченными возможностями доступ означает возможность использовать продукт или услугу так же эффективно, как и человек без инвалидности. Это означает использование принципов инклюзивного дизайна, чтобы сделать продукты и услуги доступными для более широкой части населения. В некоторых случаях это невозможно, и для устранения этого

недостатка могут потребоваться вспомогательные технологии. Изменение отношения людей к инвалидности имеет основополагающее значение для достижения большей доступности.

II. ПРАВА ЛЮДЕЙ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ В ЦИФРОВОМ МИРЕ

Правительства более ста стран подписали и ратифицировали конвенцию ООН о правах инвалидов. Обязательства включают реализацию мер по проектированию, разработке, производству и распространению доступных информационно-коммуникационных технологий на ранней стадии, поэтому они становятся доступными для людей с ограниченными возможностями при минимальных затратах.

Популярные компании сейчас имеют прогрессивное отношение к доступности. Во-первых, правительства включают критерии доступности в свою политику государственных закупок (через так называемые руководящие принципы раздела 508), тем самым стимулируя промышленность поставлять более доступные продукты своим агентствам. Во-вторых, судебное разбирательство в соответствии с законодательством о дискриминации и телекоммуникациях привлекло внимание компаний к возможным последствиям, если потребности в доступе не будут должным образом учтены.

III. ТЕХНИЧЕСКОЕ СООБЩЕСТВО

Существуют новые потенциальные технические решения, которые могут принести пользу как людям с ограниченными возможностями, так и населению в целом. Такие продукты, как распознаватели речи (изначально разработанные для людей с ограниченными движениями рук) и сканер (разработанный как часть устройства для чтения документов в сочетании с син-

тезом речи для слепых людей), теперь являются продуктами массового рынка.

Существуют международно-признанные правила доступности, разработанные W3C для веб-контента, средств разработки и пользовательских интерфейсов. Эти основополагающие принципы, обеспечивающие рекомендации по обеспечению доступности веб-контента (WCAG), используются многими правительствами для создания доступных веб-сайтов. Концепции WCAG предусматривают, что веб-сайты должны быть «воспринимаемыми», «работоспособными», «понятными» и «надежными». Без постоянных усилий по повышению осведомленности новые типы продуктов могут создавать новые барьеры. Проектирование должно осуществляться в соответствии с принципами WCAG.

IV. ВСПОМОГАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Вспомогательная технология - это общий термин, используемый для описания любого продукта или услуги на основе технологии, которая помогает людям с ограниченными возможностями жить, учиться, работать и наслаждаться жизнью. Ниже приведен пример основных категорий этих вспомогательных технологий.

Программы чтения с экрана - это программные продукты, разработанные для слепых пользователей, но они также полезны для пользователей с ограниченными возможностями обучения. Программы чтения с экрана находят информацию, видимую на экране компьютера, и озвучивают ее с помощью программного обеспечения преобразования текста в речь и, иногда, аппаратного обеспечения.

Обновляемый дисплей Брайля - это тактильное устройство, которое поднимает или понижает точечные рисунки по команде электронного устройства, обычно компьютера. Дисплей Брайля является основным средством доступа к компьютерам для глухих пользователей.

Экранные увеличители - это программные решения для людей с плохим зрением. Эти продукты позволяют пользователю увеличить размер изображений и текста, отображаемых на экране.

Адаптивные клавиатуры - предназначены для пользователей с ограниченными физическими возможностями, которые не могут использовать стандартную клавиатуру. Пользователям с ограниченным диапазоном движения могут потребоваться клавиатуры меньшего размера. И наоборот, для тех, у кого нет точного управления двигательным аппаратом, может потребоваться клавиатура несколько большего размера.

V. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЯ

На практике были реализованы WEB интерфейсы в соответствии с концепциями W3C и с использованием ARIA атрибутов и Java Script. В результате проверки и анализа разработанных интерфейсов при помощи программ чтения с экрана, в частности JAWS и NVDA, было обнаружено, что эти программы различным образом обрабатывают и озвучивают одинаковые элементы интерфейсов. Также можно выделить, что эти программы по разному взаимодействуют с различными браузерами. Например обе программы чтения с экрана достаточно хорошо озвучивают элементы интерфейсов в Google Chrome браузере, но имеют много трудностей при взаимодействии с Internet Explorer. Это говорит о том, что выбор браузеров при работе с программами для чтения с экрана очень важен. Ещё следует отметить, что NVDA распространяется в абсолютно бесплатном доступе, в то время как JAWS распространяется по лицензиям. Это также влияет на выбор людей с ограниченными возможностями между этими программами.

VI. ЗАКЛЮЧЕНИЕ

В работе рассмотрены подходы которые помогают расширить доступ к инфокоммуникативным технологиям для людей с ограниченными возможностями. Независимо от проблем, с которыми они могут столкнуться, люди с ограниченными возможностями могут внести свой вклад в общество. Было проверено взаимодействие различных браузеров с основными программами для чтения с экрана (JAWS, NVDA). Все вышперечисленные технологии способны работать с WEB интерфейсами при помощи стандартных средств разработки (JavaScript, HTML, CSS, ARIA). Правительства, промышленность и другие ключевые заинтересованные стороны должны сделать доступность приоритетом в своей текущей работе, индивидуально и совместно. Интернет-сообщество должно работать вместе, чтобы внести изменения как в социальном, так и в экономическом плане.

СПИСОК ЛИТЕРАТУРЫ

1. Доступность интернета: использование интернета людьми с ограниченными возможностями [Электронный ресурс] - Режим доступа: <https://www.internetsociety.org/resources/doc/2012/internet-accessibility-internet-use-by-persons-with-disabilities-moving-forward>. - Дата доступа: 04.10.2019.
2. Права инвалидов и доступ к цифровому миру [Электронный ресурс] - Режим доступа: <https://dredf.org/media-disability/disability-rights-and-access-to-the-digital-world>. - Дата доступа: 06.10.2019.

ГРАФИЧЕСКАЯ ВИЗУАЛИЗАЦИЯ СТАТИСТИЧЕСКИХ ДАННЫХ В ВИРТУАЛЬНОЙ РЕАЛЬНОСТИ

Кисель А. Н., Головатая Е. А.

Кафедра интеллектуальных систем, Белорусский государственный университет, Факультет радиофизики и компьютерных технологий
Минск, Республика Беларусь

E-mail: amoroz408@gmail.com, katerina-golovataya@yandex.ru

В работе предложена концепция модели представления данных в виртуальной реальности. На основе проведенного анализа создано виртуальное окружение с поддержкой взаимодействия очков виртуальной реальности Oculus Rift и манипуляторов Oculus Touch для последующего внедрения предложенных концепций.

В связи с тем, что количество информации непрерывно растет, возникла необходимость в разработке дополнительных инструментов и способов её анализа и интерпретации. На сегодняшний день использование возможностей виртуальной реальности в полной мере позволяет использовать человеческие способности воспринимать 80% информации с помощью зрения, при том, что запоминается лишь 20% увиденного, 40% услышанного и увиденного, и 70% увиденного, услышанного и сделанного. К тому же, у виртуальной реальности есть ряд преимуществ, таких как способность удерживать человеческое внимание путем взаимодействия с окружающим трехмерным миром и уменьшением информационной перегрузки. Взаимодействие с виртуальной реальностью интуитивно понятно, как и контакт с физическим миром, и позволяет передавать информацию о движении человека на основе 6 степеней свободы при помощи специального оборудования.

Цель работы – проектирование механизма виртуальной реальности для интерактивной визуализации данных. Таким образом, в работе рассматривается теория визуализации данных, проводится анализ возможностей систем виртуальной реальности, а также представляются основные этапы создания виртуальной системы, которая позволяет взаимодействовать с соответствующей аппаратурой.

I. ПОНЯТИЕ ВИЗУАЛИЗАЦИИ ДАННЫХ, ОСНОВНЫЕ ПРИНЦИПЫ ПРЕДСТАВЛЕНИЯ ИНФОРМАЦИИ

Наглядная демонстрация массивов всевозможной информации, целью которой является осуществление основной концепции информации, а именно определение отношений в информации, изображение распределения данных, композиции или сравнения данных – это и есть визуализация данных. Самые простые, а потому и самые распространенные методы визуализации – это графики.

Существует несколько типов визуализации:

- Визуальное представление количественной информации в схематической форме. К этой группе можно отнести диаграммы, гистограммы и спектрограммы, таблицы и различные точечные графики.
- Формы, усиливающие восприятие и анализ информации. Например, карта и полярный график, временная линия и график с параллельными осями, диаграмма Эйлера.
- Концептуальная визуализация позволяет разрабатывать сложные концепции, идеи и планы с помощью концептуальных карт, диаграмм Ганта, графов с минимальным путем.
- Стратегическая визуализация переводит в визуальную форму различные данные об аспектах работы организаций.
- Метафорическая визуализация организует структурную информацию с помощью пирамид, деревьев и карт данных, ярким примером является карта метро.
- Комбинированная визуализация позволяет объединить несколько сложных графиков в одну схему, как в карте с прогнозом погоды.

Эдвардом Тафти было определено всего 6 базовых принципов представления информации, которые можно применить и к системам виртуальной реальности:

- принцип многомерного представления информации,
- принцип макро- и микро- уровня восприятия информации,
- принцип размещения информации по слоям,
- принцип малых множеств,
- принцип рационального использования цвета,
- принцип пространственно-временной визуализации времени и места [1].

Основная цель данных методов – перенаправить концентрацию от способа представления информации к самой информации. Наиболее распространенными и простыми методами визуализации информации являются графиче-

ки. Они дают возможность визуализировать информацию, используя относительные представления, которые легко воспринимаются человеком. Несмотря на то, что в теории принципы построения визуальных моделей для создания графиков достаточно просты, также определяется грамматика графики (Grammar of Graphics, GoG), т.е. формальная концепция описания любой графической визуализации. Лиланд Уилкинсон предложил концепцию построения стандартизированного и однородного языка для описания любой визуализации, используя стандартные виды примитивов. Основными частями и метаданными графика являются непосредственно данные, которые необходимо визуализировать, необязательный метод плоской или агрегационной проекции данных для их трансформации в относительные числовые значения, которые будут использоваться для построения графика, описание отдельного элемента данных, состоящее из значения, проекции, геометрического представления и метаданных позиционирования, а также описание для присутствующих на графике шкал, вспомогательных линий и связанных с ними координатных систем. Основное преимущество использования формальной системы состоит в легкости описания специфики конкретного вида визуализации. Например, для описания в терминах грамматики графики круговой диаграммы могут использоваться те же принципы, как и при построении столбчатой диаграммы, после чего добавлена проекция визуализируемых данных в полярные координаты [3].

Главная идея представления статистических данных с помощью среды виртуальной реальности, которая предложена в данной работе заключается в том, что, используя очки виртуальной реальности, человек оказывается в комнате с привычными атрибутами из реальной жизни. Он имеет возможность взаимодействовать с трехмерным пространством с помощью различных интерфейсов. Для различного типа информации используется универсальный сценарий. Например, если пользователь хочет узнать о наиболее часто посещаемых странах – ему следует подойти к стенду с фотографиями, а если узнать в какой степени развито машиностроение – к столу с чертежом.

II. ПРАКТИЧЕСКАЯ ЧАСТЬ РАБОТЫ

Основное внимание работы обращено на создание системы виртуальной реальности и настройки ее управления посредством подходящих устройств (очки, манипуляторы и сенсоры). При моделировании системы виртуальной реальности можно выделить две основные задачи:

- моделирование окружения и предметов, т.е. создание пространства, в котором про-

исходит взаимодействие с данными в виде моделей,

- моделирование интерфейсов для управления виртуальной реальностью, т.е. создание объемного меню.

Во время разработки окружения виртуальной реальности использовалась межплатформенная среда разработки Unity [2], которая позволяет создавать трехмерные анимированные сцены, а также настраивать взаимодействие с дополнительными устройствами. На рисунке 1 представлены основные этапы проектирования окружения, а также предоставлены изображения, отражающие данные этапы и сделанные в процессе работы.



Рис. 1 – Основные этапы разработки окружения

На конечном этапе производится расстановка объектов на сцене, добавление физики в виртуальную среду, например гравитацию, для более правдоподобной симуляции реальности, а также добавление скриптов, для контролирования движений пользователя и его взаимодействия с предметами виртуальной реальности.

ЗАКЛЮЧЕНИЕ

Представление статистических данных с использованием виртуальной реальности позволяет наглядно вести лекции и семинары, проводить тренинги, показывать обучающимся все аспекты реального объекта или процесса, что в целом дает колоссальный эффект, улучшает качество и скорость образовательных процессов, и уменьшает их стоимость.

Таким образом, на основе созданного в работе виртуального окружения с поддержкой взаимодействия с очками виртуальной реальности Oculus Rift и манипуляторов Oculus Touch планируется дальнейшая разработка и внедрение предложенной концепции отображения статистической информации. Разрабатываемая система сможет работать как одна, так и совместно с другими курсами или программами и дополнять демонстрации и анализ различных данных.

СПИСОК ЛИТЕРАТУРЫ

1. Tufte E. Visual Display of Quantitative Information. Cheshire, CT: Graphics Press, 2001.
2. Линовес Дж. Виртуальная реальность в Unity. Москва: ДМК, 2016.
3. Wickham H. A layered grammar of graphics // Journal of Computational and Graphical Statistics. 2010. Vol. 19, № 1. P. 3–28. DOI: 10.1198/jcgs.2009.07098.

ШАБЛОНЫ РАСПРЕДЕЛЕНИЯ ОТВЕТСТВЕННОСТИ ПО ЗАПРОСАМ КОМАНД И ИСТОЧНИК СОБЫТИЙ

Кива В. С., Борисов Д. В.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: knijaz@gmail.com, divlboris@gmail.com

В статье описаны достоинства и недостатки шаблона сегрегация ответственности по запросам команд и шаблона источник событий. Эти шаблоны используются для решения таких проблем как: проблема масштабируемости всего приложения и проблема индивидуальной масштабируемости сторон приложения использующихся как для записи, так и для чтения в системах обработки информации.

ВВЕДЕНИЕ

CQRS (сегрегация ответственности запросов и команд) – это очень простой шаблон. Он является наследником шаблона CQS (разделение командных запросов), который был разработан Бертраном Мейером. В соответствии с этим шаблоном методы в системе должны быть разделены на две группы: команды, изменяющие состояние, и запросы, возвращающие значение. Применение этой концепции к объектам или компонентам вводит новую концепцию – CQRS, представленную Грегом Янгом. Основная идея заключается в том, что классы или компоненты внутри приложения, меняющие состояние (команды), должны быть отделены от компонентов, которые получают состояние приложения (запросы).[1]

Основной подход, который люди используют для взаимодействия с информационной системой, состоит в том, чтобы рассматривать ее как хранилище данных CRUD (создание, чтение, модификация, удаление). Под этим подразумевается, что у нас есть модель некоторой структуры записей, где мы можем создавать новые записи, читать записи, обновлять существующие записи и удалять записи. В простейшем случае все наши взаимодействия связаны с хранением и извлечением этих записей.

По мере того, как потребности бизнес функций растут, постепенно происходит отдаление от этой модели. Может потребоваться взглянуть на информацию в том виде, в котором она не хранится в хранилище записей. Возможно требуется сложить несколько записей в одну или сформировать виртуальные записи путем объединения информации из разных мест. Что касается обновления, то появляются бизнес-функции проверки, которые позволяют хранить только определенные комбинации данных, или подтвердить, что данные сохранились.

Изменение, которое вводит CQRS, состоит в том, чтобы разделить эту концептуальную модель на отдельные модели для обновления и отображения, которые соответственно называются команда и запрос. Логическое обоснование этого изменения состоит в том, что для

многих проблем наличие одинаковой концептуальной модели для команд и запросов приводит к более сложной модели, которую становится сложно поддерживать и изменять.

I. Основные понятия CQRS

К основным понятиям CQRS относятся: команды, командная шина, обработчик команд, шина событий, обработчик событий и запросы.

Команды представляют намерение пользователя. Они содержат всю необходимую информацию о действиях, которые пользователь хотел бы выполнить.

Командная шина – это тип очереди, которая получает команды и передает их обработчикам команд.

Обработчики команд содержат актуальную бизнес-логику, которая проверяет и обрабатывает данные, полученные в командах. Обработчики команд отвечают за генерацию и распространение событий домена на шину событий.

Шина событий отправляет события обработчикам событий, подписанным на определенные типы событий. Шина событий может распространять события как асинхронно, так и синхронно.

Обработчики событий отвечают за обработку определенных входящих событий. Их роль заключается в том, чтобы сохранять новое состояние приложения в репозитории чтения и выполнять действия терминала, такие как отправка электронных писем, хранение файлов и т. д.

Запросы – это объекты, которые представляют фактическое состояние приложения, доступное для пользователя. Получение данных для пользовательского интерфейса должно осуществляться через эти объекты.

II. Источник событий

Источник событий (Event Sourcing) – это специализированный шаблон для хранения данных. Вместо сохранения текущего состояния объекта каждое изменение состояния сохраняется как отдельное событие, которое имеет смысл для бизнес-пользователя.[2] Текущее состояние

рассчитывается путем применения всех событий, которые изменили состояние объекта. С точки зрения CQRS, сохраненные события – это результаты выполнения команды для агрегата на этапе записи. Сторона чтения может обрабатывать события которые передаёт хранилище событий и создавать целевые наборы данных, необходимые для запросов. Ключевые причины для реализации этого шаблона:

- Сохраненные события становятся полным перечнем контрольных изменений, внесенных в приложение. По этой причине источник событий традиционно реализуется приложениями с серьезными потребностями аудита. Это также облегчает отладку;
- Поскольку каждое событие сохраняется, можно создавать отчеты в любое время. Эти отчеты могут даже показывать поэтапно, что произошло в определенный момент времени в прошлом. Это облегчает применение специальной аналитики к данным, в том числе с использованием алгоритмов искусственного интеллекта. По тем же причинам можно гибко изменять сторону чтения приложения, а также интегрироваться с новыми внешними системами;
- Так как получение событий означает только добавление новых данных в хранилище, запись выполняется максимально быстро.

Как видно из перечня, Event Sourcing и CQRS очень тесно связаны и действительно поддерживают друг друга.

III. ПРЕИМУЩЕСТВА

Можно выделить много преимуществ подхода CQRS, включая следующие:

- Разделение задач по разработке между более опытным персоналом, которые будут работать над бизнес-логикой, и теми, кто будет работать над запросами. Однако, это преимущество может повредить передаче знаний;
- Возможность добиться высокой производительности операций чтения / записи, масштабируя команды и запросы на нескольких разных серверах;[3]
- Использование двух разных репозиторий (чтение / запись), которые синхронизированы, дает автоматическое резервное копирование без каких-либо дополнительных усилий;
- Операции чтения не нагружают базу данных используемую для записи, поэтому они могут работать быстрее при использовании источника событий;
- Структурирование считанных данных непосредственно для представлений, что упрощает представления и повышает производительность.

IV. НЕДОСТАТКИ

У шаблонов распределения ответственности по запросам команд и источник событий есть ряд недостатков:

- Сложность: Вся система с каждым компонентом, основанным на источнике событий, делает взаимодействие между ними сложным и трудным для чтения, если будет не достаточно проработан каждый компонент системы. Если все функциональные возможности влияют на один и тот же источник событий, он быстро превратится в монолитный источник событий. В целом, эти шаблоны добавляют значительную сложность, следует учитывать, стоит ли оно того;
- Повышенное использование дискового пространства: хранилище событий может в конечном итоге использовать много дискового пространства для хранения событий.[4]

V. ВЫВОД

Источник событий выгодно использовать в программных системах, где бизнес может извлечь выгоду из истории событий, которые произошли. Одним из ключевых преимуществ источника событий является расширяемость программных систем – когда необходимо добавить новый компонент отчетности, можно просто воспроизвести на нем исторические события и запустить его. Это очень выгодно в сине-зеленых развертываниях, когда необходимо чтобы приложения не простаивало.

Интеграция с внешними системами может быть выполнена с использованием событий. В таких сценариях события – это наш программный интерфейс приложения, и внешняя система должна их понимать. Конечно, мы должны публиковать только те события, которые являются публично важными.

СПИСОК ЛИТЕРАТУРЫ

1. A website on building software effectively. [Электронный ресурс] / CQRS – Режим доступа: <https://martinfowler.com/bliki/CQRS.html>. – Дата доступа: 06.10.2019.
2. Framework and server for event-driven microservices focused on CQRS and Event Sourcing. [Электронный ресурс] / Event Sourcing with Axon – Режим доступа: <https://axoniq.io/resources/event-sourcing>. – Дата доступа: 06.10.2019.
3. RisingStack, Full Stack Javascript, Node.js Development and Consulting [Электронный ресурс] / When to use CQRS?! – Режим доступа: <https://community.risingstack.com/when-to-use-cqrs>. – Дата доступа: 05.10.2019.
4. Benefits and drawbacks of using CQRS and Event Sourcing patterns on real life example. [Электронный ресурс] / CQRS and Event Sourcing as an antidote for problems with retrieving application states – Режим доступа: <https://www.nexocode.com/blog/posts/cqrs-and-event-sourcing>. – Дата доступа: 06.10.2019.

ЭВРИСТИКА КАК СРЕДСТВО ДЛЯ ОПТИМИЗАЦИИ ТРАНСПОРТНЫХ ПОТОКОВ ПРЕДПРИЯТИЙ

Козарь Р. В., Навроцкий А. А.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: pozitr0n.kozarroman@gmail.com, navrotsky@bsuir.by

Рассматриваются вопросы использования эвристических алгоритмов в задачах оптимизации.

ВВЕДЕНИЕ

В настоящее время большинство предприятий занимаются решением вопросов, связанных с транспортной логистикой. Как правило, для построения своей логистической системы, предприятия пользуются услугами компаний, специализирующихся на решении задач по управлению процессами перевозок. Если предприятие предполагает использовать только собственную транспортную базу, не привлекая сторонних перевозчиков, оно может столкнуться с рядом проблем, таких, как изношенность подвижного состава, сложности построения маршрутов перевозки, сложности при организации взаимодействия различных видов транспорта и т.д. В любом случае описанные выше проблемы приводят к снижению объемов поставок, и, как следствие, к снижению эффективности работы предприятия в целом. Поэтому зачастую стоит вопрос оптимизации существующих бизнес-процессов предприятия, один из которых, вопрос быстрого создания и построения оптимального маршрута доставки. Создание оптимизированных маршрутов позволяет точно определить объем перевозок грузов со снабженческо-сбытовых предприятий, количество автомобилей, осуществляющих эти перевозки, способствует сокращению простоя автомобилей под загрузкой и разгрузкой, эффективному использованию подвижного состава и высвобождению из сфер обращения значительных материальных ресурсов потребителей. Вместе с тем планирование перевозок позволяет повысить производительность автомобилей при одновременном снижении количества подвижного состава, поступающего на предприятие притом же объеме перевозок. Если созданы оптимальные маршруты и соблюдаются сроки поставки, то производственные запасы потребителей могут сокращаться в 1,5-2 раза, снижая тем самым затраты на складирование. Необходимость маршрутизации перевозок грузов обосновывается еще и тем, что маршруты дают возможность составления проектов текущих планов и оперативных заявок на транспорт, исходящих из действительных объемов перевозок. Таким образом, разработка эффективных маршрутов и проектов планов перевозок способствует своевременному и бесперебойному выполнению

поставок продукции и эффективному взаимодействию организаций-поставщиков, организаций-получателей и автотранспортных организаций.

I. ПОСТАНОВКА ЗАДАЧИ

Таким образом, разработка эффективных маршрутов и проектов планов перевозок способствует своевременному и бесперебойному выполнению поставок продукции и эффективному взаимодействию организаций-поставщиков, организаций-получателей и автотранспортных организаций. Подводя итог вышесказанному можно с уверенностью сказать, что задача оптимизации маршрутизации транспортных средств становится особо актуальной в условиях данной экономической ситуации. Так как имеется большое количество объектов доставки, то необходимо оптимизировать маршруты перевозок и оперативно реагировать на все изменения. Следовательно, можно определить цель оптимизации транспортных перевозок: разработать алгоритм оптимизации грузоперевозок с учетом временных окон и грузоподъемности транспортных средств. Для достижения поставленной цели необходимо решить следующие основные задачи: 1. Сформулировать математическую постановку задачи маршрутизации транспорта с условием ограничений; 2. Выбрать (разработать) критерии оптимизации; 3. Разработать модифицированный генетический алгоритм оптимизации грузоперевозок с учетом временных окон и грузоподъемности транспортных средств; 4. Экспериментально проверить эффективность работы алгоритма.

II. МАТЕМАТИЧЕСКОЕ ОБОСНОВАНИЕ ЭВРИСТИЧЕСКОГО АЛГОРИТМА 2-ОРТ

Поведение алгоритма зависит от следующих факторов:

1. Структуры окрестностей N . Размер окрестности любого решения должен выбираться на основе компромисса между целью получения хорошего улучшения при каждом переходе к новому решению и целью ограничения времени просмотра одной окрестности. Обычно, для любого решения s , окрестность $N(s)$ порождается с помощью некоторой операции локального изменения

s (применительно к текущей ситуации – это абсолютно непредсказуемое изменение дорожной обстановки).

2. Начального решения s_0 . Его можно находить с помощью любого алгоритма (например, конструктивной эвристики), который выдает допустимое решение, или с помощью процедуры случайной генерации значений.
3. Стратегии выбора новых решений. Например, просмотр всех решений из $N(s)$ и выбор наилучшего. Если найденное решение не является оптимальным для локальной области, то осуществляется переход к наилучшему соседу, или к первому лучшему решению, найденному в окрестности. В эвристиках локального поиска используются окрестности, определяемые с помощью последовательностей ограниченной длины операций локального обмена [2]. Пусть, при фиксированном целом $k > 0$, решается этим методом индивидуальная задача проблемы P. Решение u находится в k -обменной окрестности s , если из решения s можно получить решение u , применив не более чем k операций локального обмена. Эвристики, которые основаны на k -обменных окрестностях, часто называют k -оптимальными (k -opt) эвристиками.

III. ДЕТАЛЬНОЕ ОПИСАНИЕ АЛГОРИТМА

Как уже было сказано ранее, новый алгоритм основан на эвристическом алгоритме локального поиска (локального спуска) 3-обменной окрестности вместо 2-обменной окрестности. Данная замена позволяет алгоритму не останавливаться в точке локального оптимума, как это предусмотрено в алгоритме 2-opt, а путешествовать от одного локального оптимума к другому для того, чтобы найти среди всех обходов глобальный оптимум. Для осуществления данного процесса был разработан механизм, позволяющий алгоритму осуществлять выход из локального оптимума (список исключений $List_l(i_k)$). Он строится по истории поиска, т.е. по нескольким последним точкам $i_k, i_{k-1}, \dots, i_{k-l+1}$ и запрещает исследовать часть окрестности $N(i_k)$ текущего решения i_k . Таким образом на каждом шаге алгоритма очередная точка i_{k+1} является оптимальным решением следующей подзадачи: $m(i_{k+1}) = \min\{m(j) | j \in N(i_k) \setminus List_l(i_k)\}$ при условии, что $List_l(i_k) \subseteq N(i_k)$.

IV. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Проведен тест трех алгоритмов. В качестве среды для анализа был использован Matlab. В качестве функций для тестирования были взяты логарифмическая и линейная функции с различными коэффициентами:

– 1-opt 1-окрестности;

– 2-opt 2-окрестности;

– 2-opt 3-окрестности(новый алгоритм).

Результаты теста представлены на рисунке

1.

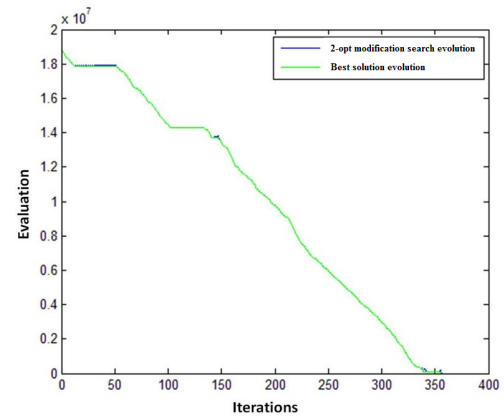


Рис. 1 – Результаты теста

Из данных этого графика видно, что производительность 1-окрестного алгоритма весьма низкая, поскольку его временные затраты очень высоки. Данные 2-окрестного алгоритма несколько лучше, но также достаточно затратные. Данные 3-окрестного алгоритма намного лучше, чем 1-окрестный и 2-окрестный алгоритмы, что показывает шкала временных затрат. Из поставленных экспериментов можно судить, что предложенный в работе алгоритм 2-opt с количеством окрестностей равным 3, имеет более высокую производительность и более высокое качество полученного результата.

V. ВЫВОДЫ

В работе представлено краткое описание и математическое обоснование нового эвристического алгоритма поиска оптимального маршрута.

СПИСОК ЛИТЕРАТУРЫ

1. <https://habrahabr.ru/post/119158/> - Документация по эвристическим и метаэвристическим алгоритмам
2. <https://www.recyclebin.ru/ВМК/II/ii.html> - Использование эвристических алгоритмов в ИИ
3. Ахо, А. В. Введение в структуры данных и алгоритмы / А. В. Ахо, Д. Э. Хопкрофт, Д. Д. Ульман // М., СПб., Киев: «Вильямс», 2016. – 370 с.
4. <https://dic.academic.ru/dic.nsf/ruwiki/703585> - Математический интернет-портал "Академик"
5. Новосибирский государственный университет. Методы локального поиска для задач перестановок столбцов бинарных матриц. [<https://nsu.ru/xmlui/handle/nsu/2150>] / Новосибирский государственный университет. – Электрон. текстовые дан. – Режим доступа <https://nsu.ru>, свободный.
6. Гудман, С. С. Введение в разработку и анализ алгоритмов / С. С. Гудман // М.: "Мир 2010. – 265 с.
7. Фирма ТЕХНОСФЕРА, Методы локального поиска для дискретных задач размещения. [<http://tekhnosfera.com/metody-lokalnogo-poiska-dlya-diskretnyh-zadach-razmesheniya>] Фирма ТЕХНОСФЕРА. – Электрон. текстовые дан. – Режим доступа <http://tekhnosfera.com>, свободный.

АНАЛИЗ ПРОЦЕССА ПРИНЯТИЯ РЕШЕНИЯ В МОДЕЛИ НЕЙРОСЕТЕВОЙ КЛАССИФИКАЦИИ МЕТОДОМ ЛОКАЛЬНЫХ ЛИНЕЙНЫХ АППРОКСИМАЦИЙ

Курочкин А. В., Садов В. С.

Кафедра интеллектуальных систем, Белорусский государственный университет, Факультет радиофизики и компьютерных технологий

Минск, Республика Беларусь

E-mail: alex.v.kurochkin@gmail.com, sadov@bsu.by

Одним из существенных ограничений нейросетевых моделей для классификации является тот факт, что процесс принятия решений в обученной нейронной сети очень сложно поддается анализу, из-за чего невозможно оценить смысловую корректность результатов и сформировать на их основе новые экспертные знания. Для решения этой проблемы предлагается использовать метод локальных линейных аппроксимаций обученной модели, позволяющий оценить влияние отдельных признаков одного экземпляра данных на полученный результат в понятной форме.

ВВЕДЕНИЕ

Искусственные нейронные сети прямого пространства на сегодняшний день являются одним из самых распространенных инструментов для построения моделей классификации на основе известных данных (обучающей выборки). Относительная простота конфигурации, обширный программный инструментарий и возможность тонкой настройки модели делают нейросетевую классификацию очень эффективным средством выведения зависимостей на основании произвольных данных в табличном виде. При наличии достаточно репрезентативной обучающей выборки алгоритм обучения нейронной сети может подобрать параметры моделей таким образом, чтобы сформировать разнообразные совокупности линейных и нелинейных комбинаций входных признаков, которые в последующем используются для получения итогового результата.

С точки зрения систем поддержки принятия решений нейронные сети часто противопоставляются экспертным системам на базе логического вывода (например, нечеткой логики). В таких экспертных системах процесс принятия решения описывается формально, на основании некоторого описательного представления экспертных знаний, и не учитывает статистические характеристики существующих данных. В то же время, нейросетевые модели не включают в себя представление экспертных знаний, а вместо этого выстраивают собственный процесс принятия решений путём подбора параметров модели в процессе обучения таким образом, чтобы минимизировать ошибку модели на некотором объеме существующих данных. С одной стороны, это позволяет устанавливать сложные зависимости в некотором объеме данных без необходимости их детального смыслового анализа. С другой стороны, обученная нейросетевая модель является «черным ящиком» – процесс принятия

решений описывается комбинацией параметров модели и, как правило, крайне сложен для интерпретации [1].

I. АНАЛИЗ ОБУЧЕННЫХ НЕЙРОСЕТЕВЫХ КЛАССИФИКАТОРОВ

Для интерпретации обученных нейросетевых моделей в первую очередь необходимо определить, какое представление процесса принятия решения может быть интуитивно понятно для восприятия человеком.

Нейросетевая модель M для решения задачи бинарной классификации по совокупности из n вещественных признаков представляет собой параметрическую функцию:

$$M(\vec{x}, \vec{\theta}) : (\mathbb{R}^n \times \Theta) \rightarrow [0; 1], \quad (1)$$

где $\vec{x} \in \mathbb{R}^n$ – n -мерный вещественный вектор значений входных признаков, $\vec{\theta} \in \Theta$ – вектор параметров модели, \mathbb{R} – множество вещественных чисел, Θ – множество значений параметров модели. Как правило $\Theta \subset \mathbb{R}^m$, где m – количество параметров модели [1].

Параметрами модели в нейронной сети прямого распространения являются веса связей между формальными нейронами соседних слоёв. Каждый нейрон реализует некоторую функцию активации, на вход которой подаётся взвешенная сумма выходов нейронов предыдущих слоёв. Увеличение количества нейронов и количества слоёв позволяет нейронной сети оперировать более сложными совокупностями признаков. Интуитивно ожидается, что в процессе обучения нейронная сеть сможет выделить такие совокупности значений признаков, которые совместно предоставляют более сложное абстрактное описание входных данных. Например, в задаче распознавания символов печатного шрифта при использовании значений пикселей как входных параметров один из нейронов второго слоя

может активизироваться по совокупности значений горизонтальных пикселей, что соответствует абстрактному описанию «горизонтальная черта в центре»; в дальнейшем, такой нейрон может иметь значительный вес при связи с выходными нейронами, соответствующими буквам «А», «В», «Е», «F», «H», «P», «R». В то же время, для принятия решения могут использоваться более сложные нелинейные совокупности признаков, которые не будут иметь простой интерпретации [2].

II. ПОСТРОЕНИЕ ИНТЕРПРЕТАЦИЙ НА ОСНОВЕ ЛИНЕЙНЫХ АППРОКСИМАЦИЙ

В процессе анализа установлено, что наиболее универсальными и интуитивно понятными являются интерпретации процесса принятия решения на основе порогового значения одного из признаков, т.е. в виде линейной модели по одному из входных значений [3]. Таким образом, для некоторой совокупности входных признаков $\vec{x}^* \in \mathbb{R}^n$ необходимо построить набор из n взвешенных описаний, каждое из которых будет характеризовать линейное локальное поведение нейросетевой модели в некоторой окрестности совокупности \vec{x}^* по каждому из входных признаков.

Для построения таких описаний для элемента \vec{x}^* осуществляется анализ пространственной окрестности полученной модели на некотором расстоянии. Для этой окрестности осуществляется сэмплирование – выборочная генерация некоторого количества точек, которые выступают в качестве небольшой локальной обучающей выборки. Для этой выборки методом линейной регрессии строится линейная модель $m[\vec{x}^*, \theta^*] : (\mathbb{R}^n) \rightarrow [0; 1]$. Параметры этой модели θ^* позволяют в n -мерном пространстве входных признаков построить гиперплоскость, которая разделяет исходное пространство на 2 части. При этом для каждого из n признаков можно рассмотреть проекцию этой гиперплоскости на ось, соответствующую этому признаку. Точка пересечения и угол наклона определяют, соответственно, пороговое значение и его влияние на конечный результат. Если гиперплоскость перпендикулярна оси признака, то этот признак оказывает определяющее влияние, поскольку принятие решения осуществляется только на основании его порогового значения. Если гиперплоскость параллельна оси признака или проходит под очень малым углом наклона, влияние этого признака на результат минимально. Таким образом, для определения влияния по углу наклона гиперплоскости относительно оси может использоваться косинус.

Другими словами, для совокупности из значений n признаков $\vec{x}^* = \{x_i^*\}$ и некоторой обу-

ченной классификационной нейросетевой модели $y = M(\vec{x}), y \in [0; 1]$ линейные аппроксимации строятся следующим образом: по окрестности точки \vec{x}^* в соответствии с выходом модели строится линейная модель, задающая некоторую гиперплоскость T^* . При проекции гиперплоскости T^* на оси признаков x_i^* можно определить точки пересечения \bar{x}_i^* , а также углы наклона φ_i^* .

Полученные значения могут быть интерпретированы следующим образом: косинус угла наклона $\cos \varphi_i^*$ показывает, насколько на полученный выходной результат $y^* = M(\vec{x}^*)$ влияет тот факт, что i -й признак x_i принял значение, большее или меньшее, чем пороговое \bar{x}_i^* . Таким образом, для любого входного значения и любого признака может быть получен набор линейных аппроксимаций, которые могут использоваться для оценки влияния конкретного значения признака на выходной результат. Кроме того, пороговое значение \bar{x}_i^* для этого признака отражает, что анализируемая модель будет иметь схожее поведение в ближайшей окрестности при изменении значения i -го признака в пределах от x_i^* до \bar{x}_i^* .

Полученные линейные аппроксимации позволяют по линеаризации модели в окрестности исследуемой точки понять, какие именно признаки оказали наибольшее влияние на полученное значение.

ЗАКЛЮЧЕНИЕ

В работе представлен метод линейных аппроксимаций, позволяющий анализировать поведение сложных нейросетевых классификационных моделей. Хотя такой подход не предоставляет полное описание процесса принятия решения, проводимый анализ позволяет, для некоторой совокупности входных признаков, получить более интуитивное представление о том, какие из этих признаков являлись определяющими в вычислении итогового значения по этой модели. Полученные результаты могут использоваться для формирования новых экспертных знаний, проверки корректности модели в граничных случаях, а также для экспертной оценки состоятельности полученной модели.

СПИСОК ЛИТЕРАТУРЫ

1. Goodfellow, I. Deep learning / I. Goodfellow, B. Yoshua, C. Aaron. – Cambridge, MA: The MIT Press, 2016. – 775 p.
2. Ioannou, Y. Decision Forests, Convolutional Networks and the Models in-Between / Y. Ioannou, D. Robertson, D. Zikic, et. al. // arXiv:1603.01250[cs] [Электронный ресурс]. – 2016. – Режим доступа: <https://arxiv.org/abs/1603.01250> – Дата доступа: 08.10.2019.
3. Lundberg, S. M. A Unified Approach to Interpreting Model Predictions / S. M. Lundberg, S.-I. Lee // Advances in Neural Information Processing Systems. – 2017. – № 30. – pp. 4765–4774.

ТАБЛИЦЫ СМЕРТНОСТИ НА ПОЖАРЕ КАК ОСНОВА ДЛЯ ТРАНСФОРМАЦИИ МЕТОДИКИ ОЦЕНКИ СТОИМОСТИ ЧЕЛОВЕЧЕСКОЙ ЖИЗНИ

Лапицкая Н. В., Игнатюк Т. А.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Научно-исследовательский институт пожарной безопасности и проблем чрезвычайных ситуаций

МЧС Беларуси

Минск, Республика Беларусь

E-mail: lapan@bsuir.by, tansmile7@gmail.com

Важной составляющей государственного богатства являются человеческие ресурсы. Человеческий потенциал реализуется посредством человеческого капитала, выступая источником дохода для своего носителя, экономического роста и социального прогресса – для общества. Для понимания феномена человеческого потенциала очень важна оценка качества социально-экономической жизни и существующих экономических условий для формирования и реализации потенциала человека в трудовой или иной общественно-признаваемой деятельности, в связи с чем возникает вопрос о необходимости стоимостной оценки человеческой жизни. Трансформация методики стоимостной оценки человеческой жизни требует научного и методического решения, а так же практической реализации.

Стоимостью человеческой жизни называется условное понятие, под которым подразумевается экономический эквивалент, а точнее, возмещение, которое должен получить человек или его родственники при потере жизни или здоровья в результате несчастного случая. Таким образом, эти показатели определяют ценность человеческой жизни с точки зрения труда каждого индивида в создании валового общественного продукта, индивидуальной полезности человека для общества [1].

На текущий момент отсутствует единый установленный законами или подзаконными актами порядок выплаты компенсаций семьям погибших при чрезвычайных происшествиях, и в зависимости от обстоятельств таких происшествий вступают в силу разные нормативные акты [2]. Компенсационные выплаты в связи с потерей жизни при чрезвычайных обстоятельствах играют важную роль в морально-этическом, экономическом, социальном и правовом отношениях. В известной степени это определяется различными масштабами и общественным резонансом, который вызвала гибель людей.

В Республике Беларусь основным источником компенсационных выплат являются бюджетные средства, в то время как за рубежом источник – это либо страховые компании, либо компании, несущие ответственность за инцидент.

В то время как в странах Западной Европы доля индивидуального добровольного страхования жизни превышает половину общего объема страховых взносов, в нашей стране институты страхования жизни находятся в процессе становления [3].

Характерной чертой постсоветского пространства, согласно статистическим данным, является значительная доля гибели людей на пожарах в жилом фонде. С целью повышения адекватности существующих методик оценки стоимости жизни был проведен анализ подходов, выделяющих факторы, существенно влияющих на качество этой оценки.

Существенность показателя возрастной структуры для оценки стоимости жизни обосновывает целесообразность построения прототипов таблиц дожития (смертности), рассматривающих только гибели людей на пожарах. На основе данных о количестве погибших в период с 2009 по 2018 гг. в Республике Беларусь, была составлена и проанализирована таблица распределение погибших за 10 лет по каждой причине пожаров. Результаты распределения данных о гибели в зависимости от причин возникновения пожара представлены на рисунках 1 и 2.



Рис. 1 – Распределение погибших по причинам пожаров 2009 - 2018 гг.



Рис. 2 – Распределение погибших по другим причинам из рисунка 1.

На рисунке 3 представлены результаты распределения данных о количестве погибших за последние 10 лет в Республике Беларусь по возрасту от 0 до 100 лет по пятилетним интервалам.

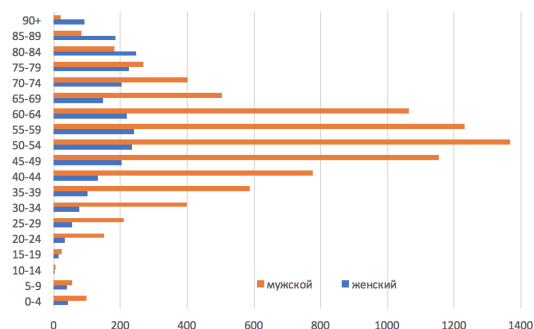


Рис. 3 – Распределение погибших по возрасту 2009-2018 гг.

На основе полученных результатов таблицы распределения погибших на пожарах за 10 лет в период с 2009 по 2018 гг. составлена таблица дожития (смертности).

В таблице 1 представлена краткая таблица дожития (смертности), с длиной возрастного интервала $n=5$. Первая колонка таблицы характеризует точное число полных лет, прошедших с момента рождения x , измеряется в годах. Начальный возраст в таблице – 0 лет (момент рождения), конечный – w (100 лет), возрасту, к которому вымирает практически вся совокупность родившихся.

Таблица 1 Краткая таблица дожития

Возраст X	Число погибших d_x	Число доживших l_x	Вероятность смерти q_x	Вероятность дожития p_x	Среднее число живущих L_x	Среднее число человеко-лет T_x	Ожидаемая продолжительность жизни e_x
0	9	100000	0,0001	1,000	99995,5	9261282	92,613
5	40	99859	0,0004	1,000	99839	8761586,5	87,740
10	2	99765	0,0000	1,000	99764	8262583,5	82,820
15	9	99757	0,0001	1,000	99752,5	7863538,5	78,827
20	26	99718	0,0003	1,000	99705	7364842	73,857
25	53	99534	0,0005	0,999	99507,5	6866662	68,988
30	83	99270	0,0008	0,999	99228,5	6369621	64,165
35	114	98797	0,0012	0,999	98740	5874355,5	59,459
40	183	98110	0,0019	0,998	98018,5	5382014	54,857
45	265	97203	0,0027	0,997	97070,5	4893723,5	50,345
50	346	95846	0,0036	0,996	95673	4411011	46,022
55	328	94243	0,0035	0,997	94079	3935792,5	41,762
60	307	92772	0,0033	0,997	92618,5	3468359	37,386
65	171	91488	0,0019	0,998	91402,5	3008020	32,879
70	115	90836	0,0013	0,999	90778,5	2552351	28,098
75	105	90233	0,0012	0,999	90180,5	2099676,5	23,269
80	83	89742	0,0009	0,999	89700,5	1649700	18,383
85	73	89312	0,0008	0,999	89275,5	1202047	13,459
90	32	89044	0,0004	1,000	89028	756233	8,493
95	11	88955	0,0001	1,000	88949,5	311295,5	3,499
100	1	88932	0,0000	1,000	44466	44466	0,500

Согласно представленным данным 82,3% гибели людей на пожарах происходят по причине неосторожное обращение с огнем, 6,0% - нарушение правил эксплуатации печей, теплогенерирующих агрегатов (устройств), электросетей и электрооборудования. Согласно возрастной структуры жертв пожаров максимальная доля гибели представителей мужского пола приходится на возраст 50-54 года, женского – 80-84, обоих полов – с 10 по 14 лет. Полученные результаты демонстрируют необходимость выделения в качестве классификационных признаков, входящих в методику оценки стоимости человеческой жизни, гендерно-возрастных показателей и определения поправочных коэффициентов в соответствии с показателями вероятности смертности на пожаре, соответствующей возрастной и гендерной группе. Полученные коэффициенты позволяют сформировать первичные оценки, которые будут уточняются факторами причины и места возникновения пожара.

СПИСОК ЛИТЕРАТУРЫ

1. Видяпина, В. И. Экономическая теория (Политэкономия) : учеб. пособие / В. И. Видяпина ; под ред. В. И.Видяпина, Г. П.Журавлевои. – М.: ИНФРА -М, 2008. – 281 с.
2. Об утверждении инструкции о порядке работы республиканских органов государственного управления, местных исполнительных и распорядительных органов, организаций и их комиссий по чрезвычайным ситуациям по определению объемов финансовой поддержки юридическим, физическим лицам и индивидуальным предпринимателям, имуществу которых нанесен ущерб в результате чрезвычайных ситуаций, решении других вопросов, касающихся ликвидации последствий этих ситуаций в пострадавших районах: Постановление Министерства по чрезвычайным ситуациям Республики Беларусь, Министерства Финансов Республики Беларусь, Министерства Экономики Республики Беларусь, 18 августа 2006 г., № 40/276/136 // Национальный правовой Интернет-портал Республики Беларусь. – Режим доступа: <http://pravo.newsby.org/belarus/postanov5/pst824>. – Дата доступа: 03.10.2019.
3. Бычкова, М. А. Состояние и задачи упорядочения выплат компенсаций в связи с потерей жизни / М. А. Бычкова // Контуры глобальных трансформаций: политика, экономика, право. – 2014. – № 7(2). – С. 112–133.

МОДЕЛИРОВАНИЕ ПРОЦЕССА ЛАЗЕРНОЙ ПРОШИВКИ ОТВЕРСТИЙ В КРЕМНИИ ПРИ ФОРМИРОВАНИИ 3D СТРУКТУР

Лаппо А. И., Боброва Т. С., Кузнецова О. В.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {lappo, t.bobrova, ovkuznetsova}@bsuir.by

Математическое моделирование температурно-временных зависимостей и распределение тепловых полей нагрева кремневой пластины при формировании переходных отверстий лазерным источником.

3D электронные модули, которые способны интегрировать множество микросхем на основе различных технологий в один единый корпус, позволяют создать устройств малых размеров, широким набором функций и доступных по стоимости. Одно из современных направлений реализации 3D-модулей является технология TSV (Through Silicon Vias – сквозные отверстия в кремнии) (рис. 1). Применение традиционного сверления при формировании отверстий необходимого диаметра, не всегда соответствует технологическим требованиям уровня качества. Лазерная обработка – один из методов формирования переходных отверстий. При воздействии лазерного излучения на подложку происходит образование отверстия за счет плавления и испарения материала [1].

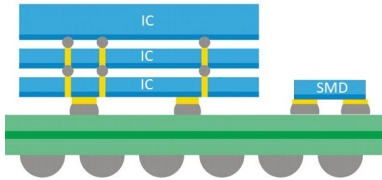


Рис. 1 – Конструкция 3D модуля на основе технологии TSV

I. МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ ЛАЗЕРНОГО НАГРЕВА

Целью математического моделирования является получение зависимостей распределения тепла на поверхности кремневой пластины подверженной лазерному нагреву. Физику и термодинамику лазерного нагрева можно описать следующими математическими зависимостями, приведенными ниже.

Температура поверхности зависит от мощности, поглощенной единицей площади. Тогда энергия поглощенного излучения [2]:

$$E = P \cdot \tau = q \cdot S \cdot \tau(1)$$

где q – плотность мощности; P – мощность введенная через поверхность; S – площадь облучаемой поверхности; τ – длительность воздействия.

Соотношение между распределением температуры по поверхности $T(x,y,z)$ в зависимости

от времени воздействия и плотностью мощности q может быть определено из уравнений теплопроводности типа [2]:

$$\frac{\delta T(x, y, z, t)}{\delta t} - \alpha \cdot \Delta T(x, y, z, t) = \frac{Q(x, y, z, t)}{\rho \cdot c} \quad (2)$$

где Δ – оператор Лапласа, Q – объемная плотность поглощенного светового потока, α – температуропроводность ($\alpha = k/\rho c$, где k – теплопроводность, ρ – плотность, c – теплоемкость), x, y, z, t – аргументы функции.

Объемная плотность поглощенного светового потока с учетом поглощательной способности и направлением по оси z [2]:

$$Q(x, y, z, t) = q_0 \cdot A \cdot e^{-\alpha z} \quad (3)$$

где q_0 – плотность мощности падающего излучения, A – поглощательная способность на волне излучения; α – линейный коэффициент поглощения на волне излучения, m^{-1} ; z – координата отсчитываемая от поверхности в глубину материала. Особенностью задач, связанных с нагревом материалов лазерным излучением является то, что распределение излучения и, следовательно, тепловой источник, как правило, обладает резко изменяющимися пространственно-временными характеристиками. Поэтому идеализация свойств тепловых источников, часто допускаемая в расчетных схемах для уменьшения математических трудностей, может приводить к отклонениям расчетных данных от экспериментальных. [2] Таким образом, функция гауссового импульса позволяет описать профиль излучения лазерного источника и задать нормальное распределение плотности мощности падающего лазерного излучения. Распределение тепла по поверхности будем считать равным во всех направлениях от точки (x_0, y_0) нормального падения луча:

$$Q(x, y, z, t) = q_0 \cdot A \cdot e^{-\alpha z} \cdot \frac{\alpha}{\pi r^2} \cdot e^{-\left(\frac{(x-x_0)^2 + (y-y_0)^2}{2r^2}\right)} \quad (4)$$

где $Q(x, y, z, t)$ – функция, характеризующая распределение плотности поглощения светового потока на поверхности материала и изменение его

во времени, z – ось перпендикулярная поверхности и отсчитываемая от поверхности вглубь материала, r – радиус пятна излучения.

Теплообменом облучаемой поверхности с окружающей средой можно пренебречь. Задаем действие источника на поверхности ($z = 0$) граничным условием [4]:

$$-k \cdot \frac{\delta T(x, y, z, t)}{\delta z} |_{(z=0)} = Q(x, y, t)$$

Если размеры нагреваемого тела достаточно велики ($l_x, l_y, l_z \gg \sqrt{\alpha\tau}$), то его можно считать полубесконечной средой:

$$T_{x\pm\infty} = T_{y\pm\infty} = T_z = T_H, \quad (5)$$

где T – начальная температура ($T_{t=0} = T_H = 293, 15$). Теплообменом облучаемой поверхности с окружающей средой можно пренебречь. Решив уравнение теплопроводности, получаем распределение температуры по поверхности при использовании источника с гауссовым профилем [3]:

$$T(x, y, z, t) = \frac{2p}{\rho c (4\pi\alpha)^{3/2}} \int_0^t \frac{1}{(\tau + t)\sqrt{t}} \cdot e^{-\frac{z^2}{4\alpha t} - \frac{x^2 + y^2}{4\alpha(\tau + t)}} dt \quad (5)$$

где p – мощность излучения, введенная в материал через площадку s ; $\tau = \alpha/4r$; r – радиус пятна излучения; α – коэффициент теплопроводности; c – удельная теплоемкость; ρ – плотность материала; t – текущее время.

II. РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ В СРЕДЕ COMSOL MULTIPHYSICS 5.3A

В качестве источника лазерного нагрева выбран: Lotis ТП LS-2132UTF с длиной волны 1064мкм, энергия импульса 170мДж, длительность импульса 5-6нс, частота импульса 15Гц, диаметр выходного пучка 5мм, мощность – 750Вт.

Моделирование распределения температуры по поверхности кремневой пластины и зависимостей скорости роста температур, выполнено в программном пакете Comsol Multiphysics 5.3a. Для расчета были заданы исходные и граничные условия, и характеристики лазерного источника. Окружающая среда – воздух в нормальных условиях.

В качестве модели использовалась кремневая пластина с диаметром 25.4мм, толщина – 200мкм, влучательная способность поверхности пластины – 0.8, плотность – 2329кг/м³, теплоемкость – 700Дж/(кг*К), температура кипения и испарения – 2623К.

В результате моделирования получены температурно-временные зависимости (рис 2) и тепловые поля на поверхности кремневой пластины (рис 3). По температурно-временным зависимостям видно, что профиль температуры

имеет мгновенный рост до температуры 2000 после чего, в течении 3 секунд температура достигает своего пика. При продолжении процесса происходит нагрев всего объема кремневой пластины.

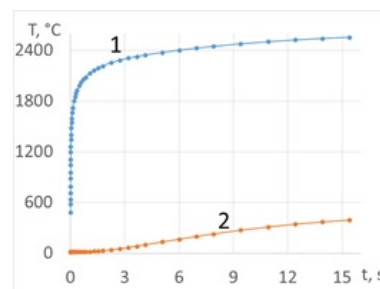


Рис. 2 – Температурно-временные зависимости на поверхности кремневых подложек, 1 – максимальная температура в точке воздействия, 2 – минимальная температура кремневой пластины

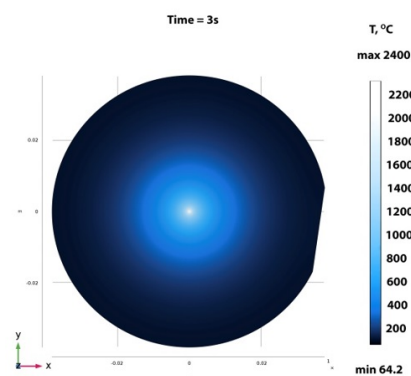


Рис. 3 – Распределение тепловых полей на поверхности кремневой пластины

ЗАКЛЮЧЕНИЕ

По результатам распределение тепловых полей на поверхности кремневой пластины, видно, что для получения искомой температуры испарения кремния необходимой для формирования переходного отверстия, составляющей 2400°С для лазера потребовалось 3 секунды. Так же выявлено что с ростом продолжительности процесса растет температура всей пластины, что может негативно сказаться на качестве формируемых отверстий из-за формирования напылов на большей площади от точки воздействия.

1. Васильев А. Современные технологии 3D интеграции // Компоненты и технологии, 2010. – №1. – С. 156–158.
2. COMSOL Multiphysics ПО для мультифизического моделирования [Электронный ресурс]. – 2019. – Режим доступа: <https://www.comsol.ru/>
3. Либенсон М.Н., Яковлев Е.Б., Шандыбина Г.Д. Взаимодействие лазерного излучения с веществом (силовая оптика). Часть II. Лазерный нагрев и разрушение материалов. Учебное пособие. Под общей редакцией В.П. Вейко – СПб: НИУ ИТМО, 2014. –181с.
4. Вейко В.П., Шахно Е.А. Сборник задач по лазерным технологиям. Изд. 3-е, испр. и дополн. – СПб: СПбГУ ИТМО, 2007. – 67 с.

СОВРЕМЕННАЯ ВЫСОКОНАГРУЖЕННАЯ И МАСШТАБИРУЕМАЯ СИСТЕМА

Ляхор Т. В., Проволоцкий В. Н., Трофимович А. Ф., Захарьев В. А.
Кафедра систем управления. Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: linoge@bsuir.by

ВВЕДЕНИЕ

В настоящее время возникает необходимость в разработке высоконагруженных программных систем, предназначенных для одновременной работы десятков тысяч пользователей и способных масштабироваться при необходимости. Одной из таких систем является программное обеспечение интернет-переписи населения Республики Беларусь (ПО И-перепись).

I. ФУНКЦИОНАЛЬНЫЕ МОДУЛИ СИСТЕМЫ

ПО И-переписи состоит из функциональных модулей изображенных на рисунке 1.



Рис. 1 – Перечень функциональных модулей ПО И-переписи

- Функциональный модуль «Авторизация» предназначен для обеспечения процессов идентификации, аутентификации и авторизации пользователя в ПО И-переписи.;
- В ходе авторизации респондентов модуль подключает подмодуль «Взаимодействие с МСИ» с целью получения подтверждения того, что пользователь зарегистрирован в МСИ;
- Модуль «Сбор событий» используется в режиме администрирования и обеспечивает управление отображением загрузки серверов ПО И-переписи, предоставление журнала отправки данных по заполненным респондентами переписным листам в ПСПД, аудит событий безопасности и ошибок функционирования системы;
- Модуль «Взаимодействие с редакторами» используется в режиме администрирования и обеспечивает управление информационным контентом, сопровождающим процесс переписи населения;
- Модуль «Взаимодействие с респондентами» используется в пользовательском ре-

жиме и предоставляет графический инструмент для заполнения переписных листов, а также для просмотра информационного контента, сопровождающего проведение переписи населения в ПО И-переписи;

- Модуль «Администрирование» используется в режиме администрирования и обеспечивает выполнение настройки параметров системы;

II. АРХИТЕКТУРА СИСТЕМЫ

Ранее для клиент-серверных систем стандартом является трёхзвенная архитектура. Она представляет собой слой представления (браузер), слой бизнес-логики (приложение на сервере) и слой хранения данных (база данных). Однако в современных системах из-за более высоких требований к гибкости пользовательского интерфейса и усложнению слоя представления, часть бизнес логики может выноситься на сторону клиента и работать в браузере. Данные приложения называются SPA (single page application) и реализовываются на языке javascript, который выполняется в браузере у конечного пользователя. Данный подход позволяет снизить нагрузку на сервер приложений, так как страница теперь полностью формируется на стороне клиента. Также такой подход ведёт к уменьшению передаваемого трафика между клиентом и сервером и позволяет инженеру отвечающему за бизнес-логику сосредоточиться только на ней.

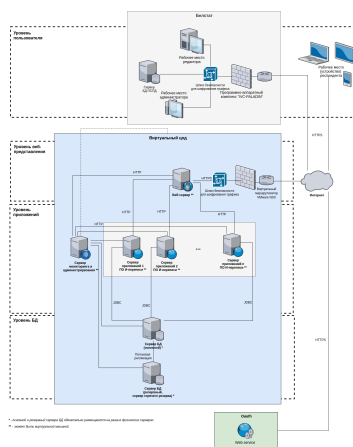


Рис. 2 – Схема комплекса технических средств

Кроме того, для работы приложения необходимы сторонние интеграции, такие как меж-

банковская система идентификации (МСИ) и подсистема сбора и передачи данных (ПСПД). МСИ необходима для авторизации и аутентификации человека, по личному идентификационному номеру и номеру мобильного телефона она позволяет аутентифицировать человека и получить данные о нём. ПСПД является системой предназначенной для осуществления переписи людей с помощью планшетов.

Основная задача ПО И-переписи это заполнения переписного листа с последующим его преобразованием к формату ПСПД и отправкой в данную систему. Для реализации клиентской части был выбран фреймворк Angular. Полученный в результате компиляции и сборки файл кладётся на web-server Nginx, который при обращении к нему пользователя отдаёт данный файл. После скачивания полученных файлов, браузер пользователя отрисовывает пользовательский интерфейс и отправляет http запросы за получением необходимых данных на web сервер.

Запросы для получения данных Nginx перенаправляет на один из шести серверов приложений, являясь в свою очередь балансировщиком нагрузки. Приложения на сервере написаны на языке Java с использованием фреймворка Spring Boot. Работа приложений заключается в сохранении и загрузке анкет, кэшировании и поиске элементов справочников, а также интеграций с МСИ и ПСПД.

В качестве слоя хранения данных используется база данных Postgres. Для надёжности хранения данных, одновременно работает 2 базы

в режиме master-slave с включенной асинхронной репликацией. В случае отказа основной базы, приложение будет работать со slave.

Для хранения пользовательских сессий и их передачи между серверами используется Hazelcast (рисунок 3).

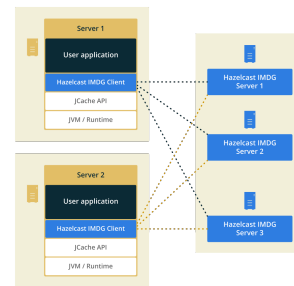


Рис. 3 – Структура работы с Hazelcast сервером

Hazelcast - это распределённое хранилище в памяти, которое позволяет быстро обращаться к элементам по ключу. Он используется для масштабирования и увеличения скорости доступа к оперативным данным, когда скорости работы с реляционными хранилищами уже не достаточно.

СПИСОК ЛИТЕРАТУРЫ

1. Мартин Клеппман. Высоконагруженные приложения. Программирование, масштабирование, поддержка. СПб., Питер, 2018. - 640 с.
2. Брэндан Бёрнс. Распределенные системы. Паттерны проектирования. СПб., Питер, 2019. - 224 с.
3. Ли Атчинсон. Масштабирование приложений. Выращивание сложных систем. СПб., Питер, 2018. - 256 с.

О ПРИМЕНЕНИИ СТАТИСТИЧЕСКИХ МЕТОДОВ В СТЕГАНОАНАЛИЗЕ ГРАФИЧЕСКИХ ИЗОБРАЖЕНИЙ

Лобач С. В., Меркулов Р. И., Акулич В. Н.
Кафедра математического моделирования и анализа данных,
Белорусский государственный университет
Минск, Республика Беларусь
E-mail: lobachS@bsu.by, {merkylovecom, fizik2009}@mail.ru

В статье рассматриваются методы стеганографии, встраивание информации в изображения, проводится сравнительный анализ статистических методов стеганоанализа.

ВВЕДЕНИЕ

Основная цель стеганоанализа [1] состоит в том, чтобы обнаружить факт наличия скрытого сообщения в модифицированном контейнере: аудиофайле, графическом изображении, видео-последовательности (пассивный стеганоанализ) — и, если возможно, извлечь, вскрыть, подменить или уничтожить скрытое сообщение (активный стеганоанализ). В стеганоанализе можно разработать эффективный алгоритм обнаружения и извлечения сообщения из модифицированного контейнера в том случае, когда известен алгоритм встраивания информации. Но, как правило, алгоритм сокрытия сообщения в контейнере неизвестен. Поэтому основная задача состоит в разработке таких алгоритмов, которые были бы достаточно эффективны при проведении стеганоанализа для определенного семейства алгоритмов сокрытия информации.

Можно выделить несколько основных направлений развития стеганоанализа [1]:

- стеганоанализ на основе теории статистического распознавания образов;
- параметрический статистический стеганоанализ;
- стеганоанализ, основанный на идентификации скрытого сообщения.

Первое направление (известное как слепой стеганоанализ) основано на использовании статистических методов классификации с обучением.

Второе направление стеганоанализа предполагает известные параметрические модели, которыми описываются контейнер, модифицированный контейнер и скрываемое сообщение.

В третьем направлении проблема стеганоанализа рассматривается как проблема идентификации. относительно контейнера и скрываемого сообщения предполагается, что они независимы.

1. СТАТИСТИЧЕСКИЕ МЕТОДЫ СТЕГАНОАНАЛИЗА

В данной статье рассматривается задача пассивного стеганоанализа, т.е. задача обнаружения самого факта наличия скрываемого сообщения в модифицированном контейнере. Применяются статистические методы анализа бинарных последовательностей, которые описывают

исходные изображения. Проводятся компьютерные эксперименты на реальных изображениях, в которых имеются скрытые сообщения.

Оценка числа переходов значений младших бит. Метод основывается на том факте, что между младшими битами соседних элементов, а также между ними и остальными битами в естественных контейнерах имеются корреляционные связи [2]. Так как последовательности битов являются двоичными последовательностями, то анализируется четыре вида перехода: $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 0$, $1 \rightarrow 1$. По полученным результатам строится гистограмма, где каждый столбец соответствует одному из переходов. Приведем пример построения такой гистограммы.

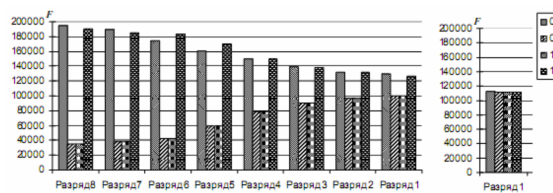


Рис. 1 – Гистограмма частот переходов битовых значений: а – пустого контейнера, б – стегоконтейнера

Для пустого и модифицированного контейнера число переходов в потоке НЗБ (наименее значимых бит) будет разным. Распределение НЗБ модифицированного контейнера имеет, как правило, случайный характер. Пустому контейнеру не свойственно примерно одинаковое число переходов в потоке НЗБ для всех состояний.

Оценка частот появления битовых серий. Метод позволяет оценить равномерность распределения элементов в исследуемой последовательности на основе анализа частоты появления нулей и единиц в серии из k бит [3]. В битовом представлении исследуемой последовательности подсчитывается, сколько раз встречаются нули и единицы ($k = 1$), серии-двойки 00, 01, 10, 11 ($k = 2$), серии-тройки 000, 001, 010, 100, 101, 110, 111 ($k = 3$). На основании этого строится гистограмма. Для незаполненных контейнеров не является характерным, чтобы значения частот всех компонентов находились достаточно близко. При внедрении информации значения частот сближаются. Этот факт используется при стеганоанализе.

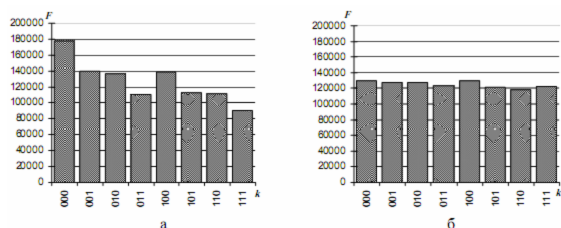


Рис. 2 – Гистограмма частот серии-тройки ($k = 3$) в потоке НЗБ: а – пустого контейнера, б – стежоконтейнера

Для незаполненных JPEG-изображений не является характерным, чтобы значения частот всех компонентов находились достаточно близко (рис. 2 а). При внедрении информации значения частот сближаются (рис. 2 б). Этот факт используется при анализе. Результаты работы метода зависят от стеганографического преобразования, используемого для встраивания скрываемых данных, а также от их объема. Как правило, выявление факта скрытия осуществимо при заполнении контейнера на 60% и выше.

Анализ распределения значений на основе критерия хи-квадрат. В методе используется анализ гистограммы, полученной по элементам изображения и оценка распределения пар значений этой гистограммы [3, 4].

Метод хи-квадрат является универсальным, так как подходит для анализа изображений, созданных различными программами скрытия. Однако результаты работы метода по критерию хи-квадрат в значительной мере зависят от способа скрытия данных. При последовательной записи в НЗБ элементов контейнера метод обеспечивает хорошие результаты (рис. 3), а при псевдослучайном выборе младших бит и рассеивании сообщения по всей длине контейнера метод не срабатывает. Кроме того, существует возможность выбора отдельных областей изображения для их последующего анализа. Такой подход позволяет выявлять наличие информации, скрытой псевдослучайным образом.

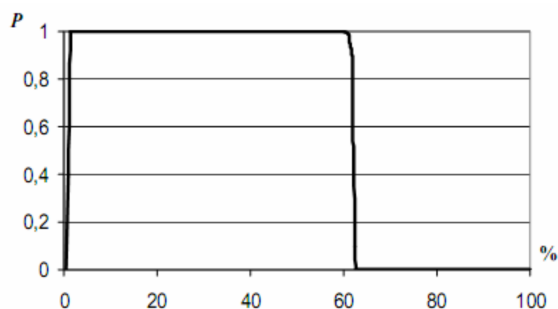


Рис. 3 – Вероятность встраивания по критерию χ^2 при анализе стежоконтейнера, полученного методом последовательной замены

Анализ гистограмм частот элементов изображения. Метод позволяет оценить равномерность распределения элементов анализируемого изображения, а также определить частоту

появления конкретного элемента [5]. Если разброс частот появления элементов в цветовых составляющих изображения стремится к нулю, то контейнер содержит скрытые данные. В противном случае контейнер считается пустым. Для изображений в JPEG-формате строится гистограмма частот квантованных дискретных косинусных коэффициентов. Экспериментально обнаружено, что огибающая гистограммы пустого изображения имеет более гладкий характер (рис. 4 а) по сравнению с гистограммами изображений, содержащими стеганографическое вложение (рис. 4 б). При значительных объемах скрываемой информации гистограммы часто приобретают ступенчатый характер, что нетипично для обычных JPEG-изображений.

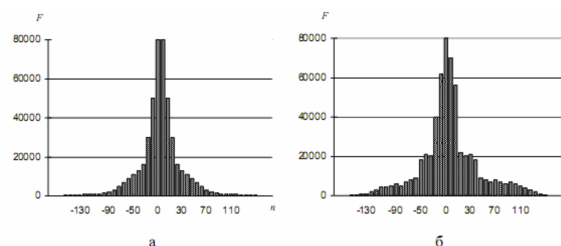


Рис. 4 – Гистограмма частот дискретных косинусных коэффициентов: а – исходного изображения, б – изображения, содержащего скрытую информацию

ЗАКЛЮЧЕНИЕ

В статье приведен краткий обзор стеганографических методов, рассмотрены некоторые статистические методы, применяемые в стеганографии. Статистические методы не являются средством, позволяющим со 100% надежностью определять наличие скрытой информации. Они дают возможность аналитику с определенной вероятностью судить о том, используется стеганография или нет. Проведенные компьютерные эксперименты не позволяют выделить тот или иной статистический метод, их надо проводить в совокупности.

СПИСОК ЛИТЕРАТУРЫ

1. Харин, Ю. С. Стеганографические методы защиты информации: обзор / Ю. С. Харин, М. С. Абрамович // Управление защитой информации. – 2009. – Т. 13, № 1. – С. 58–64.
2. Швадченко, И. В. Методы стеганоанализа для графических файлов / И. В. Швадченко // Киев, 2010.
3. Основы компьютерной стеганографии / А. В. Аграновский [и др.]. – М.: Радио и связь, 2003.
4. Варновский, И. П. Математика и безопасность информационных технологий в МГУ / И. П. Варновский, О. А. Голубев, О. А. Логачев // Современные направления стеганографии. – Издательство: МЦНО, 2005. – С. 32–64.
5. Fridrich, J. On steganographic embedding efficiency / J. Fridrich, P. Lisonek, D. Soukal // Lecture notes in Computer Science. – New York, 2006. – Vol. 4437. – P. 123–132.

STATISTICAL FORECASTING OF PANEL DATA BASED ON STATE SPACE MODELS

Lobach V. I., Merkulov R. I., Lobach S. V.

Mathematical Modeling and Data Analysis Department, Belarusian State University

Minsk, Republic of Belarus

E-mail: lobach@bsu.by, merkylovecom@mail.ru, lobashS@bsu.by

Panel (or longitudinal) data describes a set of objects which are observed during certain period of time, so they consist of repeated observations of the same objects in sequential time periods. The following examples of panel data can be mentioned: annual household studies, monthly performance indicators for economic institutions and many others. In this study we provide another approach to forecasting cross-sectional data based on state space models together with Kalman filtering procedure.

INTRODUCTION

In economic researches regression models are widely used within large number of applications [1]. Regression models for panel data allow usage of two indices to describe the data:

$$y_{i,t} = \alpha + X_{i,t}\beta + x_{i,t},$$

where i defines object index (household, company, country, etc), t denotes timestamp of an observation, α is an unknown intercept, β is a $(n \times 1)$ -vector of unknown parameters, $X_{i,t}$ is a known matrix denoting factors which influence observations. Uncontrollable factors $x_{i,t}$ are modeled with the following equation:

$$x_{i,t} = \mu_i + \epsilon_{i,t},$$

where μ_i is an unobservable individual effect of i -th object, $\epsilon_{i,t}$ is a random variable which defines random uncontrollable effect.

Statistical analysis of panel data can be carried out using state space models. In order to express panel data in a state space form it is necessary to introduce one more index i for state parameters vector x_t in classic state space model formulation. This results in $x_{i,t}$, where $t = 1, \dots, T_i$, $i = 1, \dots, K$, t denotes timestamp, i denotes object index. It means that the mathematical model for panel data is a random field $\{x_{i,t}\}$, $t = \overline{1, T_i}$, $i = \overline{1, K}$.

Based on linear state space models [2] we express panel data in a state space form:

$$x_{i,t} = Fx_{i,t-1} + \omega_{1,t},$$

$$y_{i,t} = Hx_{i,t} + \omega_{2,t},$$

where $x_{i,t}$ is an unobserved state of i -th object at moment t , $y_{i,t}$ is an observation for the object at the same moment. In common case $x_{i,t} \in R_1^n$, $y_{i,t} \in R_2^n$, $\{\omega_{1,t}\}$ and $\{\omega_{2,t}\}$ are sequences of i.i.d. random variables $\omega_{1,t}, \omega_{2,t} \sim N(0, Q)$, $x_{i,0} \sim N(\mu, P)$. The parameters of the model are F, H, μ, P . And the problem is to estimate future observations $x_{i,t+h}$, $y_{i,t+h}$ based on previous observations $y_{i,s}$, $s = 1, \dots, t$, $h > 0$.

I. KALMAN FILTER

Kalman Filter [3] allows to build optimal in mean-squared sense forecasts if they are introduced in linear state space form. Let us consider the following

$$x_{i,t}^t = E\{x_{i,t}|y_{i,0}^t\},$$

$$P_{i,t_1,t_2}^t = E\{(x_{i,t_1} - x_{i,t_1}^t)(x_{i,t_2} - x_{i,t_2}^t)|y_{i,0}^t\},$$

where $y_{i,0}^t = \{y_{i,j}, j = 1, \dots, t\}$.

Kalman Filter can be expressed using the following equations [4]

$$x_{i,t}^t = x_{i,t}^{t-1} + K_{i,t}(y_{i,t} - H_{i,t}x_{i,t}^{t-1}), \quad (1)$$

$$P_{i,t}^t = (1 - K_{i,t}H_{i,t})P_{i,t}^{t-1}, \quad (2)$$

$$K_{i,t} = P_{i,t}^{t-1}H_{i,t}^T(H_{i,t}P_{i,t}^{t-1}H_{i,t}^T + R)^{-1}, \quad (3)$$

where $i = \overline{1, K}$, $t = \overline{1, T_i}$, $x_{i,0} = \mu$, $P_{i,0}^0 = P$.

In order to compute forecasts for $x_{i,t}$ for h lags forward equations (1) – (3) are used with initial values $x_{i,t}^T, P_{i,t}^T$ instead of $x_{i,0}^0, P_{i,0}^0$.

In order to predict observed values $y_{i,t}$ for h future lags we provide the following procedure:

$$y_{i,t+h} = E\{y_{i,t+h}|y_{i,0}^T\},$$

$$B_{i,T+h}^T = E\{(y_{i,T+h} - y_{i,T+h}^T)|y_{i,0}^T\},$$

Using Kalman Filter (1) – (3) the following equations for forecasting statistics are provided:

$$x_{i,T+h}^T = Fx_{i,T+h-1}^T, \quad (4)$$

$$y_{i,T+h}^T = Hx_{i,T+h}^T, \quad (5)$$

$$P_{i,T+h}^T = FP_{i,T+h-1}^TF^T + Q, \quad (6)$$

$$B_{i,T+h}^T = H_iP_{i,T+h}^TH_i + R. \quad (7)$$

II. PANEL DATA IN LINEAR STATE SPACE FORM

Classic linear mixed regression model in a compact form can be expressed in the following way:

$$y = X\beta + Z\gamma + \epsilon, E\{\gamma, \epsilon\} = (0, 0),$$

$$\text{cov}(\gamma, \epsilon) = \begin{bmatrix} Q & 0 \\ 0 & R \end{bmatrix}$$

where y is observed variable with the following expectation and covariance $E\{y\} = XB$, $\text{cov}(y, y) = ZQZ^T + R$. Matrices X and Z describe determined and stochastic effects in observations respectively. For panel data modification of a linear mixed regression model observations for i -th object $y_i = (y_{i,1}, \dots, y_{i,T_i})^T$, $i = \overline{1, K}$ are aggregated for $t = \overline{1, T_i}$ which results in the following model:

$$y_i = X_i\beta + Z_i\gamma_i + \epsilon_i,$$

$$\gamma_i \sim N(0, G),$$

$$\epsilon_i = (\epsilon_{i,1}, \epsilon_{i,T_i})^T \sim N(0, \Sigma),$$

which leads to $y_i \sim N(X_i\beta, Z_iGZ_i^T + \Sigma_i)$.

One of the possible ways of expressing longitudinal modification of mixed regression model in state space form can be expressing observations $y_{i,t}$ as a single vector of higher dimensionality, then the state and observation equations can be formulated as following

$$y_{i,t} = x_{i,t}^T\beta_{i,t} + Z_{i,t}^T\gamma + \epsilon_{i,t}, \quad (8)$$

$$\beta_{i,t} = \beta_{i,t-1}, \quad (9)$$

where $\epsilon_{i,t} \sim N(0, \sigma^2)$.

Then we apply Kalman filtering procedure (1) – (3) to the panel data model (8) – (9) and finally construct forecasting statistics (4) – (7).

III. COMPUTATIONAL EXPERIMENTS

Let us consider the case described with the model (8) – (9). Let the observation vector be a constant vector with additive errors defined by $AR(1)$ process:

$$y_{i,t} = \beta_i + \epsilon_t,$$

$$\epsilon_t \sim N(0, \Sigma_t),$$

$$\Sigma_t(i, j) = \frac{\sigma^2\phi^{|i-j|}}{1-\phi^2}, |\phi| < 1.$$

One of possible state space models for this case can be the following:

$$y_{i,t} = \beta_i + \epsilon_t,$$

$$\epsilon_t = \phi\epsilon_t + \omega_t, \omega_t \sim N(0, \sigma^2)$$

with the initial conditions $\epsilon_t = N(0, \frac{\sigma^2}{1-\phi^2})$.

The task is to estimate model parameters which can be non-trivial due to nonlinear relationships between parameters. After parameters' estimates are built they can be used to construct forecasts $x_{i,t+h}$, $y_{i,t+h}$. In order to avoid this problem we construct another state space form

$$y_{i,t} = \mu + \beta_i + \epsilon_i,$$

$$x_{i,t} = \begin{pmatrix} \epsilon_t \\ \beta_i \end{pmatrix} = \begin{bmatrix} \phi & 1 \\ 0 & 1 \end{bmatrix} \begin{pmatrix} \epsilon_{t-1} \\ \beta_i \end{pmatrix} + \begin{pmatrix} \omega_t \\ 0 \end{pmatrix},$$

$$y_{i,t} = (1, 1)x_{i,t},$$

$$\omega_t \sim N(0, \Omega),$$

with the initial condition $(\epsilon_0, \beta_i)^T \sim N(0, G)$, where

$$G = \begin{bmatrix} \frac{\sigma^2}{1-\phi^2} & 0 \\ 0 & 0 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} \sigma^2 & 0 \\ 0 & 0 \end{bmatrix}.$$

CONCLUSION

Finally this results in linear state space model and we can apply Kalman filtering procedure (1) – (7). For computational experiments we generated two-dimensional time series according to model described above. The experiments were carried out with the following parameters: $\mu = 0$, $\sigma^2 = 1$, $\phi = 0.5$, $\beta_i = 1$. To construct forecasting statistics equations (4) – (7) were used. Forecasting horizon with $h = 10$ was used. We observed mean absolute percentage error below 2.1% which indicates possibility of modeling panel data using the described approach.

1. Baltagi, Badi H. Econometric analysis of panel data / B.H. Baltagi // John & Sons. New York, 2004. – 284 p.
2. Ivchenko G.I. Mathematical Statistics / G. I. Ivchenko // High, Sch., p. 248.
3. Harvey A. C. Forecasting, Structural Time Series Models and the Kalman Filter / A. C. Harvey // Cambridge: Cambridge University Press. p. 227.
4. Liptser R. Sp. Stochastic Processes Statistics / R. Sp. Liptser // Nauka, p. 696.

PARALLEL SKELETON ALGORITHM WITH ONE-SUBITERATION

Ma Jun, Ren Xunhuan, Tsviatkou, V. Yu., Konopelko, V. K.
 Department of Infocommunication Technologies, BSUIR
 Minsk, Belarus
 E-mail: majun469761364@gmail.com

The skeletonize of the binary image has crucial application in the field of target recognition. The thinning result of the Zhang's fast parallel thinning algorithm maintains the connection of the original image, has a good structural form and has no burs. However, Zhang's fast parallel thinning algorithm consists two subiterations in where the most of operations are similar, which cause the waste of the resources of the calculate. Besides, it can't ensure the single pixel width, which brings difficulties for post-processing.

INTRODUCTION

Skeletonization, also known as the image thinning, is a pre-processing which is widely applied in field of the pattern recognition. Image thinning refers to finding the skeleton or centerline of the original image as quickly as possible while maintaining the completeness of the topology of the original image, and then replacing the original image with a single-pixel skeleton. The processing of the image thinning can dramatically reduce the superfluous information from the image, which relieve the computation burden of the computer and shorten the time which spend on the process of the recognition [2][3][5].

In practice, there is a need for thinning images for obe or all of the following reasons: Preservation of topological and geometric properties, One-pixel thickness, Mediality, Insensitivity to boundary noise and efficiency.

Thinning algorithms can be classified in one of two broad categories: iterative thinning algorithms and non iterative thinning algorithms.

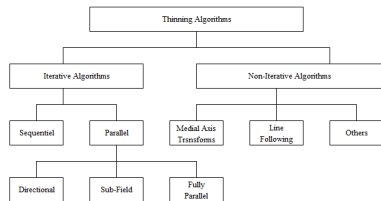


Рис. 1 – Classification of Thinning Algorithm

Iterative algorithms remove boundary pixels layer by layer until an appropriate skeleton is obtained. According to the way of examining pixels, they can be further divided into parallel and sequential. In a parallel algorithm, since the deletion of pixels depends on the results of the previous iteration, all the pixels can be processed simultaneously. By the contrary, the pixels in a sequential algorithm can not processe at the same time because the deletion of pixels not only depend on the previous iteration but also depend on the current iteration.

Non-iterative thinning algorithms unlike iterative ones are not based on examining individual

pixels. Some popular non-pixel based methods include medial axis transforms, distance transforms, and determination of center-line by line following.

Over the years the ZS algorithm has become one of the most cited and used parallel thinning algorithms, which yields very good results with respect to both connectivity and contour noise immunity. Besides, it can process all pixels simultaneously and is relatively fast also on sequential hardware. However, there are three main difficulties: excessive erosion in thinning diagonal lines, complete deletion of patterns.

In this paper a parallel skeleton algorithm with one-subiteration based on the ZS has proposed to not only overcome the drawbacks of ZS, but also accerate the speed of process.

I. ZHANG'S ALGORITHM

Zhang's algorithm consists of two sub-iterations. Iterative transformations are applied to original binary matrix point by point according to the values of a small set of neighboring points. It is assumed that a 3×3 window is used, and that each element connected with its eight neighboring elements which as figure shown below [1] [4].

P_8 (-2,-1)	P_2 (-1,-1)	P_6 (-1,1)
P_4 (0,-1)	P_0 (0)	P_0 (0,1)
P_1 (+1,-1)	P_3 (+1,0)	P_5 (+1,+1)

Рис. 2 – Designations of the nine pixels in a 3×3 window

The methods for extracting the skeleton of a picture consists of removing all contour pixes pf the picture except those belonging to the skeleton. In the first sub-iteration, the contour point P_1 is deleted from the digital pattern if it satisfies the following conditions:

1. $2 \leq B(P) \leq 6$
2. $A(P_1) = 1$
3. $P_2 \times P_4 \times P_6 = 0$
4. $P_4 \times P_6 \times P_8 = 0$

In the second sub-iteration, the contour point P_1 is deleted from the pattern, if it satisfies the following conditions:

1. $2 \leq B(P) \leq 6;$
2. $A(P_1) = 1;$

3. $P_2 \times P_4 \times P_8 = 0$;
4. $P_2 \times P_6 \times P_8 = 0$.

If any condition is not satisfied then P_1 will not be deleted from the foreground.

II. IMPROVEMENT ALGORITHM

Improvement algorithm includes 4 main parts: Search Module, Connectivity Check Module, Single Pixel Correction Module and Contour Point Delete Module. The structure of these stages addressed in Flowchart as shown in Fig. 3.



Рис. 3 – Flowchart of the proposed method

The binary image is acquisition into the proposed method as black pixels which considered as a foreground as well as consider an object pixel for deletion. The pixels having value 0 are considered as background pixels.

Before introducing the Search module, the scan window which we used in the Search Module need to present, which is a little different from Zhang's algorithm. As shown in Figure 4.

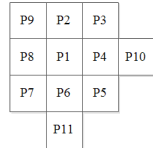


Рис. 4 – Designation of the scan window

In each iteration, if the current pixel belongs to the foreground pixel and its 8-neighbour pattern satisfied with the deletion condition, it can be considered as a candidate pixel which may be deleted and then they should be compared with the Restoring Templates. Those candidate pixels which different from the restoring templates will be deleted from the foreground.

1. $2 \leq B(P) \leq 6$;
2. $A(P_1) = 1$;
3. $\overline{P_6} + P_2 + P_{11} = 1$;
4. $\overline{P_4} + P_{10} + P_8 = 1$.

When there is not anymore deletable pixel during the iteration process, the iteration is coming to the end and an additional procedure will be deployed to ensure the one-pixel width of the result by deleting the redundancy pixel. If one foreground pixel and its 8-neighbour satisfied with one of the following criteria, a pixel will be transformed into a background pixel.

1. $P_4 \times P_6 = 1$ and $P_9 = 0$;
2. $P_6 \times P_8 = 1$ and $P_3 = 0$;
3. $P_2 \times P_4 = 1$ and $P_7 = 0$;
4. $P_2 \times P_8 = 1$ and $P_5 = 0$.

III. EXPERIMENTS AND RESULTS

To assess the performance, the improvement algorithm and the Zhang's algorithm were written in Matlab R2018b. This data set has many class shapes.

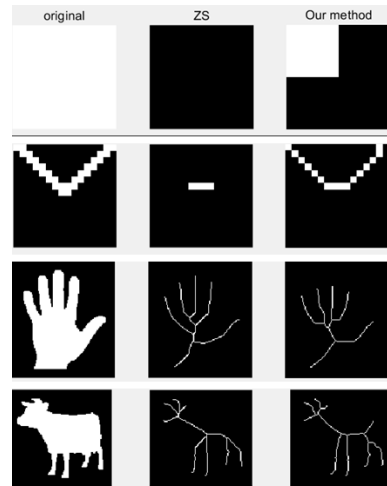


Рис. 5 – The original binary images and the thinning results by different algorithm

From the Fig.5, we know that the new method has a better performance in the single pixel.

Таблица 1 – Comparison of the compute speed between algorithms

BI_o	A_{zhang}	A_{new}	$C_{time(zh)}$	$C_{time(n)}$
$S_{2 \times 2}$	0	1	0.02	0.008
S_l	4	16	0.008	0.004
Hand	333	260	0.1111	0.086
Cow	293	285	0.095	0.064

Original Binary image - I_o ; $S_{2 \times 2}$ square - $S_{2 \times 2}$; Slope line - S_l ; Remain points of Zhang's Algorithm - A_{zhang} ; Remain points of New Algorithm - A_{new} ; CPU Time Consumed(s) of Zhang's Algorithm - $C_{time(zh)}$; CPU Time Consumed(s) New Algorithm - $C_{time(n)}$.

According to the results of the experiment, improved algorithm not only overcome the three main drawbacks of the ZS algorithm, but also present a higher efficiency in procession.

IV. CONCLUSION

In this paper, we presented an improved algorithm based on the zhang's algorithm, which has better performance in speed and single pixel. The experiments have proved the effectivity of the new algorithm.

1. Mangrum, J. M. The evaluation and management of bradycardia / J. M. Mangrum, J. P. DiMarco // N. Engl. J. Med. – 2000. – Vol. 342, № 10. – P. 703–709.
2. Zhang, Y. A fast thinning algorithm for thinning digital patterns /Y. ZHANG, C. Y. SUEN // A Communications of ACM.
3. Bushenko, D. A. Technology of thinning algorithms construction for skeletonization of extended objects /D. A. Bushenko, R. H. Sadikhov // Reports of BSUIR. – 2009. – Vol. 45, № 7. – P. 81–86.
4. Waleed, A. A. Skeletonization Algorithm for Binary Images /A. A. Waleed, B. Bataineh, T. Abu, O. P. Khairuddin //Technology, – 2013. – P. 704–709.
5. Ye, F. L. An improved Image Skeleton Extraction Algorithm /F. L. Ye //Journal of Xichuang Univeristy. – 2018. – Vol. 32, № 3. – P. 91–93.

МЕТОД АГРЕГАЦИИ ПОТОКОВ ДАННЫХ В ЗАДАЧАХ ЛОКАЛИЗАЦИИ ПРОБЛЕМ

Моженкова Е. В., Парамонов А. И.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: elena.mozhenkova@gmail.com, anton_paramonov@tut.by

В работе рассматривается проблема сбора данных на стороне клиента с целью дальнейшей локализации проблемы. Предлагается подход к решению данной проблемы, в основе которого используется алгоритм обмена данными между клиентским приложением и окружением разработчика. Алгоритм предполагает описание информационной модели программного продукта в разрезе взаимодействия потоков данных с учетом возможной разнородности структур банков данных.

ВВЕДЕНИЕ

Корпоративные информационные системы (КИС) имеют продолжительный жизненный цикл, т.к. бизнес-процессы в таких системах необходимо постоянно дорабатывать согласно изменениям законодательства и исправления найденных дефектов с целью удовлетворения требованиям клиентов. Из-за ограничений в стоимостных и временных интервалах разработки эта проблема является актуальной задачей и требует новых подходов. Для решения данной проблемы разработчики КИС проектируют функциональность экспорта-импорта данных для своей структуры базы данных (БД) в отдельности. Данный подход накладывает дополнительные затраты на сопровождение функциональности обработки банков данных (БнД). Поэтому основная цель исследования – это определение области расширения автоматизации в задачах сбора и анализа данных клиента по выявленному дефекту, с целью локализации бизнес-процесса системы на стороне разработчика.

Возникновение дефектов программного обеспечения (ПО) сопряжено в первую очередь с данными, которые обрабатываются тем или иным модулем КИС. Поэтому источником информации для локализации дефекта в КИС является база данных предприятия. Структура БД может хранить в себе не только пользовательские данные (информация, на основании которой построены бизнес-процессы предприятия), но и конфигурационные настройки системы. Это усложняет логику хранения информации в банках данных, а следовательно, и затрудняет построение алгоритма сбора информации при обработке БнД для локализации дефекта.

1. АЛГОРИТМ ОБМЕНА ДАННЫМИ МЕЖДУ КИС КЛИЕНТА И РАЗРАБОТЧИКА

Для анализа данных КИС предлагается разработать автоматизированную информационную систему обеспечения сбора данных и передачи их на уровень разработчика - АИС «Агрегатор данных» (далее АИС). Взаимодействие АИС

с банками данных клиента и разработчика можно описать схемой, представленной на рис. 1. На схеме видно, что АИС является прослойкой между рабочей и тестовой базами, обеспечивая организацию передачи данных бизнес-процесса клиента. Обмен данными будет осуществляться через промежуточные таблицы, в которые агрегируется информация, необходимая для локализации потенциального дефекта бизнес-процесса КИС.



Рис. 1 – Схема взаимодействия АИС с КИС клиента и разработчика

Предполагается следующий алгоритм процесса обмена данными. Разработчик КИС, определяет набор данных, необходимых для локализации дефекта. Администратор настраивает подключение к БД клиента. АИС формирует схему БД исследуемой КИС, и предоставляет агрегированную информацию о структуре данных. Согласно определенному разработчиком набору данных, администратор осуществляет выборку таблиц, затем помечает таблицы для выполнения фильтрации записей, и устанавливает значения фильтров. После настройки механизма обработки, АИС формирует запрос на выборку данных, выгружает их в промежуточные таблицы и формирует конфигурационный файл [1]. Этот документ будет открыт в АИС на стороне разработчика. Данные из конфигурационного файла загружены в тестовую БД. После выполнения описанного процесса разработчик сможет воссоздать ошибку при работе клиента на тесто-

вой БД. Для правильного выполнения алгоритма обмена данными необходимо формально описать обработку таблиц базы данных исследуемой КИС. Обработка данных описанным способом является нетривиальной задачей.

II. ИНФОРМАЦИОННАЯ МОДЕЛЬ КИС

Для построения алгоритма работы АИС опишем модель представления информации в КИС и рассмотрим структуры потоков данных, с которыми работает АИС. Вся информация, которая вводится пользователем, имеет временные показатели. Соответственно информация рассматривается в разрезе заданных временных интервалов. Основным пользователем – расчетчик, который вводит первичную информацию или бухгалтер, который агрегирует первичные данные. Чаще всего для пользователя КИС устанавливается расчетный период, в рамках которого выполняет манипуляции с данными в системе [2, 3]. Модель представления информации в КИС можно отобразить схемой взаимодействия потоков данных, которая представлена на рис. 2. В представленной модели вводится временная характеристика $T_{(j)}$ – расчетный период модуля, т.е. данные доступны в k интервалов. Схема взаимодействия потоков данных в КИС происходит в различных временных интервалах, для каждого выделяется свой логический слой доступа к данным – слепок метаданных. Поэтому основная задача обработки БД при локализации дефекта – выбрать слепки данных за один или несколько временных слоев.

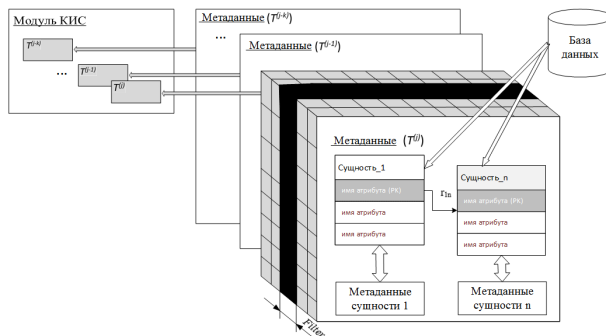


Рис. 2 – Схема взаимодействия потоков данных в КИС

Исходными данными для локализации дефекта выступают: структура метаданных, связи между таблицами и информация, хранящаяся в БД. Для ограничения набора выборки (среза метаданных) вводится понятие менеджера таблиц – базовая таблица БД в контексте методанных, которую пользователь помечает как объект для фильтрации выборки. Для локализации различного рода дефектов предлагается использовать шаблоны сущностей КИС: каждому классу де-

фектов – соответствует свой набор сущностей. Каждая сущность КИС может быть представлена несколькими таблицами БД. В зависимости от типа обрабатываемых сущностей, к модели могут быть применимы или нет, временные характеристики и фильтр значений полей базовой таблицы (Filter). Программное средство АИС должно учитывать эти входные данные в виде параметров.

В общем процесс локализации дефекта можно описать следующим алгоритмом. Структура БД состоит из набора таблиц, которые описывают сущности КИС за весь период жизненного цикла. По набору таблиц определяется менеджер таблиц – таблица, которая является первичной в обработке метаданных. На основе связей этой таблицы формируется выборка слепка данных. Фильтр Filter устанавливается в разрезе записей менеджера таблиц. Далее определяется набор связанных таблиц и отбираются необходимые данные. Процесс продолжается до тех пор, пока не будут обработаны все таблицы сущностей слепка данных. В зависимости от типа обрабатываемых сущностей к модели могут быть применимы или нет временные характеристики и фильтр значений полей базовой таблицы. Алгоритм основан на частичной выборке данных из БД, предполагает формирование диапазонов значений фильтруемых данных, соответствующих указанным в исходной формулировке задачи, с последующей проверкой наличия записей в связанных таблицах.

III. ЗАКЛЮЧЕНИЕ

Применение предложенного подхода в алгоритме обмена данными позволяет ограничить объем рассматриваемых данных для выборки и переноса на сторону разработчика, сократить временные затраты для локализации дефекта.

СПИСОК ЛИТЕРАТУРЫ

1. Моженкова, Е. В. Схема генерации и восстановления данных при локализации дефекта / Е. В. Моженкова, А. И. Парамонов // Компьютерные системы и сети: 55-я юбилейная научная конференция аспирантов, магистрантов и студентов, Минск, 22-26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники. – Минск, 2019. – С. 132 – 133.
2. Abrar Ullah, Enterprise Resource Planning Systems and User Performance / Abrar Ullah, Rohaizat Bin Baharun Khalil MD Nor, Muhammad Siddique and Abdul Sami // International Journal of Applied Decision Sciences XI(03):377-390, August 2018.
3. Калягина Н. В., Сравнительный анализ функций СУБД, с которыми работают современные корпоративные информационные системы / Н. В. Калягина // Вестник Волжского университета им. В.Н. Татищева – Тольятти, 2010. – С. 151 – 158.

DUAL CONTROL OF MULTIDIMENSIONAL-MATRIX STOCHASTIC OBJECTS

Mukha V. S., Kako N. F.

Department of Information Technologies of Automated Systems, Belarusian State University of Informatics and Radioelectronics

Minsk, Republic of Belarus

E-mail: mukha@bsuir.by, kako.nancy@gmail.com

The main results in dual control theory are analyzed. The problem of the dual control of the multidimensional-matrix stochastic objects is formulated and the functional equations for its solution are given.

INTRODUCTION

Dual control theory deals with the control of objects whose characteristics are initially unknown. The controller's objectives in controlling such an object are twofold. One objective is control action to control of the object as well as possible based on current knowledge on object and the other is experiment with the object so as to learn about its behavior and control it better in the future. The dual control theory was developed by Russian scientist A.A. Feldbaum in 1960 [1] and published later in the monograph [2] (see also [3]).

I. DUAL CONTROL OF THE DETERMINISTIC OBJECTS

The system having the block diagram in the figure 1 was considered in the works [1, 2, 3].

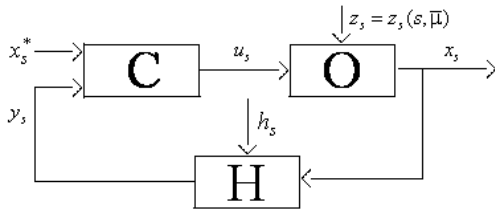


Рис. 1 – To dual control of the deterministic objects

In the figure 1 h_s is the sequence of independent random numbers in the measurement channel H of the output variable of the controlled object O , C is the controller. The controlled object O is deterministic and is described by function $x_s = F(z_s, u_s) = F(s, u_s, \bar{\mu})$, where $\bar{\mu}$ is a random vector with known priory probability density $f(\bar{\mu})$. It is assumed that the store of information about the vector $\bar{\mu}$ can be accumulated in the control process. The task consists of determining such a sequence of control actions u_0, u_1, \dots, u_n that provides minimum to the total risk R : $R = \sum_{s=0}^n E\{W_s(s, x_s, x_s^*)\} = \sum_{s=0}^n R_s$, where W_s is loss function, x_s^* is the some known sequence, R_s is the mean specific risk, E is a symbol of the mathematical expectation. In the works [1,2,3] the functional equations is derived which allow in principle to determine the optimal sequence of the control actions. However, it is practically impossible to receive the solution of the equations in the analytical form. In the case of the numerical solving

of the equations the difficulties connected with the necessity to remember and minimize the functions of a large number of variables are appeared, moreover, the number of variables increases with the increasing interval of the control. The equations are such that a reasonable choose of the method of the numerical integration can be performed in the case of determining the control action only at the penultimate time instant $n - 1$ when the integrand can be received in analytical form. The reasonable choose of the method of the numerical integration in the following calculations seems extremely difficult, because the integrand is known in discrete set of points. In the paper [4] other approach to the solution to the problem is proposed. It consists of the separation of the problem into the problem of the estimation of the parameters of the controlled object and the problem of the optimal control on the base of received estimations. This approach is more effective compared with the approach of the works [1,2,3].

II. DUAL CONTROL OF THE MULTIVARIATE STOCHASTIC OBJECTS

The problem of dual control of the deterministic objects considered in [1,2,3] was generalized in paper [5] to the multivariate stochastic objects with the unknown parameters in the form of the vector random sequence. The approach of the paper [4] consisting of the separation of the problem into the problem of the estimation of the parameters and the problem of the optimal control was used to solution to this problem. The problem is formulated as following. The feedback system showed in the figure 2 is considered, where O is the controlled object, C is the controller, $\bar{y}_s = (y_{1,s}, y_{2,s}, \dots, y_{p,s})$ is the output vector of the object with p coordinates at the time instant s , $\bar{u}_s = (u_{1,s}, u_{2,s}, \dots, u_{k,s})$ is the vector of the control action with k coordinates at the time instant s , $\bar{x} = (x_1, x_2, \dots, x_m)$ is the random vector with m coordinates, $\bar{x}_s = (x_{1,s}, x_{2,s}, \dots, x_{m,s})$ is the random vector sequence with m coordinates at the time instant s , $\bar{g}_s = (g_{1,s}, g_{2,s}, \dots, g_{l,s})$ is the vector of the 'driving action' with l coordinates at the time instant s .

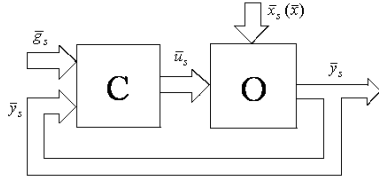


Рис. 2 – To dual control of the stochastic objects

Controlled object O at the time instant s is described by the conditional probability density $f(\bar{y}_s/\bar{x}_s, \bar{u}_s)$, $s = 0, 1, 2, \dots, n$. Such an object is stochastic and is called a regression object. The vector of the ‘driving action’ \bar{g}_s is assumed, for simplicity, to be known deterministic sequence. The probability density $f(\bar{x})$ of the random vector \bar{x} or the $(n+1)$ -dimensional probability density $f(\bar{x}_n) = f(\bar{x}_0, \bar{x}_1, \dots, \bar{x}_n)$ of the random sequence \bar{x}_s are assumed to be known. The control vector \bar{u}_s is restricted by the condition $\bar{u}_s \in U$, where U is a certain admissible region, for example, $U = \{\bar{u}_s : \bar{u}_s \geq 0, \psi_j(\bar{u}_s) \leq 0, s = 0, 1, \dots, n, j = 1, 2, \dots, l\}$, and $\psi_j(\bar{u}_s)$ are some scalar functions of the vector \bar{u}_s . It is assumed, that the controller C utilizes the all past information in the form of the measurements $\bar{u}_{s-1} = (\bar{u}_0, \bar{u}_1, \dots, \bar{u}_{s-1})$, $\bar{y}_{s-1} = (\bar{y}_0, \bar{y}_1, \dots, \bar{y}_{s-1})$ to form the control action \bar{u}_s at the time instant s . It is assumed, that the controller has a random behavior and is described at the time instant s by the conditional probability density $\Gamma_s = f(\bar{u}_s/\bar{u}_{s-1}, \bar{y}_{s-1})$ called the strategy of the controller. The quality of the functioning of the system at the time instant s is evaluated by the loss function $W_s(\bar{y}_s, \bar{g}_s)$ depending on the output vector \bar{y}_s and may on the vector of the ‘driving action’ \bar{g}_s . Mathematical expectation of the loss function $R_s = E\{W_s(\bar{y}_s, \bar{g}_s)\}$ is called mean specific risk. The sum of the mean specific risks on the $n+1$ time instants

$$R = E\left\{\sum_{s=0}^n W_s\right\} = \sum_{s=0}^n E\{W_s\} = \sum_{s=0}^n R_s \quad (1)$$

is called the mean total risk. The system will be optimal when the mean total risk R will be minimal. The problem consists of determining the sequence of the strategies of the controller $\Gamma_s = f(\bar{u}_s/\bar{u}_{s-1}, \bar{y}_{s-1})$, $s = 0, 1, 2, \dots, n$, minimizing the mean total risk R (1).

As it is known [1,2,3], the strategies of the controller $\Gamma_s = f(\bar{u}_s/\bar{u}_{s-1}, \bar{y}_{s-1})$, $s = 0, 1, \dots, n$, minimizing the mean total risk R (1) are not randomized, i.e. $\Gamma_s = \delta(\bar{u}_s - \bar{u}_s^*)$, δ is the unit impulse function, \bar{u}_s^* is optimal control action. Provided the \bar{x} is random vector, the sequence of the control actions $\bar{u}_n^*, \bar{u}_{n-1}^*, \dots, \bar{u}_0^*$ is defined from the following functional equations:

$$\begin{aligned} f_n^*(\bar{u}_{n-1}, \bar{u}_n^*, \bar{y}_{n-1}) &= \min_{\bar{u}_n \in U} \phi_n(\bar{u}_n, \bar{y}_{n-1}), \quad (2) \\ f_{n-r}^*(\bar{u}_{n-r-1}, \bar{u}_{n-r}^*, \bar{y}_{n-r-1}) &= \\ &= \min_{\bar{u}_{n-r} \in U} (\phi_{n-r}(\bar{u}_{n-r}, \bar{y}_{n-r-1}) + \\ &+ \int_{\Omega(\bar{y}_{n-r})} f_{n-r+1}^*(\bar{u}_{n-r}, \bar{u}_{n-r+1}^*, \bar{y}_{n-r}). \end{aligned}$$

$$\cdot f(\bar{y}_{n-r}/\bar{u}_{n-r}, \bar{y}_{n-r-1})d\Omega), r = 1, 2, \dots, n, \quad (3)$$

where

$$\phi_s(\bar{u}_s, \bar{y}_{s-1}) = \int_{\Omega(\bar{y}_s)} W_s(\bar{y}_s, \bar{g}_s) f(\bar{y}_s/\bar{u}_s, \bar{y}_{s-1})d\Omega, \quad (4)$$

$$f(\bar{y}_s/\bar{u}_s, \bar{y}_{s-1}) = \int_{\Omega(\bar{x})} f(\bar{y}_s/\bar{x}, \bar{u}_s) f_s(\bar{x})d\Omega, \quad (5)$$

$$f_s(\bar{x}) = \frac{f_0(\bar{x}) \prod_{i=0}^{s-1} f(\bar{y}_i/\bar{x}, \bar{u}_i)}{\int_{\Omega(\bar{x})} f_0(\bar{x}) \prod_{i=0}^{s-1} f(\bar{y}_i/\bar{x}, \bar{u}_i)d\Omega}. \quad (6)$$

III. DUAL CONTROL OF THE MULTIDIMENSIONAL-MATRIX STOCHASTIC OBJECTS

The representation of the scalar-vector or the vector-vector regression function of the many input variables by the polynomial of the degree more than one has a serious drawback such as absent of the strict theoretic formalization and algorithmic generality. In this paper the multidimensional-matrix approach to the description of the controlled object is proposed [6]. It does not have this drawback. It is the analogue to the scalar approach, generalizes the vector-matrix approach and owns the strict theoretic formalization and algorithmic generality. P -dimensional-matrix regression function $\bar{y}_s = \phi(\bar{x}, \bar{u}_s)$ of the q -dimensional matrix argument \bar{u}_s describing the controlled object O in the form of the polynomial of the degree r has the following expression [7]:

$$\bar{y}_s = \sum_{i=0}^r 0, i^q (\bar{x}_i \bar{u}_s^i).$$

So, we propose to consider the system in the figure 2, but we will consider the variables in the figure 2 as multidimensional-matrix. The theory in this case will be practically identical to the formulas (2) – (6).

REFERENCES

1. Feldbaum, A.A. Dual control theory. // Automation and remote control. 1960, V. 21: No 9, pp. 1240–1249; No 11, pp. 1453–1464; 1961, V. 22: No 1, pp. 3–16; No 2, pp. 129–142. In the Russian.
2. Feldbaum, A.A. Fundamentals of the theory of the optimal automatic systems. Moscow, Nauka, 1963. 553 p. In the Russian.
3. Feldbaum, A.A. Optimal Control Systems. Academic Press, New York and London, 1965. 452 p.
4. V.S. Mukha. On the dual control of the inertialess objects // Proceedings of the LETI. Issue 130, 1973. Pp. 31–37. In the Russian.
5. V.S. Mukha, E.V. Sergeev. Dual control of the regression objects // Proceedings of the LETI. Issue 202, 1976. Pp. 58–64. In the Russian.
6. V.S. Mukha. Analysis of multidimensional data. Minsk, Technoprint, 2004. 368 p. In the Russian.
7. V.S. Mukha. Analysis of multidimensional data. Laboratory manual. Minsk, BSUIR, 2018. 64 p. In the Russian.

ОБОСНОВАНИЕ КЛАССИФИКАЦИОННЫХ ПРИЗНАКОВ ДЛЯ РАСПОЗНАВАНИЯ МАЛОВЫСОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ В ЭЛЕКТРОСТАТИЧЕСКОЙ СИСТЕМЕ ПАССИВНОЙ ЛОКАЦИИ

Нефёдов Д. С., Курилович В. В., Панасюк М. Н.
Научно-исследовательская часть, факультет противовоздушной обороны,
Учреждение образования «Военная академия Республики Беларусь»
Минск, Республика Беларусь
E-mail: noof85@mail.ru

В докладе обоснована целесообразность использования спектрального портрета для распознавания класса маловысотного летательного аппарата в пассивной системе электростатической локации. В качестве информативных признаков предложено использовать ширину планерной составляющей и центральную частоту модуляционной составляющей амплитудно-частотного спектра возмущений напряженности электрического поля.

ВВЕДЕНИЕ

Третьим этапом получения локационной информации принято считать распознавание. Задача распознавания представляет собой установление принадлежности объекта наблюдения к определенному классу, например, вертолет, стратегический бомбардировщик, ложная цель. Решение задачи распознавания, как правило, осуществляется на основании сравнения локационного портрета объекта наблюдения с эталонными портретами объектов распознаваемых классов [1].

Эталонные портреты, представляют собой априорно известное распределение параметров сигналов, создаваемых объектами распознаваемых классов при их наблюдении в системе локации. Априорные портреты могут быть получены методом математического моделирования или экспериментально.

В электростатической локации летательных аппаратов (ЛА) задача распознавания может быть решена путем сравнения, зарегистрированных возмущений напряженности электрического поля с возмущениями, создаваемыми одним из распознаваемых классов ЛА. Исходя из практических приложений системы пассивной электростатической локации, которые включают обнаружение маловысотных ЛА и автономные системы поражения вертолетов, в качестве распознаваемых классов целесообразно выделить самолет тактической авиации (ТА), крылатую ракету (КР) и вертолет.

I. ПОСТАНОВКА ЗАДАЧИ РАСПОЗНАВАНИЯ В ЭЛЕКТРОСТАТИЧЕСКОЙ ЛОКАЦИИ

По аналогии с радиолокацией, задачу распознавания в электростатической локации можно разделить на следующие этапы [2]:

1. Выбор типа портрета (дальномерный, картинный, спектральный).

2. Анализ информационных признаков портрета, характеризующих его принадлежность к распознаваемым классам.
3. Формирование базы данных эталонных портретов распознаваемых классов.
4. Сравнение полученного в результате наблюдения портрета ЛА с эталонными и присвоение ему степени соответствия тому или иному классу.

Эффективность решения многоальтернативной статистической задачи распознавания в значительной мере зависит от обоснованности решений, принятых на этапах 1—3.

В электростатической локации наблюдению доступна реализация возмущений напряженности электрического поля. Распознавание основано на предположении зависимости временной структуры возмущений электрического поля от типа объекта их порождающих. Подтверждением сказанному являются результаты экспериментальных исследований [3, 4].

Также в результате исследований получено, что выбор типа портрета зависит от расстояния между наблюдаемым ЛА и электростатическим датчиком.

На расстояниях более 20...30 метров ЛА представляется как точечный заряд. В этом случае использование дальномерного или картинного портрета невозможно ввиду отсутствия разрешения элементов ЛА. Поэтому для решения задачи распознавания целесообразно рассмотреть спектральные портреты ЛА.

II. АНАЛИЗ СПЕКТРАЛЬНЫХ ПОРТРЕТОВ МАЛОВЫСОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

Спектральный портрет (амплитудно-частотный спектр) маловысотных ЛА в электростатической локации может быть получен путем преобразования Фурье сигнала на выходе чувствительного элемента датчика. При использовании электростатического датчика с диффе-

ренцирующим преобразователем напряженности электрического поля в напряжение, временная структура сигнала на его выходе описывается выражением [5,6]:

$$U_s(t) = -3\lambda_a \lambda_\tau^3 (\lambda_\tau^2 + t^2)^{-\frac{5}{2}} \quad (1)$$

$$\lambda_\tau = \frac{(P^2 + H^2)^{\frac{1}{2}}}{V} \quad (2)$$

где λ_a - амплитуда полезного сигнала ЛА, В/м; λ_τ - временной параметр полезного сигнала, зависящий от траекторных параметров ЛА, с; H - высота полета ЛА, м; P - параметр полета ЛА относительно точки наблюдения, м. Амплитудно-частотный спектр сигнала (1) описывается выражением:

$$S(f) = \int_{-\infty}^{\infty} U_s(t) e^{-2\pi i f t} dt \quad (3)$$

На рисунке 1 представлены усредненные нормированные АЧС самолета ТА, КР и вертолета.

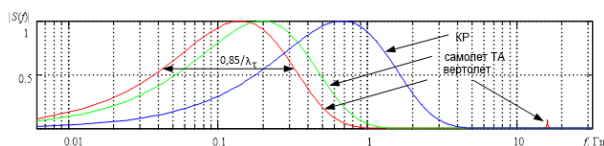


Рис. 1 – Усредненные нормированные АЧС различных классов ЛА

В результате анализа АЧС можно сделать следующие выводы:

1. АЧС всех распознаваемых классов маловысотных ЛА содержит основную (планерную) составляющую на частотах 0,1...2 Гц. Планерная составляющая обусловлена перемещением заряда корпуса ЛА относительно точки размещения электростатического датчика.

2. Ширина планерной составляющей АЧС зависит от временного параметра и равна $0,85/\lambda_\tau$.

Как видно из выражения (2), временной параметр определяется диапазоном высот и скоростей полета маловысотных ЛА. На основании анализа летно-технических характеристик распознаваемых классов ЛА получено, что диапазоны значений временного параметра отличаются. Области возможных значений параметра λ_τ самолетов ТА, КР и вертолета изображены на рисунке 2.

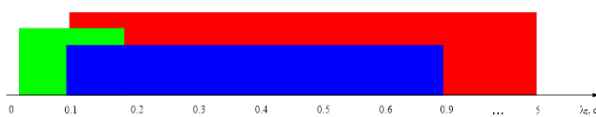


Рис. 2 – Усредненные нормированные АЧС различных классов ЛА

Таким образом, в качестве первого информационного признака целесообразно выбрать ширину планерной составляющей АЧС.

3. В отличие от КР и самолетов ТА, АЧС вертолета содержит вторую (модуляционную) составляющую которая обусловлена вращением несущего винта. Центральная частота модуляционной составляющей определяется произведением скорости оборота несущего винта на количество лопастей и для современных вертолетов принимает значение 10...30 Гц [3].

Наличие модуляционной составляющей и ее центральная частота АЧС вертолета являются вторым информационным признаком для решения задачи распознавания. Примечательно, что по измеренному значению центральной частоты модуляционной возможно установление типа вертолета (Ми-8, Ми-24, АН-64 и др.).

4. Отношение сигнал/шум для планерной и модуляционной составляющих на дальностях до ЛА 150...200 м составляет не менее 15 дБ, что позволяет решать задачи оценки информационных признаков и распознавания.

ЗАКЛЮЧЕНИЕ

Распознавание ЛА в электростатической локации целесообразно производить по их спектральным портретам. В качестве информационных признаков спектральных портретов маловысотных ЛА, необходимо использовать ширину планерной составляющей, а также центральную частоту модуляционной составляющей АЧС. Разработка алгоритмов распознавания является направлением дальнейших исследований.

СПИСОК ЛИТЕРАТУРЫ

1. Охрименко, А.Е. Основы радиолокации и радиоэлектронная борьба / А.Е.Охименко. – М.: Воениздат, 1983. – 456 с.
2. Радиолокационное распознавание: теоретические основы радиолокации / Я.Д.Ширман [и др.]. – Харьков, 1994. – 122 с
3. Исследование характеристик сигналов электростатической локации, создаваемого вертолетом: материалы XV международной конференции молодых ученых НАН РБ 29.10.2018 – 01.11.2018/НАН РБ.
4. Мелец, А.Ф. Экспериментальные исследования по оценке скорости полета неуправляемых авиационных ракет с использованием модуля пассивной электростатической локации / А.Ф. Мелец, Д.С. Нефедов // Вестн. Воен. Акад. Респ. Беларусь. – 2015. – №4. – С. 96-103.
5. Успенский, А. К. Вопросы теории и техники использования электростатических полей для локации объектов: дис. ... д-ра техн. наук / А. К. Успенский. – Минск: МВИЗРУ ПВО, 1979. – 414 с.
6. Мелец, А. Ф. Обнаружение и дистанционное измерение электрического заряда маловысотных аэродинамических объектов: дис. ... канд. техн. наук / А. Ф. Мелец. – Минск: МВИЗРУ ПВО, 1991. – 193 с.

ИСПОЛЬЗОВАНИЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ ДЛЯ КЛАССИФИКАЦИИ И АНАЛИЗА ТОНАЛЬНОСТИ ТЕКСТОВ

Нестеренков С. Н., Федоров П. А., Денисов В. А.

Кафедра программного обеспечения информационных технологий, кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: nsn@bsuir.by, pavl.fedorov@gmail.com, q1munys@gmail.com

В последние несколько лет применение сверточных нейронных сетей становится всё более широким. В данной работе приведены основные задачи и методы классификации текста с использованием сверточных нейронных сетей. Рассмотрена структура сверточных нейронных сетей и их принцип работы.

ВВЕДЕНИЕ

Отнесение документов к определённой категории на основании его содержимого называется классификацией документа. Данная задача является одной из задач информационного поиска. Процесс классификации может осуществляться как полностью в ручную так и с помощью применения методов машинного обучения в частности сверточных нейронных сетей. Также следует отличать классификацию от кластеризации, где в последнем случае тексты тоже группируются по категориям, которые заранее определены. Задача классификации текстов становится все более востребованной в связи с постоянным ростом информации в интернете и необходимостью в ней ориентироваться. Например, задача классификации текстов применима к решению следующих задач:

1. Персонафикация рекламы;
2. Разделение сайтов по тематическим каталогам;
3. Борьба со массовой рассылкой корреспонденции рекламного характера;
4. Распознавание тональности текстов.

В тоже время задача интеллектуального анализа текстовой информации, которая способна определять автора и пол автора текста, возраст, уровень образования, эмоциональное состояние автора в момент написания текста также является актуальной задачей. Под тональностью будем понимать эмоционально окрашенную лексику и эмоциональную оценку, выраженную автором относительно чего-либо. Анализ тональности имеет важное практическое применение:

1. Оценка качества товаров и услуг на основании отзывов пользователей интернет-ресурсов;
2. Противодействие экстремизму и терроризму;
3. Анализ ситуации на фондовых рынках и прогнозирование волатильности финансовых активов;
4. Составление текстов с заранее заданными эмоциональными характеристиками.

I. ПОСТАНОВКА ЗАДАЧИ

Задача классификации текстовой информации определяется следующим образом. Пусть существует конечное множество категорий $C = \{c_1, c_2, \dots, c_m\}$, конечное множество документов $D = \{d_1, d_2, \dots, d_n\}$ и неизвестная целевая функция Φ , определяющая соответствие для каждой пары <документ, категория> $\Phi : D \times C \rightarrow \{0,1\}$. Задача состоит в нахождении функции Φ' , которая является максимально близкой к целевой функции Φ . Эта функция называется классификатором.

Существует различные виды классификации текстов:

1. Субъективность/объективность
2. Классификация по многополосной шкале;
3. Классификация по бинарной шкале;
4. Классификация по многополосной шкале.

Основной задачей анализа тональности является выявление мнений в тексте и определить их свойств. Мнения делятся на два типа:

1. сравнение;
2. непосредственное мнение.

Непосредственное мнение содержит высказывания автора об одном объекте. Более формальное определение мнения выглядит как кортеж из пяти элементов (op, e, f, t, h), где:

1. orientation или polarity — тональная оценка;
2. (entity, feature) — объект тональности e или его свойства f;
3. момент времени time, когда было оставлено мнение;
4. holder — субъект тональности (автор).

Примеры тональных оценок:

1. нейтральная;
2. негативная;
3. позитивная.

Нейтральная оценка подразумевает отсутствие какого-либо эмоционального окраса. Также могут существовать и другие оценки.

Чаще всего в современных системах автоматизированного определения тональности текста применяется одномерное эмотивное пространство: позитив или негатив (хорошо или плохо).

II. НЕЙРОННЫЕ СЕТИ

Нейронная сеть представляет собой систему соединённых и взаимодействующих между собой простых процессоров. На рисунке 1 изображена схема нейронной сети, где голубым — скрытые нейроны, зелёным цветом обозначены входные нейроны, жёлтым — выходной нейрон.

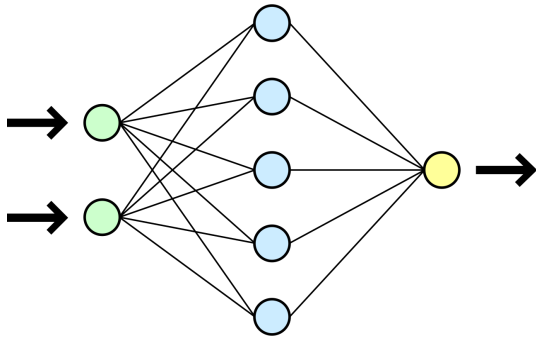


Рис. 1 – Схема простой нейронной сети.

По характеру обучения нейронные сети бывают:

1. использующие обучение без учителя;
2. использующие обучение с учителем.

Многослойная нейронная сеть состоит из слоев:

1. Входной - определение входных признаков;
2. Скрытые - расчет промежуточных результатов;
3. Выходной - вывод окончательного значения, вычисленного по гипотезе.

III. СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ

Применение нейронных сетей становится всё более широким. Особую популярность получили сверточные нейронные сети. Типовая архитектура сверточной нейронной сети изображена на рисунке 2.

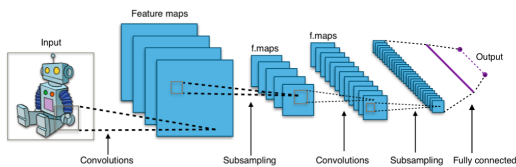


Рис. 2 – Типовая архитектура сверточной нейронной сети

Сверточные нейронные сети - специальная архитектура искусственных нейронных сетей.

Название сеть получила из-за наличия операции свертки, которая изображена на рисунке 3.

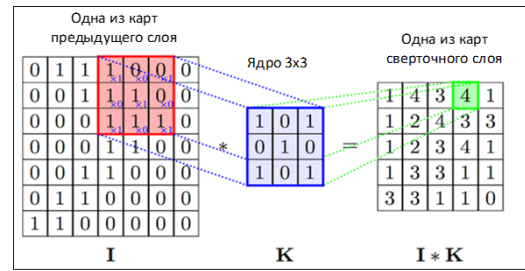


Рис. 3 – Операция свертки и получение значений сверточной карты

Особое внимание свёрточные нейронные сети получили после конкурса ImageNet, посвящённого распознаванию объектов на фотографиях. Победитель данного конкурса Алекс Крижевский, используя сверточную нейронную сеть, значительно превзошёл остальных участников. После успеха применения свёрточных нейронных сетей к классификации изображений привело к попыткам использования данного метода и к другим задачам. В последнее время их стали активно использоваться для задачи классификации текстов.

IV. ЗАКЛЮЧЕНИЕ

На сегодняшний день применение сверточных нейронных сетей не ограничивается анализом изображений. Их применяют для решения различных типов задач: анализа тональности, классификации текста, распознавания речи и многое другое. Благодаря применению сверточных нейронных сетей можно значительно сократить количество обучаемых параметров и получать высокое качество классификации.

СПИСОК ЛИТЕРАТУРЫ

1. Нестеренков, С. Н. Модифицированный генетический алгоритм для обучения нейронной сети / С. Н. Нестеренков, К. П. Белов // Информационные технологии и системы 2017 (ИТС 2017) : материалы междунар. науч. конф., Минск, 25 окт. 2017 г. / Белорус. гос. ун-т информатики и радиоэлектроники ; редкол.: Л. Ю. Шилин [и др.]. - Минск, 2017. - С. 204-205.
2. Нестеренков, С.Н. Применение больших данных в электронном образовании / С.Н. Нестеренков, М.И. Макаров, Н.В. Ющенко, А.Д. Радкевич // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня : сб. материалов V Междунар. науч.-практ. конф. (Республика Беларусь, Минск, 13-14 марта 2019 года). В 2 ч. Ч. 2 / редкол. : В. А. Богущ [и др.]. - Минск : ВГУИР, 2019. - С. 242-245.
3. Нестеренков, С.Н. Использование генетического алгоритма для нахождения коэффициентов и структуры нейронной сети / С.Н. Нестеренков, К.П. Белов // Информационные технологии и системы 2018 (ИТС 2018) : материалы междунар. науч. конф., Минск, 25 окт. 2018 г. / Белорус. гос. ун-т информатики и радиоэлектроники ; редкол.: Л. Ю. Шилин [и др.]. - Минск, 2018. - С. 124-125.
4. Yanyan W., Qun C., Jiquan S., Boyi H., Murtadha A, Zhanhuai Li G. Machine Learning for Aspect-level Sentiment Analysis // arXiv:1906.02502 – 2019.

TON: TELEGRAM OPEN NETWORK

Новицкий И. О., Шилин Л. Ю.

Кафедра систем управления, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: ionovitsky@gmail.com

ВВЕДЕНИЕ

Больше года назад стало известно о планах мессенджера Telegram выпустить собственную децентрализованную сеть Telegram Open Network. Тогда стал доступен объемный технический документ, который, предположительно, был написан Николаем Дуровым и описывал структуру будущей сети. Недавно в сеть был выложен документ с описанием технической составляющей новой сети TON

I. УРОВНИ СИСТЕМЫ

- TON Blockchain. Самый низкий уровень системы;
- TON P2P Network. Пиринговая сеть, на основе которой будет построена работа системы;
- TON Storage. Файловое хранилище, которое независимо от блокчейна будет построено на вышеупомянутой пиринговой сети;
- TON Proхy. Это сервис, цель которого повысить анонимность участников сети. Любой пакет можно отправить не напрямую, а через туннели-посредники с дополнительным шифрованием, подобно I2P или TOR;
- TON DHT. Распределенная хэш-таблица для хранения произвольных значений. Она тоже построена поверх TON Network и помогает TON Storage находить «раздающие» узлы, а TON Proхy — промежуточные ретрансляторы;
- TON Services. Платформа для произвольных сервисов. Обмен данными производится через TON Network/TON Proхy, а логика приложений создается в смарт-контрактах самого TON Blockchain;
- TON DNS. Преобразователь url адресов в 256-битные адреса — аккаунтов, контрактов, сервисов и узлов;
- TON Payments. Платежная часть системы, использует собственную валюту «грамм», также поддерживает возможность работать с валютой других блокчейн сетей.

II. ADNL: ABSTRACT DATAGRAM NETWORK LAYER (АБСТРАКТНАЯ ДАТАГРАММА СЕТЕВОГО УРОВНЯ)

Как и у в других распределенных сетях, работа сети TON поддерживается узлами. На низ-

ком уровне узлы имеют IPv4/IPv6-адреса и общаются по протоколу UDP, на более высоком — обладают абстрактными адресами и реализуют протокол ADNL. Чтобы один узел мог послать пакет другому, он должен знать один из его публичных ключей (и, следовательно, адрес, который им определяется). Он зашифровывает пакет этим ключом и добавляет в начало пакета 256-битный адрес получателя — поскольку один узел может иметь несколько таких адресов, это позволит ему определить, какой ключ использовать для расшифровки.

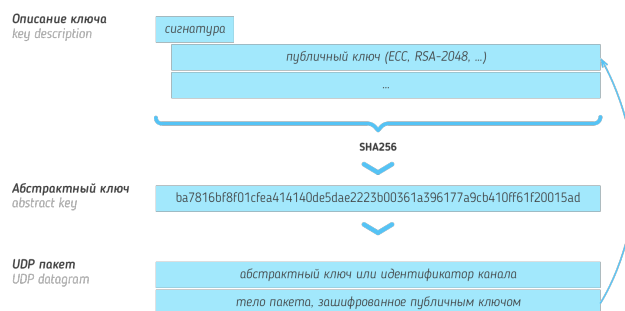


Рис. 1 – Процесс передачи сообщения от узла к узлу

Кроме того, вместо адреса получателя в начале пакета данных может находиться идентификатор канала. В таком случае обработка пакета уже зависит от конкретных договорённостей между узлами — например, отправленные в некий канал данные могут предназначаться другому узлу и должны быть ему переадресованы (это и есть сервис TON Proхy). Другим частным случаем может быть взаимодействие напрямую между узлами, но с шифрованием по индивидуальной паре ключей для этого канала.

Наконец, специальным случаем является нулевой канал — если узел ещё не знает публичных ключей соседних узлов, он может посылать им пакеты без шифрования совсем. Это предназначено только для инициализации — как только узлы пришлют информацию о своих ключах, их стоит использовать для дальнейшего взаимодействия.

Вышеописанный протокол (256 бит идентификатора канала + содержимое пакета) называется ADNL. Документация TON упоминает возможность реализации аналога TCP поверх него или собственной надстройки — RLDP (Reliable Large Datagram Protocol).

III. TON DHT: РАСПРЕДЕЛЁННАЯ ХЭШ-ТАБЛИЦА

Как в случае с другими распределёнными системами, TON предполагает реализацию DHT — распределённой хэш-таблицы. Более конкретно — таблица является Kademlia-подобной.

В абстрактном смысле, DHT ставит в соответствие 256-битным ключам некие бинарные значения произвольной длины. При этом ключи в таблице — это хэши от определённой TL-структуры (сами структуры тоже хранятся вместе с DHT). Это очень похоже на формирование адресов узлов — и они действительно могут присутствовать в DHT (например, по такому ключу может находиться IP-адрес узла соответствующего заданному абстрактному адресу, если он не скрывает его). Но в общем случае, прообразы ключей (их описания, *key descriptions*) — это метаданные, которые указывают на владельца записи в хэш-таблице (то есть публичный ключ какого-то узла), тип хранимого значения и правила, по которым эта запись может впоследствии изменяться. Например, правило может разрешать изменять значение только владельцу — или запрещать изменение значения в меньшую сторону (чтобы защититься от *geplay*-атак).

Кроме 256-битных ключей вводится понятие DHT-адресов. Разница с обычными адресами узлов в том, что DHT-адрес обязательно привязан к IP-адресу. Если узел не скрывает своего IP, он может использовать обычный адрес для DHT. Но чаще для нужд DHT будет заводиться отдельный, полу-постоянный адрес.

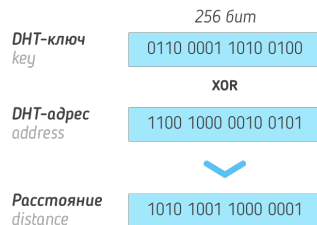


Рис. 2 — Формирование адреса получателя

Над ключами и DHT-адресами вводится понятие расстояния — в этом всё совпадает с таблицами Kademlia — расстояние между ключами равно XOR (побитовому исключающему ИЛИ) от них. Как и в таблицах Kademlia, значение, соответствующее некоему ключу, должно храниться на s узлах, имеющих наименьшее расстояние до этого ключа (s тут — относительно небольшое число).

Для того, чтобы узел DHT мог взаимодействовать с другими такими узлами, он держит в памяти таблицу маршрутизации DHT — DHT- и IP-адреса узлов, с которыми он взаимодействовал до этого, сгруппированные по расстоянию до них. Таких групп 256 (они соответствуют старшему выставленному биту в значении расстояния — то есть узлы на расстоянии от 0 до 255 попадут в одну группу, от 256 до 65535 — в следующую, и т.д.). Внутри каждой группы хранится ограниченное число «лучших» узлов (в плане пинга до них).

Каждый узел должен поддерживать несколько операций: сохранение значения для ключа, поиск узлов и поиск значений. Поиск узлов подразумевает выдачу по заданному ключу ближайших к нему узлов из таблицы маршрутизации; поиск значений — то же самое, за исключением ситуации, когда узлу известно значение для ключа (тогда он просто возвращает его). Соответственно, если узел хочет найти в DHT значение по ключу, он посылает запросы небольшому числу ближайших к этому ключу узлов из своей таблицы маршрутизации. Если среди их ответов нет искомого значения, но есть другие адреса узлов, то запрос повторяется уже к ним.

СПИСОК ЛИТЕРАТУРЫ

1. Николай Дуров. Telegram Open Network [Электронный ресурс] // URL: <https://test.ton.org/ton.pdf>

ПРОГНОЗИРОВАНИЕ ЗНАЧЕНИЙ ЭКСПЛУАТАЦИОННЫХ ПАРАМЕТРОВ ТЕЛЕКОММУНИКАЦИОННОГО ОБОРУДОВАНИЯ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

Олейников А. А., Сорокин А. А.

Кафедра математических и естественнонаучных дисциплин, Федеральное государственное бюджетное образовательное учреждение высшего образования "Волжский государственный университет водного транспорта"

Каспийский институт морского и речного транспорта

Астрахань, Россия

E-mail: a.oleynikov.astu@mail.ru, alsorokin.astu@mail.ru

В работе рассматривается возможность использования нейронных сетей прямого распространения и нечёткой нейронной сети для задачи прогнозирования эксплуатационных параметров оборудования операторов связи. Производится сравнение расхождений между реальными и прогнозными значениями сетей прямого распространения с 2 и 3 слоями и нечёткой нейронной сети полученными на основе ретроспективных данных. Получаемые показатели позволят более полно оценивать состояние оборудования оператора связи и повысить качество принимаемых решений в процессе проведения модернизации.

ВВЕДЕНИЕ

В процессе эксплуатации сетевого оборудования операторов связи возможно наступление аварийного, либо пред аварийного состояния. Если же оборудование отработало указанный, производителем срок, то неизбежно моральное устаревание и как следствие снижение производительности участка сети влекущее отток абонентов. Своевременно принятое решение о замене оборудования находящегося в предельном состоянии позволит не только остановить миграцию абонентов к другим операторам, но и привлечь новых, за счёт предоставления сервисов нового поколения[1]. Для принятия такого решения, необходимо обладать полнотой информации об оборудовании, в том числе о возможных его состояниях. Параметры этих состояний целесообразно получать за счёт прогнозирования. Рассматриваемые методы, основанные на нейронных сетях могут быть использованы для этой цели так как получают всё большее распространение для прогнозирования временных рядов, к которым относятся эксплуатационные параметры оборудования связи.

Цель работы, сравнительный анализ работы математического аппарата нейронных сетей прямого распределения и нечёткой нейронной сети для решения задачи прогнозирования эксплуатационных параметров телекоммуникационного оборудования.

I. ПОЛУЧЕНИЕ И ПОДГОТОВКА ДАННЫХ ДЛЯ АНАЛИЗА

Выбор источника и подготовка данных для анализа осуществляется с участием экспертов имеющих значительный опыт в области телекоммуникаций. Экспертной группе было предложено заполнить опросные листы для выбора параметров наиболее полно описывающих состояние

оборудования, оценив каждый по десятибальной шкале. По мнению экспертов комплексно оценить работоспособность узлового оборудования можно по скорости соединения. Остальные параметры, имеют меньшее влияние или коррелируют с выбранной величиной. Тем не менее за счёт их использования возможно повысить точность оценки состояния оборудования, так как будет происходить учёт факторов косвенного характера. Данные для анализа могут быть сняты с использованием программно аппаратных средств, входящих в состав систем поддержки бизнес процессов предприятия связи. Следующий этап, подготовка данных для анализа. Первоначальный объём, делится на обучающую, тестовую и обучающую выборку. На основе обучающей происходит тренировка нейронных сетей, а тестовая, позволяет определить расхождение между прогнозными значениями и результатами работы нейронной сети[2, 3].

II. НЕЙРОННЫЕ СЕТИ ПРЯМОГО РАСПРЕДЕЛЕНИЯ

В ходе моделирования были созданы две нейронные сети прямого распределения с 2 и 3 скрытыми слоями, с числом нейронов 20, 1 и 10, 4, 1 соответственно.

Таблица 1 – Результаты моделирования на 2-х слойной нейронной сети

Измерения	Прогноз Мбит/сек	Реальные данные Мбит/сек	Расхождение %
1.	8269200	8282203	0.16
2.	8262900	8279528	0.20
3.	8298000	8282608	0.19
4.	8273200	8283080	0.12

Проведённое моделирование показало, что точность прогнозирования увеличивается с ростом числа нейронов в сети. Однако вместе с

этим растут требования к аппаратным ресурсам для построения более сложных моделей (Таблица 1 и 2).

Таблица 2 – Результаты моделирования на 3-х слойной нейронной сети

Измерения	Прогноз Мбит/сек	Реальные данные Мбит/сек	Расхождение %
1.	8273900	8282203	0.10
2.	8288900	8279528	0.11
3.	8271100	8282608	0.14
4.	8273800	8283080	0.11

III. НЕЧЁТКАЯ НЕЙРОННАЯ СЕТЬ

В качестве альтернативы предлагается использовать нечёткую нейронную сеть Сугено, являющуюся структурой с многими слоями не имеющими обратных связей использующей нечёткие сигналы. Операции суммирования производятся на основе фиксированной Т-нормы и S-конормы или другой операции. Нечёткость в сетях этого типа проявляется при использовании ретроспективной информации, при формировании параметров функций принадлежности. Сами же параметры находятся на основе методов обучения нейронных сетей. На основе ретроспективной информации, были подготовлены три матрицы данных. Матрица обучающих данных, матрица тестовых данных и матрица проверочных значений. В процессе моделирования, была построена нечёткая нейронная сеть имеющая в своём составе четыре входные переменные, а так же по три термина в виде гауссовской функции.

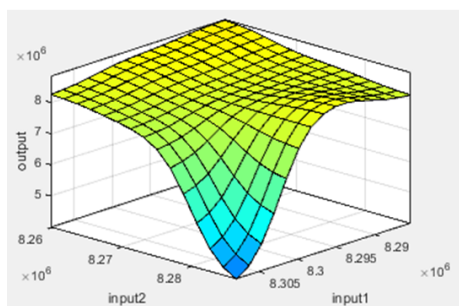


Рис. 1 – Зависимость выходной переменной нечёткой нейронной сети от входных переменных input2 и input1

На рисунках 1 - 3 показаны графики зависимостей выходного значения от четырёх входных переменных input1 - скорость доступа в первые сутки, Мб/сек; input2 - скорость доступа во вторые сутки, Мб/сек; input3 - скорость доступа в третьи сутки, Мб/сек; input4 - скорость доступа в четвёртые сутки, Мб/сек; полученные на базе правил нечёткой продукции, сформированных экспертной группой. Данные были получены от реального оборудования эксплуатируемого на се-

тях передачи данных одного из операторов связи.

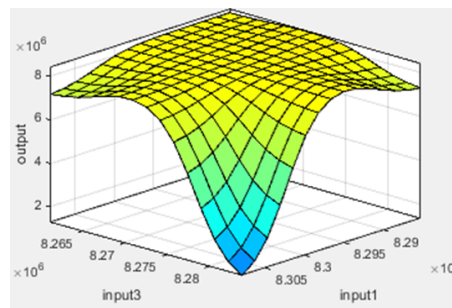


Рис. 2 – Зависимость выходной переменной нечёткой нейронной сети от входных переменных input3 и input1

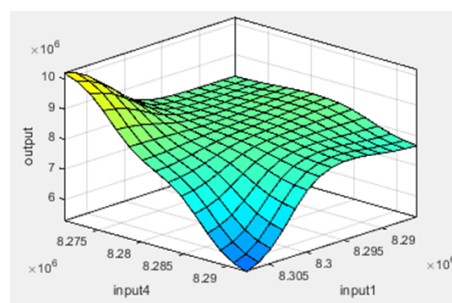


Рис. 3 – Зависимость выходной переменной нечёткой нейронной сети от входных переменных input4 и input1

Вывод нечёткой нейронной сети показал близость значений сформированных моделью и значения контрольной выборки (Таблица 3).

Таблица 3 – Результаты моделирования нечёткой нейронной сети

Прогноз Мбит/сек	Реальные данные Мбит/сек	Расхождение %
8289900	8283075	0.08

Наименьшее расхождение у сетей с 2 и 3 слоями составило 0.12% и 0.10% соответственно, лучший результат прогнозирования показывает нечёткая нейронная сеть 0.08%. По аналогии возможно прогнозирование других параметров оборудования. Полученные, с помощью нечёткой нейронной сети, прогнозы целесообразно использовать при планировании работ по замене узлового оборудования оператора связи в процессе устранения предаварийных ситуаций или планирования модернизации. В общем случае это позволит повысить готовность сетей связи для предоставления современных услуг.

СПИСОК ЛИТЕРАТУРЫ

1. Вишневецкий, В. М. Теоретические основы проектирования компьютерных сетей / В. М. Вишневецкий. – М.: Техносфера, 2000. – 520 с.
2. Тарик, Р. Создаём нейронную сеть / Р. Тарик. – М.: Вильямс, 2017. – 272 с.
3. Хайкин, С. Нейронные сети. Полный курс. / С. Хайкин. – М.: Вильямс, 2016. – 1104 с.

ИСПОЛЬЗОВАНИЕ МАКРОСОВ ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССОВ ИНЖЕНЕРНЫХ САЕ-СИСТЕМ

Петров А. В., Фролов И. И.

Кафедра информатики, Кафедра электронных вычислительных машин, Белорусский государственный университет информатики и радиоэлектроники
220013, П. Бровка, 6, Минск, Республика Беларусь
E-mail: petrov@tut.by, frolov@bsuir.by

В докладе рассмотрены проблемы автоматизации процессов в инженерных системах для решения задач моделирования и конструирования. Проанализированы ключевые требования, структура и особенности работы. Исследована и определена практическая необходимость упрощения визуального восприятия отдельных этапов при создании и запуске сценариев. На основе проведенного исследования авторами представлены алгоритмы использования скриптов в разрабатываемой системе, дается детальное описание их работы и полученные характеристики.

ВВЕДЕНИЕ

С развитием технического прогресса упростился процесс создания инженерных изделий и проектов. Повысилось качество продукта, количество создаваемых и выпускаемых изделий. Современные многофункциональные инженерные системы предоставляют пользователям достаточно многообразные инструменты. Однако гибкость в реализованных продуктах может требовать управления множеством настроек и параметров, требующих в свою очередь дополнительной автоматизации.

1. ОСОБЕННОСТИ РАБОТЫ САЕ-СИСТЕМЫ

Современные системы инженерного анализа (или системы автоматизации инженерных расчётов) (САЕ – Computer-aided engineering) применяются совместно с САД-системами (САД – Computer-aided design), зачастую интегрируются в них, в этом случае получают гибридные САД/САЕ-системы [1]. САЕ-системы – это разнообразные программные продукты, позволяющие при помощи методов моделирования, численных методов, методов анализа данных (метод конечных элементов, метод конечных разностей, метод конечных объёмов) оценить, как поведёт себя компьютерная модель проектируемого изделия в реальных условиях эксплуатации; помогают убедиться в работоспособности изделия, без привлечения больших затрат времени и средств [2].

При использовании САЕ-систем инженерам приходится выполнять много монотонных и единообразных действий для выполнения даже простых, на первый взгляд, операций. Конечные пользователи предпочитают сокращать время разработки, концентрируясь на вопросах более высокого уровня, при этом используя предоставляемые макросы и скрипты, объединяющие низкоуровневые операции для выполнения стандартных процедур проектирования.

Определенные программные пакеты не поддерживают запись макросов и работу с ними «из

коробки». Такой подход характерен для коммерческих систем, не допускающих изменения исходного кода и не предоставляющих инструментов для создания «плагинов» поверх исходного программного обеспечения.

В мире проектирования и моделирования инженерных систем достаточно широкое распространение получил продукт HyperWorks. Система обладает богатыми возможностями для построения моделей и проектирования, однако, в то же время, не позволяет выполнять объединение элементарных операций, что определяет актуальность разработки дополнительного интеграционного слоя, решающего данную задачу.

В данной работе представлено описание разрабатываемого программного обеспечения, интегрированного с системой HyperWorks. В обсуждаемом программном обеспечении использованы скриптовые языки для решения задач объединения множества команд с последующей разработкой интеграционного слоя и UI для конечного пользователя – инженера-конструктора.

Достаточно сложным в работе САЕ-систем, также как и других программируемых системах является необходимость описывать в деталях фактически каждое минимальное действие. Конечно, такой уровень детализации увеличивает гибкость и расширяет функциональность, но с другой – увеличивает время разработки, не позволяя сконцентрироваться на высокоуровневых задачах.

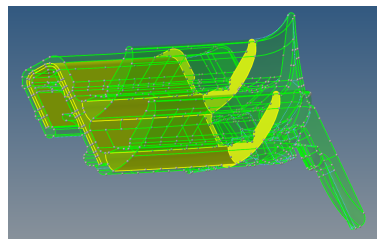


Рис. 1 – Образец проектируемого изделия на стадии нанесения объекта сетки

Одной из типовых задач является выбор смежных поверхностей для нанесения на по-

следующих шагах проектирования Mesh-объекта (сетки), пример сложного объекта приведен на рисунке 1. Данная операция не является тривиальной с точки зрения ее воспроизведения и описания в САЕ-системе, и занимает, в зависимости от конкретного программного пакета, опыта и квалификации инженера, по разным оценкам от 40 до 70 минут.

Автоматизированный скрипт, содержащий набор «элементарных» действий, составляющих совокупность шагов алгоритма выбора смежных поверхностей, и построенных на использовании базовых циклических конструкций и условных операций сокращают вышеуказанный временной интервал до 5-8 минут. Такой выигрыш во времени не только позволяет сократить объем рутинных операций и ускорить процесс проектирования, но и позволяет перенести основное внимание на более высокий уровень восприятия проекта в целом.

После завершения работы алгоритма поиска всех смежных плоскостей [3] в разрабатываемом проекте, система может в таком же автоматизированном режиме, с использованием заранее запрограммированного скрипта выполнить автоматическое нанесение сетки по заранее заданным критериям проектировщика. Более того, данные параметры могут быть также внесены в файлы с кодом скрипта и использоваться по умолчанию при нанесении сетки, либо же можно задать их явно.

Следующий скрипт выполняет последовательный проход по всей сетке в поисках «плохих» элементов, которые определяются по результатам сравнения со скриптом эталона. В случае, если элемент не удовлетворяет заданным параметрам и имеет отклонение от нормы более определенной допустимой погрешности, то такой элемент заносится в специальный «set bad elements». Далее, указанный набор подвергается повторной обработке, либо, если проект позволяет исключить «подозрительные» элементы – они подлежат удалению.

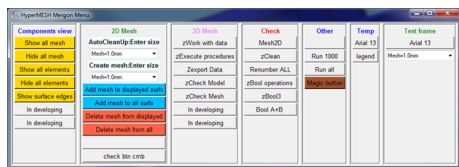


Рис. 2 – Меню автоматизации выполняемых команд в HyperWorks

Сам процесс проектирования адаптирован для удобной работы с использованием пользовательского интерфейса и операции инкапсулированы и связаны либо напрямую с пунктами меню, либо пользователю предоставляется возможность загрузить и использовать в системе самостоятельно написанный макрос, обобщающий требуемые операции. Пример пользовательского меню представлен на рисунке 2.

II. ИСПОЛЬЗОВАНИЕ МАКРОСОВ

Использование языков программирования C#, Python, встроенного и поддерживаемого в HyperWorks TCL/TK 8.5 (использовался в создании графического интерфейса и взаимодействии с оптимизированными и автоматизированными процедурами) позволяет снизить порог вхождения в разработку скриптов и увеличивает возможность использования системой макросов пользователями с разным набором знаний.

Алгоритм работы системы автоматизации заключается во взаимодействии пользователя посредством графического интерфейса с ядром комплекса, но с обновленными и оптимизированными файлами (имеют расширение .tcl), содержащими уже расширенный функционал. После запроса в базу с файлами и её отклика, пользователь получает результат. Схема работы продемонстрирована на рисунке 3.

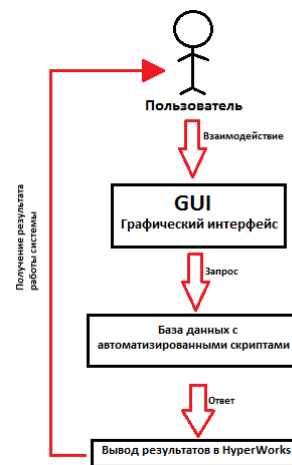


Рис. 3 – Обобщенная схема работы системы автоматизации в HyperWorks

ЗАКЛЮЧЕНИЕ

Перспективой развития автоматизации САПР является тесная интеграция с программами смежных направлений. Основной целью создания данной системы является упрощение восприятия пользователем графического интерфейса, разгрузке от дополнительных и неиспользуемых команд, быстрой настройке нужных команд, упрощении алгоритмов и улучшением их быстродействия, настройке команд под конкретную задачу (разбиение по блокам).

СПИСОК ЛИТЕРАТУРЫ

1. Латышев, П. Н. Каталог САПР. Программы и производители: Каталогное издание / П. Н. Латышев. – Москва: ИД СОЛОН-ПРЕСС. – 2011. – С. 700–736.
2. Малох, В. Н. Введение в современные САПР: Курс лекций / В. Н. Малох. – Москва: ДМК Пресс. – 2010. – С. 180–188.
3. Madsen, A. D. Engineering Drawing & Design / David A. Madsen. – Clifton Park, NY: Delmar. – 2012. – P. 5–20.

ИНФОРМАЦИОННЫЕ СИСТЕМЫ ИДЕНТИФИКАЦИИ РУКОПИСНОЙ ПОДПИСИ

Пинтусов Д. Н., Латушко Д. И.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: dzmitrypintusov@gmail.com, denis.latushko92@gmail.com

Рукописная подпись является наиболее распространенным способом биометрической проверки личности. Данный способ более привычен людям, в сравнении с проверкой отпечатков пальцев и радужной оболочки, а также не требует значительных финансовых вложений в оборудование и найм специалистов. В Республике Беларусь с марта 2018 года введен новый способ подписания (подтверждения) документа в электронном виде – цифровая рукописная подпись. Использование новых технологий предполагает разработку и внедрение информационных систем идентификации рукописной подписи.

ВВЕДЕНИЕ

Рукописная подпись была отличительной чертой для идентификации личности на протяжении веков, так как она биологически связана с конкретным человеком. Задачи по распознаванию подписи относят к вопросам биометрической аутентификации (или биометрии) – процессу идентификации человека по его характеристикам или признакам. Биометрическую аутентификацию можно разделить на две категории: поведенческую (проверка подписи, динамики нажатия клавиш) и физиологическую (характеристики радужной оболочки, отпечатки пальцев). Такие способы, как проверка отпечатков пальцев и радужной оболочки требуют установки дорогостоящего оборудования и, следовательно, не могут быть использованы в повседневных местах, таких как банки, государственные учреждения и прочие, где процесс идентификации личности является обязательным для осуществления некоторых операций. Хотя состав рукописной подписи юридически не установлен, банки и государственные органы признают подпись как законное средство аутентификации.

В Республике Беларусь с марта 2018 года введен новый способ подписания (подтверждения) документа в электронном виде – цифровая рукописная подпись. Планируется, что цифровой способ подписания документов будет способствовать развитию финансовых услуг, предоставляемых банками посредством цифровых технологий, а также сокращению бумажного документооборота. Стоит отметить, что людям легче перейти с использования привычного способа подписания документов на бумаге на тот, где рукописная подпись онлайн регистрируется и проверяется электронным способом, по сравнению с другими электронными методами идентификации. Использование новых технологий предполагает разработку и внедрение информационных систем идентификации рукописной подписи [5]. Таким образом, все больше проявляется интерес к идентификации личности на основе систем проверки рукописной подписи.

I. ТЕХНОЛОГИЯ ПРОВЕРКИ ПОДПИСИ

Технология проверки подписи использует отличительные аспекты подписи личности для идентификации человека. Графологические эксперты обычно сравнивают подозрительную подпись с несколькими примерами известных оригиналов подписей. Они ищут признаки подделки, которые включают: скорость написания (медленнее или быстрее, чем подлинный вариант); частота изменения пишущего инструмента; закругление линий окончания и начала; плохое качество линии с колебаниями и дрожанием линии; восстановление и исправление ошибок. Проведение графологической экспертизы усилиями человека иногда не способно в полной мере оценить различие предоставленной подписи с подлинным вариантом, а также не может быть применено повсеместно, поэтому в настоящее время все больше прилагаются усилия для разработки компьютеризированных алгоритмов, которые могли бы проверять и/или удостоверять личность человека на основе предоставленного экземпляра подписи.

Существует два подхода к проверке подписи: онлайн и оффлайн, различающиеся по способу сбора данных. В автономном случае подпись получается на листе бумаги, а затем сканируется. Проверка подписи в автономном режиме относится к записи статического 2D-изображения подписи. Это полезно при автоматической проверке подписи на банковских чеках и проверке подлинности документов. Методы автономной проверки основаны на ограниченной информации, доступной только из формы и структурных характеристик изображения подписи. Фундаментальной проблемой в области распознавания подписи в автономном режиме является отсутствие существенного представления формы и факторов ее происхождения. Онлайн системы проверки подписи более точны. Они требуют присутствия автора как во время получения справочных данных, так и в процессе проверки. В онлайн режиме рукописная подпись обычно получается с использованием электронного планше-

та и ручки. Проверка подписи в режиме онлайн отслеживает путь и другие переменные (факторы) относительно определенного промежутка времени. Автоматическая онлайн-проверка подписи представляет собой сложную интеллектуальную систему со многими практическими приложениями. Эта технология исследует поведенческие компоненты сигнатуры, такие как: порядок удара, скорость и давление, в отличие от сравнения визуальных изображений сигнатур. В отличие от традиционных технологий сравнения подписи, онлайн проверка подписи позволяет измерить физическую активность личности [2-4].

II. ИДЕНТИФИКАЦИИ РУКОПИСНОЙ ПОДПИСИ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

Существуют различные технические методы реализации проверки рукописной подписи, такие как сопоставление локальных экстремумов, алгоритм аппроксимации кривыми Безье, различные реализации на основе нейронных сетей. Основной задачей является нахождения наиболее эффективного алгоритма для максимального приближения уровня успешного распознавания подписи. Искусственные нейронные сети могут выполнять роль классификатора, хорошо моделирующего сложную функцию распределения рукописных подписей на изображениях, тем самым увеличивая точность распознавания по сравнению с остальными методами.

Этапы работы информационной системы идентификации рукописной подписи во многом связаны с выбранным подходом, и в случае решения задач с помощью нейронных сетей являются:

1. Сбор данных для обучения нейронной сети;
2. Подготовка и нормализация данных;
3. Выбор топологии сети;
4. Экспериментальный подбор характеристик сети;
5. Экспериментальный подбор параметров обучения;
6. Собственно обучение;
7. Проверка качества обученной сети;
8. Корректировка параметров, окончательное обучение;
9. Оптимизация и подготовка сети с целью дальнейшего использования.

Для распознавания подписи чаще рационально использовать подход «обучение с учителем» так как предполагается что образец подписи заранее предоставлен. Сверточная нейронная сеть является одним из лучших алгоритмов

по распознаванию и классификации изображений [1].

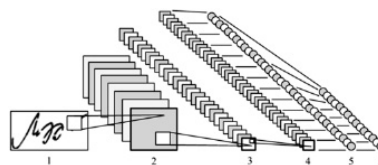


Рис. 1 – Архитектура свёрточной нейронной сети для распознавания рукописных подписей

Идея сверточных нейронных сетей заключается в чередовании сверточных слоев и субдискретизирующих слоев (слоев подвыборки). Структура сети является однонаправленной (без обратных связей), принципиально многослойная. Для обучения используются стандартные методы, чаще всего метод обратного распространения ошибки. Преимуществами данного подхода является удобное распараллеливание вычислений, а следовательно, возможность реализации алгоритмов работы и обучения сети на графических процессорах, относительная устойчивость к повороту и сдвигу распознаваемого изображения, обучение при помощи классического метода обратного распространения ошибки. К недостаткам данного подхода можно отнести слишком большое количество варьируемых параметров сети, что может существенно влиять на результат.

СПИСОК ЛИТЕРАТУРЫ

1. Gideon, S. J. Handwritten Signature Forgery Detection using Convolutional Neural Networks / S. J. Gideon, A. Kandulna // *Procedia Computer Science*. – 2018. – Vol. 143. – P.978-987.
2. Fahm Maged, M. M. Online handwritten signature verification system based on DWT features extraction and neural network classification / M. M. Fahm Maged // *Ain Shams Engineering Journal*. – 2010. – Vol. 1, № 1. – P. 59–70.
3. Bhattacharya, I. Offline Signature Verification Using Pixel Matching Technique / I. Bhattacharyaa, P. Ghoshb, S. Biswas // *Procedia Technology*. – 2013. – Vol. 10.. – P. 970–977.
4. Mangrum, J. M. The evaluation and management of bradycardia / J. M. Mangrum, J. P. DiMarco // *N. Engl. J. Med*. – 2000. – Vol. 342, № 10. – P. 703–709.
5. О внесении изменений и дополнений в постановление Правления Национального банка Республики Беларусь от 6 мая 2016 г. № 241 [Электронный ресурс]: постановление Правления Национального банка Республики Беларусь, 19 февр. 2018 г., № 65 // Национальный правовой Интернет-портал Республики Беларусь. – Минск, 2019. – Режим доступа: <http://www.pravo.by>. – Дата доступа: 02.10.2019.

МЕТОДЫ БОРЬБЫ СО СПАМОМ В VOIP ТЕЛЕФОНИИ

Полудворянин С. М., Нестеренков С. Н.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: siarheipoludvaranin@gmail.com, nsn@bsuir.by

Стремительное развитие сервисов VoIP телефонии в последние годы привело к появлению нового вида спама: навязчивых телефонных звонков рекламного характера. В англоязычной литературе это явление известно под термином SPIT - spam over Internet telephony. Каждый год операторы связи и правоохранительные органы получают огромное количество жалоб на эти нежелательные звонки. Данный вид спама не только приносит финансовые потери пользователям телефонии, но также раздражает их нежелательными оповещениями. По этой причине операторам связи важно блокировать спамеров телефонии на уровне сети, чтобы завоевать доверие своих клиентов. В этой статье мы рассмотрим некоторые подходы для борьбы с нежелательными звонками на уровне сети.

ВВЕДЕНИЕ

На данный момент существует множество подходов для борьбы с телефонным спамом. Эти подходы могут быть сгруппированы в несколько категорий. К первой категории можно отнести подходы, основанные на контентном анализе. Они обрабатывают речевой поток между участниками звонка и блокируют абонентов, если обнаруживается, что поток содержит нежелательные сообщения. Ко второй категории относятся подходы, основанные на списках. Они составляют и поддерживают базу данных черных, белых и находящихся под наблюдением пользователей. Третья категория - подходы, основанные на IVR (Interactive Voice Response - система предварительно записанных голосовых сообщений), которые требуют от вызывающего абонента решить какую-либо задачу. Наконец, к четвертой категории относятся подходы, основанные на репутации системы, которые вычисляют репутацию вызывающего абонента путем получения обратной связи от вызываемого абонента или используют данные предыдущих вызовов.

I. СПИСОЧНЫЕ ПОДХОДЫ

Подходы на основе списков доступа авторизуют личность вызывающего с помощью локальной и глобальной базы данных. Вариант черного списка содержит базу абонентов, звонки от которых будут заблокированы. Абоненты вне списка могут совершать звонки без ограничений. Как и в случае с E-Mail спамом, черные списки для интернет-телефонии мало эффективны из-за простоты подмены идентификаторов звонящего в большинстве протоколов сигнализации, применяемых в VoIP. Вариант белого списка - обратный черному. Он содержит доверенных абонентов, вызовы от которых могут быть авторизованы. В отличие от черных списков, спамер не может изменить идентификатор аккаунта, чтобы обойти белый список. Тем не менее, белые списки не являются полноценным решением пробле-

мы, так как они будут запрещать звонки от абонентов, ранее не звонивших пользователю, т. е. тех, кто не был явно включен в белый список. В результате белые списки требуют решения проблемы представления - каким образом авторизовать кого-то в первый раз и решить, должен ли он быть помещен в белый список. Серые списки содержат абонентов, которые находятся под наблюдением из-за подозрительного поведения. Основанные на списке подходы обычно реализуются в сочетании с другими подходами, которые отвечают за принятие решения о том, помещать ли вызывающего абонента в белый, серый или черный список [1].

II. АУТЕНТИФИКАЦИИ АБОНЕНТА

Подходы, основанные на аутентификации, реализуют механизм проверки звонящего перед тем, как авторизовать звонок. Для фильтрации спам-звонков, распространяемых с помощью ботов, был предложен механизм обеспечения доверия, который позволяет вызывающему абоненту посредством IVR решать сложные головоломки. Тесты такого типа известны под термином CAPTCHA - Completely Automated Public Turing test to tell Computers and Humans Apart — полностью автоматизированный публичный тест Тьюринга для различения компьютеров и людей. Такой подход может быть успешным при блокировании вызовов, генерируемых компьютером, но для обработки большого количества одновременных вызовов потребуются сетевые ресурсы. Кроме того, решение головоломок может увеличить время установления вызова и создать дополнительные трудности для легитимного абонента, вынужденного решать головоломки при каждой попытке вызова. Другим примером аутентификации может служить голос звонящего абонента. Система хранит в базе данных известные голоса спам-ботов, анализирует входящий звуковой поток и блокирует звонок при обнаружении совпадений.

III. РЕПУТАЦИЯ АБОНЕНТА

Подходы, основанные на репутации, используют социальные отношения между участниками звонка для оценки репутации звонящего. Прямое доверие между абонентами обеспечивает доверие вызывающего абонента к взаимодействию с вызываемым абонентом. Глобальный рейтинг репутации играет важную роль, когда вызываемый абонент получает вызов от неизвестного контакта, и опирается на совокупный рейтинг с другими абонентами, которые уже взаимодействовали со звонящим. В VoIP телефонии доверие между конкретными абонентами и глобальная репутация могут быть вычислены двумя способами: с использованием обратной связи вызываемого абонента и с помощью информации из CDR (Call Detail Record - подробная запись о вызове). Во втором подходе репутация вызывающего абонента может быть вычислена из средней продолжительности звонка, количества коротких и длинных звонков, таких свойств социальных сетей, как количество контактов пользователя, коэффициент локальной кластеризации, индекс сплоченности, количество входящих и исходящих звонков пользователя (алгоритм SymRank - адаптация алгоритма Google PageRank) [2].

Существуют исследования, где глобальная репутация вычисляется с использованием алгоритма репутации Eigentrust. Это алгоритм, позволяющий вычислить репутацию каждого узла в сети на основе истории взаимодействия между узлами сети. Историю взаимодействия можно представить как матрицу связности между узлами, в каждой ячейке которой хранится оценка доверия к узлу, представленная как разность между количеством успешных и неуспешных попыток передачи данных.

Для расчета репутации также часто применяются алгоритмы машинного обучения, в частности, обучение с частичным привлечением учителя - способ машинного обучения, разновидность обучения с учителем, которое также использует неразмеченные данные для тренировки — обычно небольшое количество размеченных данных и большое количество неразмеченных данных. Примером имплементации может служить система на основе алгоритма MRSC-Means. Система кластеризует спам-звонки и легитимные звонки по некоторым свойствам, после чего «учитель» отмечает какой из кластеров является спамом. Свойством звонка в данном случае может быть как информация, полученная из сигнального трафика, такая как номер звонящего, направление звонка, длительность звонка, а также данные, полученные из медиа трафика, такие как длина пауз между словами. Однако такой подход требует обратной связи с пользователем, а также дополнительной настройки клиентских устройств [3].

Данту Р. и Колан П. в своих исследованиях представили многоступенчатую систему обнаружения спама, основанную на трех модулях: модуль доверия и репутации, модуль агрегирования обратной связи и модуль черного и белого списков. Доверие вычисляется путем получения прямой обратной связи от вызываемого абонента, а репутация вызывающего абонента вычисляется с использованием функции байесовского вывода.

Еще один подход, основанный на репутации, разработали Мохаммед Азад и Риккардо Морла. Для вычисления прямого доверия между абонентами он использует количество партнеров абонента для исходящих звонков, количество повторяющихся исходящих звонков и их длительность, количество входящих звонков и их длительность. Для подсчета глобальной репутации используется метод степенных итераций. Входными данными алгоритма является матрица нормализованных значений прямого доверия между каждой парой пользователей. Вызывающий получает высокую репутацию в сети, если ему удастся иметь хорошие оценки прямого доверия с большим количеством вызываемых абонентов. Репутация снижается, если у него небольшие оценки доверия с большим количеством вызываемых [4].

Методы, основанные на репутации, также могут применяться в сочетании с другими подходами обнаружения спама, которые являются многоступенчатыми и взаимодействуют с другими этапами для принятия окончательного решения о вызывающем абоненте.

ЗАКЛЮЧЕНИЕ

В данной статье мы рассмотрели основные подходы борьбы с спамом в VoIP телефонии. Более детально описали подходы основанные на вычислении репутации абонентов сети. К сожалению, нет никакого универсального способа для предотвращения спама в VoIP телефонии так же, как не существует такого решения для борьбы со спамом в электронной почте. Тем не менее, сочетание нескольких методов может обеспечить основу для борьбы со спамом в VoIP сетях.

СПИСОК ЛИТЕРАТУРЫ

1. The Session Initiation Protocol (SIP) and Spam (RFC 5039) [Electronic resource] / Internet Engineering Task Force. – Mode of access: <http://tools.ietf.org/html/rfc5039> – Date of access: 20.09.2019.
2. Bokharai, H. K. You can SPIT, but you can't hide: Spammer identification in telephony networks / H. K. Bokharai, A. Sahraei [etc.] // 2011 Proceedings IEEE INFOCOM – 2011. – P. 41–45.
3. Wu, Y. S. Spam detection in voice-over-IP calls through semi-supervised clustering / Y. S. Wu, S. Bagchi [etc.] // IEEE/IFIP International Conference on Dependable Systems Networks – 2009. – P. 307–316.
4. Azad, A. M. Caller-REP: Detecting unwanted calls with caller social strength / A. M. Azad, R. Morlla // Computers & Security – 2013. – Vol. 39 – P. 219–236.

АЛГОРИТМ ПОСТРОЕНИЯ СЕМАНТИЧЕСКОЙ СЕТИ С ОБРАБОТКОЙ МЕСТОИМЕНИЙ ТЕКСТА, ОСНОВАННЫЙ НА ДИАЛОГЕ С ПОЛЬЗОВАТЕЛЕМ

Потараев В. В.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: vic229@rambler.ru

На протяжении последних десятилетий количество текстовой информации, хранимой в электронном виде, значительно выросло. Поэтому автоматизированная обработка текстов является актуальной. Многие тексты содержат местоимения, что приводит к многозначности. Рассмотрим алгоритм построения семантической сети для многозначного текста, основанный на диалоге с пользователем.

ВВЕДЕНИЕ

К настоящему моменту количество информации, накопленной человечеством, довольно велико. Неструктурированная текстовая информация, или обычный текст, является одним из наиболее распространённых видов информации [1]. В связи с этим автоматизированная обработка текстов является актуальной. Семантический анализ является методом, позволяющим достаточно эффективно обрабатывать информацию за счёт выделения смысловых связей между различными понятиями.

Целью данной работы является построение методов автоматизированной обработки данных, основанных на семантическом анализе.

I. ЗАДАЧА АВТОМАТИЗИРОВАННОГО ПОСТРОЕНИЯ СЕМАНТИЧЕСКОЙ СЕТИ

Информационные системы довольно часто используются для ответа на некоторый запрос [2]. Для решения данной задачи может быть использован алгоритм, основанный на представлении текста в виде семантической сети. В данной сети узлами являются слова текста, а связями - семантические отношения между словами [3].

II. ОБРАБОТКА МЕСТОИМЕНИЙ ТЕКСТА

При обработке текста важно учитывать смысл всех его слов. Так, в естественном языке зачастую используются местоимения, и их корректный учёт важен для более полного извлечения информации из текста.

Местоимением называют самостоятельную часть речи, которая используется вместо существительных, прилагательных, числительных и наречий (или их характеристик), чтобы указать на эти существительные, прилагательные, числительные и наречия (а также их признаки и количество), не называя их. Личные местоимения указывают на объект речи - лицо, о котором идет речь. Местоимения 1 (я / мы) и 2 (ты / вы) лица обозначают участников речи. Местоимения

3 лица (он, она, оно / они) указывают на лиц, которые в речи не участвуют [4].

Одни и те же местоимения в каждом конкретном случае могут означать различные понятия. Поэтому при построении семантической сети имеет смысл заменять местоимения на те слова, которые они означают. Для обработки личных местоимений может быть использован специальный алгоритм [5]. Он основан на поиске предыдущего существительного, согласованного с местоимением по форме (род и число).

Пример построения семантической сети с обработкой местоимений представлен на рисунке 1. Сеть построена для текста, состоящего из двух предложений: «Лампа стоит на столе. Она освещает всю комнату». При её построении набор используемых связей ограничен связями «подлежащее-сказуемое» и «место», поэтому в сеть добавлены не все слова текста.

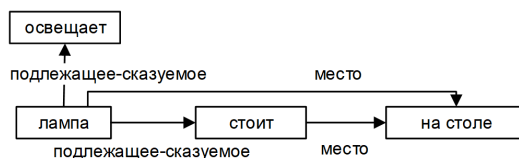


Рис. 1 – Пример семантической сети

III. МНОГОЗНАЧНОСТЬ ТЕКСТА, СОДЕРЖАЩЕГО МЕСТОИМЕНИЯ

При автоматизированной обработке неструктурированной текстовой информации (т.е. обычного текста) могут возникнуть проблемы, связанные с неоднозначностью естественного языка. Рассмотрим пример текста: «Книга лежит в сумке. Она покупалась в подарок.»

После прочтения данного текста нельзя точно сказать, какой из предметов был куплен в подарок. Очевидно, в некоторых ситуациях можно сделать вывод о покупке конкретного предмета на основании некоторой заранее известной дополнительной информации (например, может быть известно, что сумка была куплена для себя); но такая информация доступна далеко не всегда.

Таким образом, при построении семантической сети в данном случае мы не можем точно сказать, о каком из предметов идёт речь.

IV. РАЗРЕШЕНИЕ МНОГОЗНАЧНОСТИ МЕСТОИМЕНИЙ

Модифицируем модель семантической сети, чтобы учитывать многозначность. Для этого при построении сети будем использовать особый вид связей - назовём их «вероятными».

Для приведённого примера текста с многозначностью получится семантическая сеть, представленная на рисунке 2.



Рис. 2 – Пример семантической сети, построенной для текста с многозначностью

Две связи, образованные в результате обработки местоимения, для удобства последующей обработки помечены как относящиеся к одной группе (группа 1).

Для уточнения, какая из вероятных связей действительно имеет место, можно задать вопрос пользователю. Очевидно, что используя союз «или» и информацию, содержащуюся в сети, можно сформулировать вопрос: «Книга или сумка покупалась в подарок?». Ответ пользователя позволяет вместо группы вероятных связей оставить одну «достоверную» связь.

Алгоритм построения семантической сети с обработкой личных местоимений 3 лица может быть представлен следующим образом:

1. Перейти к очередному предложению текста (начиная с первого).
2. Найти в предложении существительные. Для каждого существительного определить его тип (род и число).
3. Найти в предложении личное местоимение 3 лица (он, она, оно, они).
4. Если личное местоимение было найдено, то загрузить существительные из предыдущего предложения, соответствующие ему по типу (род и число). Каждое из них является вероятным подлежащим.
5. Найти сказуемое и добавить его в сеть, а также добавить в сеть соответствующие связи «подлежащее-сказуемое» для каждого вероятного подлежащего. Если таких подлежащих несколько, то связи помечаются как вероятные и относящиеся к одной группе.
6. Добавить остальные виды связей, на основе которых строится сеть.
7. Повторять шаги 1-6 для каждого предложения в тексте.

8. Перейти к очередной группе вероятных связей.
9. Составить вопрос из подлежащих, входящих в группу, а также соответствующего сказуемого и обстоятельств.
10. Отобразить вопрос пользователю и получить ответ.
11. В зависимости от полученного ответа оставить только одну связь «подлежащее-сказуемое» и пометить её как достоверную. Остальные связи данной группы необходимо удалить из сети.
12. Повторять шаги 8-11 для каждой группы вероятных связей в семантической сети.

Таким образом, постановка вопросов перед пользователем позволяет построить корректную семантическую сеть для многозначного текста. Пользователю стоит один раз ответить на вопрос, и тогда последующие пользователи смогут получить ответ много раз. Если ответ не известен, в программном средстве можно предусмотреть возможность пропустить вопрос.

V. ВЫВОД

Семантическая сеть является моделью, которая довольно часто используется при обработке информации. Автоматизированное построение семантической сети позволяет значительно ускорить обработку информации, однако для текста с многозначностью может быть необходимо вмешательство человека.

Предложен алгоритм, который позволяет разрешить неоднозначность местоимений при помощи формулировки вопроса пользователю. Аналогичный подход может быть использован и при разрешении других неоднозначностей, которые могут встречаться в обрабатываемом тексте.

СПИСОК ЛИТЕРАТУРЫ

1. Усталов, Д. А. Модели, методы и алгоритмы построения семантической сети слов для задач обработки естественного языка: дис. канд. физ.-мат. наук. Институт математики и механики им. Н. Н. Красовского Уральского отделения Российской академии наук, Екатеринбург, 2017.
2. Информационные системы и сети. [Электронный ресурс]. – Электронные данные. – Режим доступа: https://tsput.ru/res/informat/sist_seti_fmolekci/lekci/lekcij-1.html. Дата доступа : 08.10.2019.
3. Потараев, В. В. Алгоритм применения семантической сети для поиска ответа на вопрос / В. В. Потараев // Компьютерные системы и сети: Материалы 54-ой научной конференции аспирантов, магистрантов и студентов. – Минск : БГУИР, 2018. – С. 103–105.
4. Все, что вы хотели узнать о местоимениях, но не знали, как спросить [Электронный ресурс]. – Электронные данные. – Режим доступа: <https://blog.tutoronline.ru/vse-cto-vy-hoteli-uznat-o-mestoimenijah-no-ne-znali-kak-sprosit>. Дата доступа : 08.10.2019.
5. Потараев, В. В. Алгоритм построения семантической сети с обработкой местоимений текста / В. В. Потараев // Компьютерные системы и сети: Материалы 55-ой юбилейной научной конференции аспирантов, магистрантов и студентов. – Минск : БГУИР, 2019. – С. 147–148.

ПРИМЕНЕНИЕ МЕТОДОВ ВЕРОЯТНОСТНОГО ТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ ДЛЯ АНАЛИЗА ДЕФЕКТОВ В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Романов А. А., Иванин Н. С.

Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: antekromanov@gmail.com, nikivnik@gmail.com

Тематическое моделирование – способ статистического анализа текстов, предназначенный для выявления скрытых тем в коллекции документов. В данной работе рассматривается применение методов вероятностного тематического моделирования для обработки текстовых данных в системах отслеживания ошибок программного обеспечения. В качестве используемого алгоритма вероятностного тематического моделирования было отдано предпочтение методу латентного размещения Дирихле.

ВВЕДЕНИЕ

Одной из основных задач обработки естественного языка является автоматическое извлечение тем, которые содержатся в коллекциях текстовых документов. Примером таких документов являются описания дефектов в системах отслеживания ошибок программного обеспечения. Дефектами в системах данного типа называются не только ошибки и неполадки найденные в программном обеспечении, но также пожелания и вопросы пользователей. Знание тематик дефектов, характерных для конкретного программного обеспечения, важно для более глубокого понимания проблем проекта, учета пожеланий пользователей и повышения общего качества разработки и поддержки программного обеспечения. Однако проанализировать все дефекты проекта вручную трудно а иногда и не возможно. Выявление тем дефектов можно автоматизировать, используя методы вероятностного тематического моделирования. Наиболее популярным и зарекомендовавшим себя алгоритмом является метод латентного размещения Дирихле (latent Dirichlet allocation – LDA). В настоящей работе рассматривается задача построения тематической модели для коллекции дефектов с использованием алгоритма LDA.

I. СБОР И ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ДАННЫХ

Для анализа был собран корпус, состоящий из более 29 тыс. описаний дефектов проекта Apache Spark. Выбор был сделан на основе того, что система отслеживания ошибок данного проекта находится в открытом доступе и предоставляет REST API для извлечения данных. Качество исходных текстов существенно влияет на результаты тематического моделирования, поэтому описания дефектов были предобработаны, а именно:

- все символы приведены к нижнему регистру;
- удалены знаки препинания и табуляции;

- удалены стоп-слова английского языка;
- удалены тэги форматирования;
- удалены адреса электронной почты и гиперссылки;
- произведена лемматизация слов.

Современные алгоритмы машинного обучения работают с документами, которые представлены в виде векторов признаков. В качестве векторной модели используется «мешок слов». Текст в данной модели рассматривается как неупорядоченное множество слов. Каждому слову сопоставляется некий вес, отражающий его значимость. Вектор же формируется при упорядочивании всех уникальных слов в пространстве. Размерность вектора определяется числом уникальных слов во всей коллекции и является постоянной для всех документов [1]. Для оценки значимости слов была использована схема $tf - idf$ [2]:

$$tf - idf(w, d, D) = tf(w, d) \times \log \frac{|D|}{|d_w|},$$

где $tf(w, d)$ – отношение числа вхождений слова w к общему числу слов документа d ,

$|D|$ – число документов в коллекции D ,

$|d_w|$ – число документов из коллекции D , в которых встречается слово w .

II. ПОСТРОЕНИЕ ТЕМАТИЧЕСКОЙ МОДЕЛИ

Метод латентного размещения Дирихле предложен Дэвидом Блеем в 2003 году. Он основан на вероятностной модели [3]:

$$p(d, w) = \sum_{t \in T} p(d)p(w|t)p(t|d),$$

при дополнительных предположениях:

векторы документов $\theta_d = (p(t|d) : t \in T)$ порождаются одним и тем же вероятностным распределением из параметрического семейства распределений Дирихле $\text{Dir}(\theta, \alpha)$, $\alpha \in \{R^{|T|}$ на нормированных $|T|$ -мерных векторах;

векторы тем $\phi_t = (p(w|t) : w \in W)$ порождаются одним и тем же вероятностным распределением из параметрического семейства распре-

делений Дирихле $\text{Dir}(\theta, \beta)$, $\beta \in \mathbb{R}^{|W|}$ на нормированных векторах размерности $|W|$.

Инструментом реализации LDA выбрана библиотека `gensim` языка Python. Для построения модели коллекция документов была разделена на обучающую и контрольную выборки в отношении 4:1. Визуализация тематических моделей осуществлялась с помощью библиотеки `pyLDAvis`. Данная библиотека отображает тематическую модель в виде интерактивной пузырьковой диаграммы (см. рис. 1).

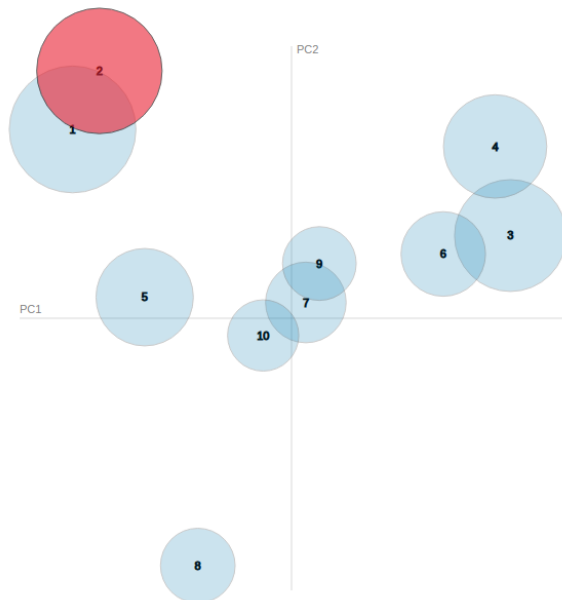


Рис. 1 – Визуализация полученной оптимальной тематической модели

Каждый пузырек на диаграмме представляет тему. Чем больше его размер, тем больше распространена данная тема. Хорошая тематическая модель должна иметь довольно большие непересекающиеся пузырьки, разбросанные по всей диаграмме. Также для каждой темы `pyLDAvis` показывает упорядоченный список наиболее характерных слов (см. табл. 1).

Таблица 1 – Примеры тем полученной оптимальной модели

№	Топ-5 релевантных слов темы
1	table, partition, select, hive, column
2	class, write, output, generate, implementation
3	raw, size, array, want, dataset
4	support, model, well, vector, documentation
5	spark, anonfun, run, apply, scala
6	value, key, type, function, string
7	info, executor, task, driver, run
8	file, none, set, default, build
9	code, test, follow, example, time
10	use, add, datum, create, need

III. ОЦЕНКА КАЧЕСТВА И ВЫБОР ОПТИМАЛЬНОЙ МОДЕЛИ

Общепринятой мерой качества вероятностной тематической модели является перплексия

контрольной выборки. Это мера несоответствия или «удивлённости» модели $p(w|d)$ термину w , наблюдаемым в документах d коллекции D :

$$P(D, p) = \exp\left(-\frac{1}{n} \sum_{d \in D} \sum_{w \in d} n_{dw} \ln p(w|d)\right)$$

Чем меньше эта величина, тем лучше модель p предсказывает появление терминов w в документах d коллекции D [4]. Перплексия измеряется по контрольной выборке документов, не используемых для построения модели. Это позволяет избежать занижения оценки в результате переобучения. Для поиска оптимальных параметров алгоритма LDA был использован подход, называемый `grid search`. Для этого алгоритм создания модели запускался с различными комбинациями значений параметров и в результате из множества построенных моделей, была выбрана модель с наименьшей перплексией. В результате подбора параметров было выявлена модель с оптимальным числом тем $T = 10$, с перплексией $P = -8.0917$.

ЗАКЛЮЧЕНИЕ

В рамках данной работы решена задача автоматического извлечения тем из текстовых описаний дефектов программного обеспечения. С помощью REST API осуществлен сбор дефектов из системы отслеживания ошибок, полученные данные преобразованы для построения тематической модели с использованием инструментов обработки естественного языка. Были подобраны оптимальные параметры для тематической модели, использованы средства визуализации.

Перспективой для улучшения тематической модели является использование не только описания дефектов, но и дополнительной информации, например, приложений и комментариев к дефектам. Актуальными направлениями для дальнейшего исследования могут быть определение изменения тем дефектов со временем и построение иерархического каталога тем.

Полученная тематическая модель может быть использована в качестве плагина, расширяющего функциональность системы отслеживания ошибок.

1. Manning C. D. Introduction to Information Retrieval / C. D. Manning, P. Raghavan, H. Schütze. – Cambridge: Cambridge University Press, 2008, – 496 p.
2. Lan M. Supervised and Traditional Term Weighting Methods for Automatic Text Categorization / M. Lan [et al.]. // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2009. – Vol. 31, P. 721–735.
3. Blei D. M. Latent Dirichlet allocation / D. M. Blei, A. Y. Ng, M. I. Jordan // Journal of Machine Learning Research – 2003. – Vol. 3, P. 993–1022.
4. Вероятностное тематическое моделирование [Электронный ресурс]. – Режим доступа: <http://www.machinelearning.ru/wiki/images/2/22/Voron-2013-ptm.pdf>. – Дата доступа: 08.10.2019.

АВТОМАТИЗАЦИЯ МАРКЕТИНГОВЫХ КАМПАНИЙ ПРИ ПОМОЩИ УНИВЕРСАЛЬНОЙ СИСТЕМЫ АВТОМАТИЗАЦИИ ЗАДАЧ

Сабадаш М. А., Петров А. В., Ярошевич Д. И.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: themaextra@gmail.com, gdiyaro@gmail.com, alexander_petrov@tut.by

В настоящее время корпоративной мировой тенденцией все чаще становятся процессы автоматизации, которые лежат в основе оптимального построения любого отдела организации. Применение автоматизированного подхода позволяет сократить издержки предприятия и значительно ускорить его процессы. В докладе рассмотрены проблемы автоматизации маркетинговых процессов и схема настройки универсальной системы автоматизации задач для их решения.

ВВЕДЕНИЕ

Область маркетинга все чаще занимает ключевую позицию в стратегическом и экономическом развитии, она является основополагающей в узнаваемости бренда, а получение обратной связи от клиентов и конечных пользователей может отразить настоящее положение продукта или услуги на рынке и позволит скорректировать план развития.

Для автоматизации процессов обработки информации часто используется специализированное программное обеспечение, однако значительно выгоднее использовать одну универсальную систему для автоматизации работ разных отделов, настроенную с учетом особенностей и требований каждого из них [1].

Как правило, пользователи систем автоматизации обычно сталкиваются с трудностями эксплуатации из-за высокого порога вхождения, переусложненного интерфейса и необходимости знания подробностей технической реализации выбранной системы. Основными задачами для эффективного построения и распространения систем автоматизации являются простой интерфейс, низкий порог вхождения, разделение зоны ответственности пользователей, обеспечение возможности работы с внешними источниками данных, возможность быстрого создания и отладки сценариев, а также возможность быстрого внесения изменений [2].

I. ЭТАПЫ МАРКЕТИНГОВЫХ ИССЛЕДОВАНИЙ

Для построения процесса любого маркетингового исследования необходимо определить ключевые задачи проекта. Задачи для маркетинговых исследований исходят из актуальных проблем бизнеса, стратегического плана предприятия, а также отдела управления рекламного развития. Для исключения ошибок валидации результатов и контроля за сроками выполнения маркетинговой кампании проводить этапы необходимо последовательно. Часть этапов мож-

но объединять, проводить параллельно, но не исключать.

Процесс исследований в области маркетинга состоит из следующих этапов [3]:

1. определение проблемы и целей исследования;
2. определение объектов исследования;
3. разработка плана исследования;
4. сбор данных;
5. анализ информации данных;
6. представление результатов.

Подробная схема процесса представлена на рис. 1.



Рис. 1 – Схема процесса маркетинговых исследований

Первый этап исследований содержит описание существующей бизнес-проблемы, перечень целей исследования и список возможных гипотез, которые необходимо подтвердить или опровергнуть. После определения целей и задач необходимо составить детальный проект маркетингового исследования. Данный проект представляет собой документ, в котором описаны следующие пункты:

- источники получения информации;
- метод исследования;
- инструменты исследований;
- выборка;
- способ взаимодействия с респондентами;
- сроки и бюджет работ.

Второй этап формулирует задачи исследования. Как правило, исследование содержит решение одной из четырех задач: разработка, описание, проверка гипотезы и предсказание. Ис-

следование задачи описания проводится, когда необходимо описать такие объекты, как рынок или его часть, определяя их характеристики на основе статистических данных. Если задача маркетинговых исследований является проверкой взаимосвязи между независимыми и зависимыми переменными, в таком случае маркетинговый отдел проводит исследования с целью выявления причин, вызвавших появление проблемы.

Третьим этапом является создание общего плана проведения маркетинговых исследований. Он определяет потребности в различных данных, их сбор, обработку и анализ. Эта стадия включает выбор определенных методов проведения исследований и разработку специфических задач. Также определяется потребность в информации, тип требуемой информации, источники и методы ее получения.

Четвертый этап позволяет собрать всю необходимую информацию для дальнейшего анализа. Информация представляет самую важную часть исследования, и она должна быть тщательно собрана с соблюдением всех алгоритмов и требований. Недостоверная, ошибочная, неполная информация может привести к неверным выводам и к неправильным решениям. В свою очередь, это приводит к негативным последствиям для развития всего бизнеса.

Пятым этапом является анализ полученных данных. Как правило, он требует преобразования исходных данных в необходимую форму (это может быть их форматирование, проверка на наличие ошибок, агрегация, представление в матричной форме). Это позволяет перевести массу необработанных данных в осмысленную и структурированную информацию.

Шестым этапом является представление результатов. Они могут быть оформлены в виде отчета, презентации и набора диаграмм. Результаты всегда представляются руководству предприятия [4].

Итогом маркетингового процесса являются результаты исследований, которые служат тем источником информации, который используется руководством для принятия правильных и обоснованных решений.

II. ОСОБЕННОСТИ НАСТРОЙКИ СИСТЕМЫ

С целью автоматизации маркетингового отдела была настроена созданная ранее универсальная система автоматизации задач. Опишем все этапы более подробно.

Согласно этапам маркетинговых исследований были проанализированы ключевые требования к системам автоматизации в области маркетинга, структура и особенности работы сценариев в данной области, необходимость поддержки внешних источников данных и сервисов, а также варианты переиспользования готовых сценариев.

Была выявлена необходимость унификации и упрощения визуального восприятия всех эта-

пов при создании и запуске сценариев, а также обоснована необходимость расширенной поддержки совместного использования системы с поддержкой ролевого доступа и контроля изменения общих данных.

Для взаимодействия между частями системы установлена самостоятельная очередь сообщений, через которую должны взаимодействовать все компоненты системы и обмениваться входящими данными. Между сервером со всеми экземплярами микросервисов системы и сервером базы данных настроен доступ, проверить который можно используя запуск тестового сценария.

Для формирования проблем и целей исследования используется сервис создания сценариев, заголовком и описанием которого являются кратко описанные цели маркетинговой кампании. Результаты второго этапа исследований заполняются в интерфейсе пользовательского сценария – в блоке аннотаций.

Для сбора данных настроены внешние источники данных: веб-сервисы, различные реляционные и нереляционные БД. При запуске маркетингового сценария запрашиваемые данные попадают в систему по заданному алгоритму.

Анализ полученных данных выполняется отдельным сервисом и позволяет редактору сценариев задать все правила фильтрации и кросс-валидации на подготовленных наборах данных.

Сервис анализа данных содержит именованные правила, которые могут быть переиспользованы в любой предстоящей маркетинговой кампании.

Для формирования отчетов используется самостоятельный микросервис с гибкой поддержкой построения графиков.

III. ЗАКЛЮЧЕНИЕ

Таким образом, настроенная система позволяет получить настоящую универсальность в сфере применения и может быть переиспользована в одной и той же организации для разных целей, что является отличным решением для организаций любого размера и повышает интерес со стороны потенциальных клиентов.

СПИСОК ЛИТЕРАТУРЫ

1. Котлер, Ф. Основы маркетинга. Краткий курс // Ф. Котлер, Л. Келлер // Вильямс. – 2019. – № 3. – С. 120–135.
2. Микроэкономика [Электронный ресурс] - Режим доступа: <http://be5.biz/ekonomika/m018/index.html>. – Дата доступа: 01.10.2019.
3. Идеальный план маркетингового исследования [Электронный ресурс] - Режим доступа: <http://powerbranding.ru/marketing-research/idealnyj-plan-marketingovogo-issledovaniya>. – Дата доступа: 01.10.2019.
4. Дробинцев, П. Автоматизация создания верифицированных тестовых сценариев на основе гидов // справочник. Санкт-Петербургский государственный политехнический университет, 2016. – 77с.

АЛГОРИТМЫ МАШИННОГО ОБУЧЕНИЯ В ПРИЛОЖЕНИЯХ ДЛЯ ИЗУЧЕНИЯ ИНОСТРАННЫХ ЯЗЫКОВ

Саскевич А. В., Стержанов М. В., Климович М. А.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: asaskevich@ya.ru, msterjanov@gmail.com

В данной работе определены возможности применения алгоритмов машинного обучения, таких как алгоритмы прогнозирования, распознавания образов или рекомендательные системы, в приложениях, специализирующихся на изучении иностранных языков. Кроме того, приведены непосредственные примеры их применения, а также представлены результаты, указывающие на то, что персонализация материала для обучения благоприятно сказывается на качестве обучения.

ВВЕДЕНИЕ

Активное развитие информационных технологий, а также расширение доступа к сети Интернет среди жителей всего мира создали необходимость разработки алгоритмов, позволяющих с высокой точностью настраивать параметры генерируемого контента непосредственно для каждого участника Сети. В первую очередь такая необходимость возникла в системах рекламных объявлений – правильный подбор объявлений для показа повышает продажи товара и число просмотров контента. Следом за рекламным контентом персонализация контента пришла в сферу новостей – пользователям, заинтересованным, например, в новостях политики, стали предлагаться соответствующие статьи. Развлекательный контент также стал подстраиваться под пользователя – появилось множество сервисов типа Netflix или Spotify, анализирующих то, как их пользователи слушают музыку и просматривают фильмы, предлагая аналогичные по жанру/стилю/содержимому. Естественно, персонализация не могла пройти мимо образовательных приложений. Появились такие системы как Coursera, Udemy, KhanAcademy, предлагающие пользователям онлайн-курсы на основе уже пройденных.

Аналогичные подходы нашли свое применение и в приложениях по изучению иностранных языков, например: Duolingo, LinguaLeo. Подача материала по изучению иностранного языка организована таким образом, чтобы пользователь мог регулярно повторять забытые слова, фразы и правила, в то же время изучая новые, подходящие ему по уровню сложности и текущих успехов в изучении. Для этого при разнообразных действиях формируется статистика, которая в дальнейшем применяется для работы алгоритмов машинного обучения.

I. АЛГОРИТМЫ РЕГРЕССИИ

Простейшей группой алгоритмов машинного обучения можно обозначить алгоритмы регрессии. Данная группа алгоритмов позволяет

проанализировать, например, количество ошибок пользователя и предсказать, сколько ошибок на тех же или аналогичных заданиях он будет совершать в будущем, или предсказывать, сколько времени займет выполнение аналогичного задания, позволяя оценивать прогресс в скорости, например, перевода текстов.

В приложении Duolingo поставили перед собой задачу спрогнозировать, через какое время пользователю необходимо повторить материал для лучшего его запоминания. Для решения задачи за основу взяли логистическую регрессию, изменив функцию потерь таким образом, чтобы можно было применить данные, собранные не только с конкретного пользователя, но и с множества других пользователей, вовлеченных в выполнение аналогичных заданий. Для этого была определена собственная функция потерь, основанная на понятии "кривой забывания" и представленная в общем виде формулой 1. Сам алгоритм получил название "регрессии полураспада" (half-life regression, HLR). Применение собственной функции потерь и алгоритма регрессии позволило повысить точность результата в поставленной задаче, а также уменьшить скорость забывания материала у пользователей [1].

II. АЛГОРИТМЫ РАСПОЗНАВАНИЯ ОБРАЗОВ

Другая группа алгоритмов, которая также нашла свое применение в изучении иностранных языков – это алгоритмы распознавания образов. В качестве примера можно предложить следующий режим работы – пользователь предоставляет системе некоторое графическое изображение – например, фотографию, сделанную камерой телефона, а затем система распознает объекты и предлагает слова для добавления в словарь и дальнейшего изучения.

В мобильном приложении-словаре Memrise для платформы iOS реализован соответствующий функционал, описанный в примере выше. Мобильная платформа iOS версий 12 и выше предоставляет набор инструментов CoreML для

разработки приложений с интеграцией алгоритмов машинного обучения. Среди них присутствует модуль для распознавания образов как на статическом изображении, так и в режиме реального времени с камеры устройств. За основу в данном модуле взята нейронная сеть YOLOv2, реализованная в оптимизированном виде для мобильных устройств на платформе iOS, и показавшая свою скорость, точность и качество работы во множестве реализованных систем [2–3].

III. АЛГОРИТМЫ РАСПОЗНОВАНИЯ ГОЛОСА

Для поддержания качества обучения пользователь должен не только выучить слова выбранного иностранного языка, но и поставить свою речь так, чтобы иностранец понял, что хочет сказать учащийся. Известно, что некоторые языки, например, китайский, являются тональными языками – смысл слову или фразе дает манера ее произношения. Стандартные алгоритмы распознавания голоса в данной ситуации могут оказаться бесполезными, так как в данной задаче необходима высокая точность и скорость распознавания [4].

Так, команда исследователей Google представила свой алгоритм потокового распознавания голоса на основе рекуррентных нейронных сетей (RNN). В своей работе они предложили свою модель рекуррентной нейронной сети, устроенной таким образом, чтобы обучение на разных языках не зависело от набора данных – в некоторых языках набор данных может достигать десятков миллионов примеров (например, наиболее популярные языки), а в некоторых диалектах число примеров может ограничиваться тысячами. Дисбаланс в формировании наборов данных в классических моделях приведет к увеличению точности вплоть до переобучения на насыщенных наборах и к высокому уровню ошибок на небольших наборах [5].

IV. РЕКОМЕНДАТЕЛЬНЫЕ СИСТЕМЫ

Рекомендательные системы могут быть применены практически без изменений поставленной задачи – подобрать контент для пользователя таким образом, чтобы он оказался адекватным по уровню сложности, интересным и актуальным. Для этого можно выделить такие критерии для построения модели как – число ошибок на выбранном задании, время его решения, число других пользователей, справившихся и несправившихся с заданием, общий рейтинг задания и аналогичные.

Среди готовых моделей, которые могут быть применены – алгоритмы коллаборативной фильтрации, модели на основе контента и мно-

жества критериев, гибридные модели и прочие. Кроме того, возможно создание системы под свои нужды, например, в приложении LinguaLeo используется собственная модель рекомендаций, позволяющая рекомендовать пользователю подходящие статьи и видео для обучения.

V. РЕЗУЛЬТАТЫ

Алгоритмы машинного обучения могут быть применены в широком спектре задач – начиная прогнозированием результатов пользователя, рекомендацией заданий и заканчивая распознаванием голоса и графического контента.

Персонализация контента позволила улучшить запоминание материала, а игрофикация повышает заинтересованность и вовлеченность пользователя.

В рамках мобильных приложений Busuu и Duolingo проведены академические исследования, показавшие следующие положительные результаты: более 84% пользователей улучшили свои письменные навыки, скорость обучения повысилась в 1.5 раза по сравнению с традиционными методами. Два месяца обучения через приложения позволили 75% пользователей улучшить свои навыки устной речи, 22 часа использования приложения оказались равноценны одному семестру изучения языка в университете, а учащиеся, никогда ранее не сталкивавшиеся с изучаемым языком, показали результаты, сравнимые с учащимися, имеющими некоторый опыт изучения языка [6-7].

СПИСОК ЛИТЕРАТУРЫ

1. Settles, B. A Trainable Spaced Repetition Model for Language Learning / B. Settles, B. Meeder // Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics – 2016. – Vol. 1 – P. 1848–1858.
2. CoreML – Machine Learning – Apple Developer Platform [Electronic resource] / – Mode of access: <https://developer.apple.com/machine-learning/core-ml/>. – Date of access: 08.10.2019.
3. Object detection – Memrise – User Guide [Electronic resource] / – Mode of access: <https://memrise.helpshift.com/a/memrise-learn-a-new-language/?s=getting-started&f=what-is-the-explore-mode>. – Date of access: 03.10.2019.
4. Helmer, C. Automatic speech recognition for second language learning: How and why it actually works / C. Helmer, S. Helmer // Speech Communication – 2013. – Vol. 1.
5. Kannan, A. Large-Scale Multilingual Speech Recognition with a Streaming End-to-End Model / A. Kannan, A. Datta // Interspeech.2019 – 2019. – Vol. 9 – P. 2130–2134.
6. Vesselinov, R. The busuu Efficacy Study / R. Vesselinov, J. Grego // Research Report – 2016.
7. Vesselinov, R. Duolingo Effectiveness Study / R. Vesselinov, J. Grego // Research Report – 2012.

$$\ell(\langle p, \Delta, \mathbf{x} \rangle; \Theta) = \left(p - 2^{-\frac{\Delta}{2^{\Theta \cdot \mathbf{x}}}} \right)^2 + \alpha \left(\frac{-\Delta}{\log_2(p)} - 2^{\Theta \cdot \mathbf{x}} \right)^2 + \lambda \|\Theta\|_2^2 \quad (1)$$

ПОСТРОЕНИЕ КЛАССИФИКАТОРА ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ С ИСПОЛЬЗОВАНИЕМ ПАРАМЕТРИЗОВАННЫХ ШАБЛОНОВ

Савёнок В. А., Медведев С. А., Селедец В. Н.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: savionak@gmail.com, vadik446s@gmail.com, msa@bsuir.by

В статье приводится постановка задачи классификации текстов на естественном языке с применением параметризованных шаблонов; описывается построение классификатора с использованием параметризованных шаблонов на примере классификации текстов, содержащих описание вакансий. Также приводится пример оценки точности построенного классификатора и подходы к ее повышению.

ВВЕДЕНИЕ

Задача классификации текстов на сегодняшний день широко распространена. В общем случае, для того, чтобы точно классифицировать множество текстов, необходимо проанализировать их содержимое на предмет наличия ключевых слов и фраз, которые определяют принадлежность данных текстов к заданным классам. Подход с использованием параметризованных шаблонов позволяет эффективно осуществить данный анализ и построить классификатор, обладающий достаточно высокой точностью.

I. ПОСТАНОВКА ЗАДАЧИ

При применении параметризованных шаблонов для решения задачи классификации множества текстов T на множество классов C , необходимо составить такое множество шаблонов P , чтобы множество возможных совпадений M_i , порождаемых шаблоном P_i , однозначно принадлежало множеству текстов, образующих класс C_i (см рис. 1).

Другими словами, каждому классу ставится в соответствие целевой шаблон. Наличие совпадения целевого шаблона в тексте говорит о принадлежности данного текста к классу, которому был поставлен в соответствие данный целевой шаблон. Такая формулировка решения позволяет определять принадлежность текста к нескольким классам [1].

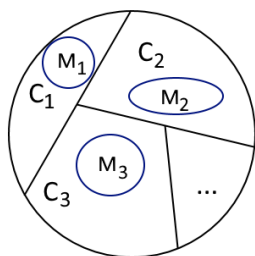


Рис. 1 – Принадлежность множеств совпадений к классам

II. ПОСТРОЕНИЕ КЛАССИФИКАТОРА ТЕКСТОВ НА ОСНОВЕ ПАРАМЕТРИЗОВАННЫХ ШАБЛОНОВ

Пусть необходимо построить классификатор текстов, содержащих описания вакансий. Обозначим заданное множество классов, состоящее из двух элементов: «Transportation» и «Others». К первому классу относятся тексты, содержащие описания вакансий водителя (грузового автомобиля, легкового автомобиля, такси, автобуса и т. д.), а ко второму классу – все остальные тексты (см. рис. 2 а). Результатом классификации должны стать два множества T_1 и T_2 , содержащие тексты, которые относятся к классам «Transportation» и «Others» соответственно. Общий вид результата классификации представлен на рисунке 2 б.

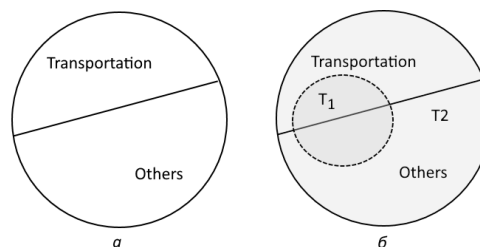


Рис. 2 – Классы для разделения текстов (а) и предварительный результат классификации (б)

Для решения поставленной задачи достаточно составить один параметризованный шаблон, такой, чтобы все совпадения, порождаемые им, принадлежали множеству текстов, относящихся к классу «Transportation». Тексты, в которых не будут найдены совпадения указанного шаблона, будут отнесены к классу «Others».

Множество возможных совпадений, которое должно порождаться целевым шаблоном, определяется путем анализа предметной области. Элементами данного множества выступают ключевые, специфические для данной предметной области, слова и фразы, которые употребляются в определенном контексте. Для текстов, содержащих описания вакансий, ключевыми словами являются названия профессий. Следовательно, они являются частью множества совпа-

дений, которые должен порождать целевой шаблон. Такой целевой шаблон позволит определить множество текстов T_1 , принадлежащих классу «Transportation». Все оставшиеся тексты составят множество текстов T_2 , относящихся к классу «Others».

III. ОЦЕНКА ТОЧНОСТИ КЛАССИФИКАТОРА

Для определения готовности классификатора к решению поставленной задачи необходимо оценить его точность. Способ определения точности классификатора зависит от типа входных данных, которые могут являться как обучающей выборкой, так и уже классифицированными реальными данными.

Принимая во внимание тот факт, что для обучающей выборки количество классов и количество текстов в каждом классе известны заранее, точность классификатора можно рассчитать по формуле

$$a = \frac{\sum_{i=1}^N c_i}{\sum_{i=1}^N n_i},$$

где N – количество заданных классов, c_i – число текстов, которые отнесены классификатором к классу i ; n_i – общее число текстов, принадлежащих классу i [2].

При создании классификатора на основании реальных данных, в которых тексты уже отнесены к одному или нескольким классам некоторым классификатором L_1 , заранее известны и количество классов, и количество текстов, принадлежащих данным классам. Однако, если классификатор L_1 не предоставляет требуемой точности, возникает необходимость в создании нового классификатора L_2 на основании полученных данных.

В силу того, что классификатор L_1 не является абсолютно точным, не все тексты, отнесенные к заданным классам, на самом деле принадлежат им. Для расчета точности создаваемого классификатора L_2 можно воспользоваться следующей формулой:

$$a = \frac{|Q| + |U_{L_2}|}{|F| + |U_{L_1}|},$$

где Q – множество текстов, для которых класс определен верно обоими классификаторами; U_{L_k} – множество текстов, класс которых определен верно классификатором L_k , при этом $U_{L_k} \cap Q = \emptyset$, $k = 1, 2$; F – множество текстов, обработанное классификатором L_2 [2].

IV. ПОВЫШЕНИЕ ТОЧНОСТИ КЛАССИФИКАТОРА

Как видно из рисунка 2 б, полученное после предварительной классификации множество T_1 не является полным, так как не все тексты класса «Transportation» вошли в T_1 . Кроме того, T_1 содержит тексты, принадлежащие классу «Others», что требует дальнейшего уточнения целевого шаблона. Для повышения точности

классификатора необходимо решить две задачи (см. рис. 3):

1. исключить из T_1 тексты, принадлежащие классу «Others»;
2. включить в T_1 оставшиеся тексты, принадлежащие классу «Transportation».

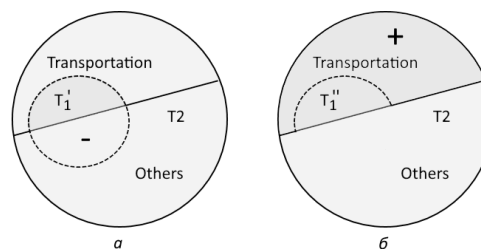


Рис. 3 – Уточнение результатов классификации путем исключения (а) и включения (б) текстов, принадлежащих соответствующим классам

Для исключения из T_1 текстов, принадлежащих классу «Others», необходимо проанализировать эти тексты и добавить в целевой шаблон конструкцию исключения, которая содержит ключевые слова и фразы, не относящиеся к «Transportation». Также необходимо уточнить контекст употребления уже заданных ключевых выражений. Для этого можно использовать параметризованный шаблон, совпадения которого будут состоять из заданного ключевого слова или фразы с захватом окружающих слов. Полученный шаблон необходимо применить к текстам, которые ошибочно попали во множество T_1 , после чего найденные совпадения следует добавить в качестве исключений в целевой шаблон.

Для расширения множества T_1 необходимо проанализировать тексты, ошибочно не отнесенные к классу «Transportation», и дополнить ключевые слова и фразы новыми элементами.

Значительное повышение качества классификатора достигается путем выполнения описанных операций в несколько итераций.

ЗАКЛЮЧЕНИЕ

Описанная методика построения классификатора текстов с использованием параметризованных шаблонов была успешно применена на практике для выявления текстов на английском языке, содержащих описание вакансии водителя. Достигнутая точность классификатора на выборке текстов из открытых источников [3] по полученным оценкам составляет более 91%.

1. Загоруйко, Н. Г. Прикладные методы анализа данных и знаний / Загоруйко, Н. Г // Издательство: Новосибирск: ИМ СО РАН, 1999. – 270 с.
2. Оцека классификатора (точность, полнота, F-мера) [Электронный ресурс] – Режим доступа: <http://bazhenov.me/blog/2012/07/21/classification-performance-evaluation.html>. – Дата доступа: 08.09.2019.
3. Driver Jobs, Employment | Indeed.com [Электронный ресурс] – Режим доступа: <https://www.indeed.com/q-driver-jobs.html>. – Дата доступа: 15.09.2019.

АДАПТАЦИЯ УСТРОЙСТВА ПОСЛЕДОВАТЕЛЬНОГО РАДИОЛОКАЦИОННОГО РАСПОЗНАВАНИЯ К ПРОСТРАНСТВЕННОЙ ОРИЕНТАЦИИ ЦЕЛИ

Свинарский М. В., Ярмолик С. Н., Храменков А. С., Зайко Е. В.

Кафедра автоматики, радиолокации и приемо-передающих устройств, Военная академия Республики Беларусь

Минск, Республика Беларусь

E-mail: mechislav1993@gmail.com, yarmsergei@yandex.ru, xras.tech@mail.ru, zayko.eugene@mail.ru

В докладе рассмотрена структура устройства последовательного радиолокационного распознавания по сигнальным и траекторным признакам с учетом адаптации к пространственной ориентации наблюдаемого объекта. Адаптация эталонных классификационных признаков к условиям наблюдения объекта достигается за счет учета плотностей вероятности оценок неизвестных параметров изменяющихся условий наблюдения. Совместное использование сигнальных и траекторных признаков повышает качество принимаемых решений о классе объекта наблюдения.

ВВЕДЕНИЕ

Задача радиолокационного распознавания заключается в установлении факта принадлежности наблюдаемого объекта к определенному классу. Информация о классе объекта радиолокационного наблюдения может использоваться для целераспределения, целеуказания, адаптации средств поражения, вскрытия замысла противника и т.п. При этом задача радиолокационного распознавания должна решаться в реальном масштабе времени и обеспечивать высокую достоверность принимаемых решений. В связи с этим повышение эффективности алгоритмов распознавания путем внедрения процедур адаптации с учетом совместного использования сигнальных и траекторных признаков является весьма актуальной задачей.

I. ОСНОВНАЯ ЧАСТЬ

При использовании сигнальной и траекторной информации о классе наблюдаемого объекта целесообразно применять байесовское решающее правило по совокупности признаков в мультипликативной форме [1]:

$$\text{если } \sum_{g=1}^M P(A_g) \Lambda(\xi_{k+1}|A_g) L_{\eta}(\xi_{k+1}|A_g) \Pi_{jg}(k+1) < \sum_{g=1}^M P(A_g) \Lambda(\xi_{k+1}|A_g) L_{\eta}(\xi_{k+1}|A_g) \Pi_{lg}(k+1),$$

$$j, l = 1 \dots M, j \neq l, \text{ то } A_l^*$$

где M - число распознаваемых классов объектов; $P(A_g)$ - априорная вероятность наблюдения объекта g -го класса; $\Lambda(\xi_{k+1}|A_g)$ - отношение правдоподобия, сформированное по сигнальным признакам, ξ_{k+1} ; $L_{\eta}(\xi_{k+1}|A_g)$ - коэффициент правдоподобия, сформированный по траекторным признакам; $\Pi_{jg}(k+1)$ - стоимость решения о g -м классе при наличии j -го на $(k+1)$ -м шаге наблюдения.

Отношение правдоподобия $\Lambda(\xi_{k+1}|A_g)$ характеризует степень достоверности гипотезы о соответствии наблюдаемых сигнальных признаков ξ_{k+1} эталонным значениям объекта g -го класса. Следует отметить, что сигнальные классификационные признаки существенно зависят от ориентации летательного аппарата (ЛА) в пространстве [2]. Пространственную ориентацию цели в системе координат линии визирования радиолокатора принято характеризовать совокупностью трех углов: курса, тангажа и крена ($\Theta_{k+1}^{\text{ЛВ}} = \|\psi^{\text{ЛВ}} \phi^{\text{ЛВ}} \gamma^{\text{ЛВ}}\|$) [2]. В процессе полета углы пространственной ориентации (УПО) ЛА относительно радиолокатора постоянно изменяются, что обуславливает необходимость адаптации эталонных сигнальных признаков к условиям наблюдения.

Перспективным направлением в создании адаптивных систем распознавания является использование плотности вероятности (ПВ) оценок УПО наблюдаемого объекта. Суть данного метода заключается в адаптивном формировании эталонных сигнальных признаков, в соответствии с текущими оценками совместного распределения УПО ЛА [2]:

$$\Lambda(\xi_{k+1}|A_g) = \int_{\Gamma_{\Theta_{k+1}^{\text{ЛВ}}}} p(\hat{\Theta}_{k+1}^{\text{ЛВ}}|A_g) \Lambda(\xi_{k+1}|\hat{\Theta}_{k+1}^{\text{ЛВ}}, A_g) \times d(\hat{\Theta}_{k+1}^{\text{ЛВ}}), \quad g = 1 \dots M,$$

где $p(\hat{\Theta}_{k+1}^{\text{ЛВ}}|A_g)$ - ПВ УПО ЛА в системе координат линии визирования, условная по классу наблюдаемого объекта; $\Lambda(\xi_{k+1}|\hat{\Theta}_{k+1}^{\text{ЛВ}}, A_g)$ - отношение правдоподобия, условное по классу наблюдаемого объекта и значению УПО ЛА; $\Gamma_{\Theta_{k+1}^{\text{ЛВ}}}$ - область определения возможных значений УПО ЛА.

Коэффициент правдоподобия $L_{\eta}(\xi_{k+1}|A_g)$ характеризует достоверность гипотезы о соответствии наблюдаемых траекторных признаков

η_{k+1} априорным диапазоном распределения признаков объекта g -го класса. Расчет коэффициента правдоподобия осуществляется в соответствии с выражением [1]:

$$L_{\eta}(\xi_{k+1}|A_g) = \int_{V_{\eta}^{(g)}} p(\eta_{k+1}|A_g)p(\eta_{k+1}|\xi_{k+1}, A_g) \times \\ \times d\eta_{k+1}, g = 1 \dots M,$$

где $p(\eta_{k+1}|A_g)$ - априорная ПВ траекторных признаков η_{k+1} ; $p(\eta_{k+1}|\xi_{k+1}, A_g)$ - апостериорная ПВ траекторных признаков; $V_{\eta}^{(g)}$ - область определения возможных значений траекторных признаков η_{k+1} объекта g -го класса.

II. РЕЗУЛЬТАТЫ И ВЫВОДЫ

Структурная схема устройства радиолокационного распознавания, учитывающая сигнальные признаки адаптивные к пространственной

ориентации объекта и траекторные признаки, представлена на рисунке (смотри рисунок 1).

Таким образом, синтезирована структурная схема устройства радиолокационного распознавания наблюдаемого класса объекта, позволяющая с требуемой эффективностью решать поставленную задачу.

СПИСОК ЛИТЕРАТУРЫ

1. Горшков, С. А. Сочетание сигнальных и траекторных признаков в задачах измерения-распознавания / С. А. Горшков, А. С. Солонар // Прикладная радиоэлектроника, 2009. – Т. 8. – № 4. – С. 497–500.
2. Ярмолик, С. Н. Учет распределения углов пространственной ориентации летательного аппарата при адаптации радиолокационных портретов к изменяющимся условиям наблюдения / С. Н. Ярмолик, М. В. Свинарский, А. С. Храменков, Е. В. Зайко // Доклады БГУИР, 2018. – № 5. – С. 28–36.

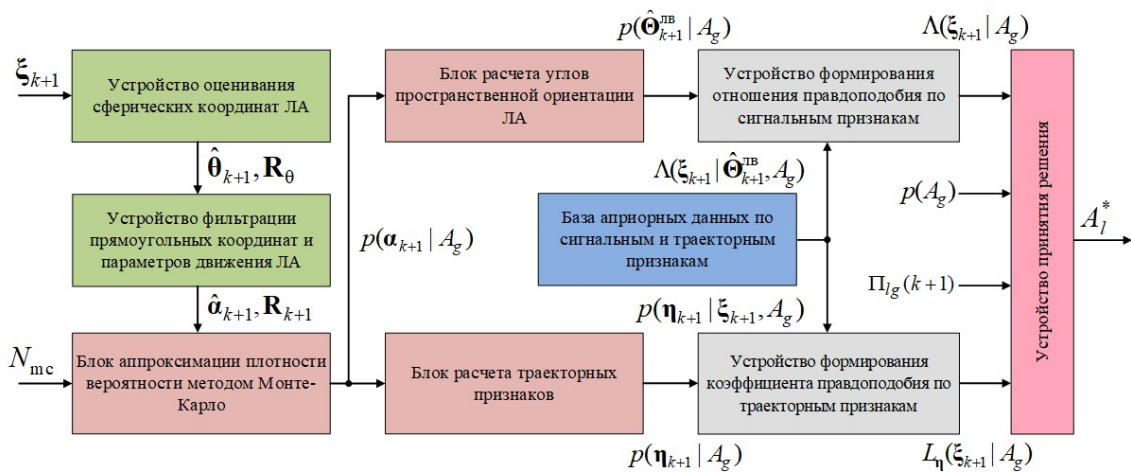


Рис. 1 – Структурная схема устройства радиолокационного распознавания по сигнальным и траекторным признакам с адаптацией к пространственной ориентации объекта

СТАТИСТИЧЕСКОЕ ПОСТРОЕНИЕ ДАТЧИКОВ БАЗОВОЙ СЛУЧАЙНОЙ ВЕЛИЧИНЫ НА ОСНОВЕ ФЛУКТУАЦИЙ ФИЗИЧЕСКИХ СИСТЕМ

Сосинович Е. С., Жук Е. Е.

Кафедра математического моделирования и анализа данных, Белорусский государственный университет
Минск, Республика Беларусь
E-mail: fpm.sosinovi@bsu.by, zhukee@mail.ru

В данной работе излагаются особенности физических датчиков базовых случайных величин и методы их построения. Предложен метод построения датчиков на основе реализации любой случайной величины путем функционального преобразования. Исследована эффективность предложенного метода, а также приведен пример реализации функционального преобразования на реальных данных.

ВВЕДЕНИЕ

Проблема имитационного статистического моделирования случайной величины ξ с заданной функцией распределения вероятностей $F_\xi(x)$ сводится к проблеме моделирования базовой случайной величины (БСВ) α , равномерно распределенной на отрезке $[0, 1]$. Датчик БСВ – устройство, позволяющее по запросу получать реализацию случайной величины (СВ) или несколько независимых реализаций. Выделяют несколько типов датчиков: табличный, физический и программный. Физический датчик БСВ – специальное радиоэлектронное устройство, являющееся приставкой к ЭВМ. Он состоит из источника флуктуационного шума, значение которого является некоторой СВ [1]. Основным преимуществом физического датчика БСВ является его криптографическая стойкость за счет невозможности повторения некоторой ранее полученной реализации. Для построения датчика БСВ, обладающего улучшенными вероятностными свойствами, предлагается подвергать выборку-реализацию произвольной СВ ξ функциональному преобразованию на основе эмпирической функции распределения (ЭФР). Эффект от такого преобразования исследуется при помощи стандартного набора тестов Д. Кнута: критерий согласия с функцией распределения (критерий Колмогорова и Пирсона), тестов «совпадение моментов» и «ковариация» [1].

I. ФУНКЦИОНАЛЬНОЕ ПРЕОБРАЗОВАНИЕ

Приведем функциональное преобразование, позволяющее построить датчик БСВ на основе реализации произвольной СВ.

Базовая случайная величина α , равномерно распределенная на $[0, 1]$, имеет функцию распределения:

$$F_\alpha(x) = \begin{cases} 0, & x \leq 0, \\ x, & 0 < x < 1, \\ 1, & x \geq 1, \end{cases}$$

математическое ожидание $E\{\alpha\} = \frac{1}{2}$ и дисперсию $D\{\alpha\} = \frac{1}{12}$.

Установлен следующий факт [2,3]. Если существует обратная $F_\xi^{-1}(\cdot)$ функции распределе-

ния $F_\xi(\cdot)$ некоторой случайной величины ξ , то БСВ α можно смоделировать следующим образом: $\alpha = F_\xi(\xi)$, так как

$$\begin{aligned} F_\alpha(x) &= P(\alpha \leq x) = P(F_\xi(\xi) \leq x) = \\ &= P(\xi \leq F_\xi^{-1}(x)) = F_\xi(F_\xi^{-1}(x)) = x. \end{aligned}$$

Пусть имеется выборка-реализация $\Xi = \{\xi_1, \dots, \xi_n\}$ СВ ξ . По ней строим ЭФР:

$$\begin{aligned} \hat{F}_\xi(x) &= \frac{1}{n} \sum_{t=1}^n I(x - \xi_t), \quad x \in \mathbf{R}, \\ I(x) &= \begin{cases} 1, & x \geq 0, \\ 0, & x < 0, \end{cases} \end{aligned} \quad (1)$$

в качестве статистической оценки неизвестной функции распределения $F_\xi(\cdot)$.

Таким образом, посредством функционального преобразования элементов исходной последовательности Ξ на основе ЭФР (1) получаем реализацию $X = (x_1, \dots, x_n)$ БСВ, где $x_i = \hat{F}_\xi(\xi_i)$, $i = \overline{1, n}$.

II. ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ

Проверим эффективность преобразования на основе ЭФР (1) с помощью набора тестов Д. Кнута. В тестах Колмагорова и Пирсона относительно последовательности X проверяется гипотеза согласия с равномерным распределением вероятностей $H_0: F_x(\cdot) \equiv F_\alpha(\cdot)$ против альтернативы $H_1 = \bar{H}_0: \exists x \in \mathbf{R}: F_x(x) \neq F_\alpha(x)$. Тесты строятся при малом уровне значимости $\epsilon \in (0, 1)$, представляются через P -значение $P(X, n) \in [0, 1]$ и имеют вид:

$$\begin{cases} \text{принимается } H_0, P(X, n) \geq \epsilon, \\ \text{принимается } H_1, P(X, n) < \epsilon. \end{cases}$$

В [2] было установлено, что для критериев Колмагорова, Пирсона и «совпадения моментов» [1] P -значения будут иметь вид соответственно:

- $P(X, n) = 1 - K(1/\sqrt{n})$ для критерия Колмагорова, где $K(z) = 1 - 2 \sum_{j=1}^{\infty} (-1)^{j-1} e^{-2j^2 z^2}$, $z \geq 0$ – функция распределения Колмагорова.
- $P(X, n) = 1 - F_{\chi_{k-1}^2}(\chi^2)$ для χ^2 -критерия Пирсона, где $k \geq 2$ – число ячеек одинаковой

длины, на которые разбивается отрезок $[0, 1]$, $F_{\chi^2_{k-1}}(\cdot)$ – функция χ^2 -распределения с $k - 1$ степенями свободы, $\chi^2 = \sum_{i=1}^k \frac{(n_i - \frac{n}{k})}{\frac{n}{k}} - \chi^2$ -статистика, n_i – число элементов преобразованной последовательности, попавших в i -тую ячейку.

- $P(X, n) = 2(1 - \Phi(\sqrt{12n}|\bar{x} - \frac{1}{2}|))$,
 $P(X, n) = 2(1 - \Phi(6\sqrt{2(n-1)}|s_x^2 - \frac{1}{12}|))$
 для гипотез относительно математического ожидания μ_x и дисперсии σ_x^2 соответственно для теста «совпадение моментов», где $\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-\frac{\omega^2}{2}} d\omega$ – функция распределения стандартного нормального закона, $\bar{x} = \frac{1}{n} \sum_{t=1}^n x_t$ – оценка для μ_x , $s_x^2 = \frac{1}{n-1} \sum_{t=1}^n (x_t - \bar{x})^2$ – оценка для σ_x^2 .

Установлено [2], что приведенные выше P -значения: $P(X, n) \rightarrow 1, n \rightarrow \infty$. Следовательно, всегда можно добиться принятия гипотезы H_0 .

Для теста «ковариация» справедлив следующий факт [3]:

$$E\{x_t\} = \frac{1}{2} + \frac{1}{2n}, D\{x_t\} = Cov\{x_t, x_t\} = \frac{1}{12} - \frac{1}{12n^2}, Cov\{x_t, x_k\} = \frac{1}{12n} - \frac{1}{12n^2},$$

что соответствует БСВ при $n \rightarrow \infty$.

Можно предположить, что зависимость между элементами преобразованной последовательности можно «ослабить», если преобразованию на основе ЭФР $\hat{F}_\xi(\cdot)$, построенной по последовательности $\Xi = \{\xi_1, \dots, \xi_n\}$, подвергать так называемые «вновь поступающие» значения $\Xi_n^m = \{\xi_{n+1}, \dots, \xi_{n+m}\}$. Как было установлено в [3], относительно такого преобразования уже нельзя утверждать, что тесты Колмогорова, Пирсона и «совпадение моментов» будут «заведомо проходить». Следовательно, преимущество преобразования «вновь поступающих» значений не оправдалось.

III. ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Применим предложенное преобразование к реальным данным для построения датчика БСВ по реализации нормально распределенной СВ.

Для проверки свойств преобразования используются ежедневные значения атмосферного давления на уровне метеостанции в Минске за

01.05.2015 – 22.04.2019. Установлено, что данная последовательность $\Xi = \{\xi_1, \dots, \xi_n\}$ подчиняется нормальному закону.

Построим две последовательности БСВ: $X^{(1)}$ и $X^{(2)}$. $X^{(1)}$ получим путем применения преобразования на основе ЭФР (1), построенной по Ξ , ко всей выборке Ξ . Для построения $X^{(2)}$ разобьем исходную последовательность на две части: $\Xi = (\Xi_0^{\lfloor \frac{n}{2} \rfloor}, \Xi_{\lfloor \frac{n}{2} \rfloor}^n)$, по выборке $\Xi_0^{\lfloor \frac{n}{2} \rfloor}$ построим ЭФР (1) и применим ее для преобразования выборки $\Xi_{\lfloor \frac{n}{2} \rfloor}^n$.

Результаты проверки эффективности предложенных преобразований с помощью набора тестов Д. Кнута представлены в Таблице 1.

ЗАКЛЮЧЕНИЕ

Результаты тестирования позволяют сделать вывод, что построенная с помощью предложенного преобразования последовательность действительно может служить базовой случайной величиной. Таким образом, данное функциональное преобразование позволяет моделировать датчик БСВ на основе произвольной последовательности СВ. Если входная последовательность пополняется «вновь поступающими» значениями, то их целесообразно присоединять к предыдущим и подвергать все доступные на данный момент значения преобразованию.

Преимуществом такого построения датчика БСВ является отсутствие в нем детерминированного алгоритма. Это позволяет утверждать, что такой датчик защищен от взлома, так как СВ, используемая для построения датчика, не является псевдослучайной.

СПИСОК ЛИТЕРАТУРЫ

1. Харин, Ю. С. Практикум на ЭВМ по математической статистике: Для мат. спец. ун-тов / Ю. С. Харин, М. Д. Степанова. – Мн. : изд-во «Университетское», 1987. – 304 с.
2. Жук Е. Е. Улучшение статистических свойств датчиков БСВ на основе эмпирической функции распределения / Е. Е. Жук // Весці НАН Беларусі. Сер. фіз.-мат. навук. – 2008. – №3. – С. 4–10
3. Жук Е. Е. Исследование ковариационных свойств псевдослучайных последовательностей, преобразованных на основе эмпирической функции распределения // Весці НАН Беларусі. Сер. фіз.-мат. навук. – 2010. – №2. – С. 18–22

Таблица 1 – Результаты экспериментального подтверждения эффективности

Выборка	Объем, n	Кр. Пирсона, $p - value$	Кр. Колмогорова, $p - value$	Величина $ \bar{x} - \frac{1}{2} $	Величина $ s_x^2 - \frac{1}{12} $
$X^{(1)}$	1452	1	0.9217	0.002869605	2.430142e-05
$X^{(2)}$	726	1	0.4057	0.004454766	5.063509e-05

КЛАССИФИКАЦИЯ ВРЕДОНОСНОГО ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПО СПОСОБУ ВНЕДРЕНИЯ

Царегородцев Д. В.

Кафедра интеллектуальных информационных технологий, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: tsaregorotsev.d@gmail.com

В представленной работе рассматривается классификация вредоносного программного обеспечения по способу внедрения с помощью древовидного классификатора, описаны способы улучшения качества классификации и способы предотвращения переобучения классификатора

ВВЕДЕНИЕ

Ежедневно в мире создаётся большое количество новых либо модифицированных экземпляров вредоносных программ, каждую из которых необходимо исследовать для того, чтобы узнать, какие вредоносные действия она совершает, разработать защиту от неё и восстановить систему после заражения, если это возможно.

Очевидно, что подобный анализ каждого вредоносного файла вряд ли является возможным. Таким образом, возникает такая задача, как автоматическая классификация вредоносного программного обеспечения.

В данной работе будет рассмотрена классификация вредоносных файлов по способу внедрения. По способу внедрения мы будем разделять файлы на пять классов: `failed_analysis`, `clear`, `trojan`, `worm`, `downloader`.

I. ПОЛУЧЕНИЕ ДАННЫХ

Получение данных для данной работы осуществлялось с помощью системы автоматического анализа вредоносных файлов Cuckoo Sandbox[1]. Преимуществом данной системы являются генерация отчётов в формате json, который крайне удобен для дальнейшего использования и наличие легко расширяемого набора сигнатур.

Для разработки классификатора было проанализировано порядка 4000 исполняемых файлов с помощью системы автоматического анализа Cuckoo Sandbox.

II. КОНСТРУИРОВАНИЕ ПРИЗНАКОВ

Изначально полученные данные для классификации представлены в виде времени работы виртуальной среды, в которой исполнялся файл, которое выглядит как строка вида «ЧЧ:ММ:СС» и бинарных сигнатур, которые сигнализируют об определённом поведении анализируемого исполняемого файла.

В таком виде данные не подходят для построения модели, поэтому они должны быть преобразованы в подходящий вид. Для начала время было переведено из строковой переменной в

числовую, путём перевода в количество секунд, затраченных на анализ.

Существует некоторый набор сигнатур, которые сигнализируют об одном и том же поведении файла, но делают это разными способами (таким образом увеличивается точность результата, так как при не срабатывании одной сигнатуры по какой-либо причине, её будет дублировать другая). Использовать все такие сигнатуры для классификации будет избыточно, более того, они могут создавать ненужный шум, так что было решено объединять подобные наборы сигнатур в один признак, значение которого будет принимать положительное значение при срабатывании хотя бы одной сигнатуры из его набора.

Были выделены следующие наборы сигнатур:

1. сигнализирующие о скачивании файла из сети;
2. сигнализирующие о распаковке других файлов из анализируемого.

Так же из обучающей выборки были удалены все сигнатуры, энтропия которых равна нулю, то есть все, которые не сработали ни на одном файле и все, которые сработали сразу на всех.

III. ВЫБОР АЛГОРИТМА

Для классификации был выбран алгоритм машинного обучения «дерево принятия решений». В данном случае дерево будет являться деревом для классификации, так как предсказываемый результат является классом, к которому принадлежат данные [2].

Дерево классификации было построено по алгоритму ID3[3]:

1. Взять все неиспользованные признаки и посчитать их энтропию относительно тестовых образцов;
2. выбрать признак, для которого энтропия минимальна (а информационная выгода соответственно максимальна);
3. сделать узел дерева, содержащий этот признак.

Выбор атрибута был проведён на основании прироста информации (IG), который был расчи-

тан по формуле (1)[4].

$$IG(Q) = S_0 - \sum_{i=1}^q N_i / NS_i, (1)$$

где Q – признак, S_0 – энтропия до деления, N – количество экземпляров до деления, S_i – энтропия i -той части после деления, N_i – количество экземпляров в i -той группе после деления, а q – количество возможных значений переменной Q .

Сама энтропия была рассчитана по формуле (2).

$$S = - \sum_{i=1}^N p_i \log_2 p_i, (2)$$

где N – количество возможных классов, p_i – вероятность нахождения экземпляра в классе i .

Дерево классификации останавливает своё деление при достижении глубины равной 3. При такой глубине дерево даёт верный ответ для 70,3% экземпляров, что является довольно низким результатом.

IV. КРОСС-ВАЛИДАЦИЯ

Для того, чтобы снизить вероятность переобучения дерева на обучающей выборке, для оценки качества был использован метод перекрёстной проверки[5].

Вся обучающая выборка была разделена на пять частей, после чего дерево тренировалось на 4 частях, а качество классификации проверялось на пятой. Данный процесс повторялся пять раз, после чего общим результатом качества классификации принималось среднее значение качества для всех результатов.

V. НАХОЖДЕНИЕ ОПТИМАЛЬНЫХ ПАРАМЕТРОВ

Для нахождения оптимального параметра глубины дерева дерево было обучено много раз с глубиной от 2 до 10, после чего было рассчитано качество классификации для каждого варианта (Рис. 1).

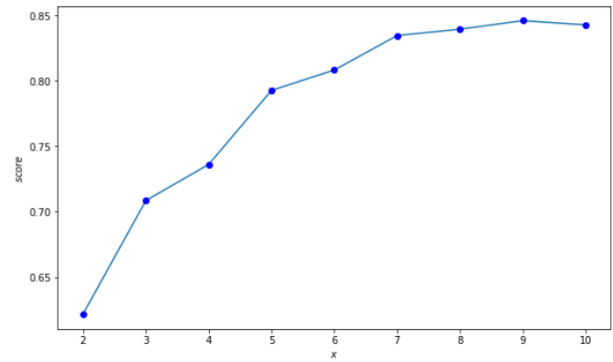


Рис. 1 – Зависимость качества классификации от глубины дерева

Таким образом, оптимальной глубиной для дерева является 9, которая даёт качество классификации 84,5%.

Далее была построена матрица ошибок дерева классификации, которая представлена на таблице 1.

Таблица 1 – Матрица ошибок дерева классификации

	clear	trojan	failed	d-loader	worm
clear	217	0	212	0	0
trojan	0	83	0	0	1
failed	16	0	364	0	0
d-loader	1	0	1	415	0
worm	0	0	0	0	10

При анализе ошибок дерева классификации можно заметить, что абсолютное большинство ошибок происходит между двумя классами – невредоносным программным обеспечением и программным обеспечением, которое не было проанализировано корректно. Улучшение качества классификации между этими двумя классами будет проработано в будущем, так как для поставленной задачи ошибки между этими классами не критичны, потому что классы вредоносного программного обеспечения были классифицированы с высокой степенью точности, что и являлось основной целью данного классификатора.

СПИСОК ЛИТЕРАТУРЫ

1. Cuckoo Sandbox – automated malware analysis [Электронный ресурс]. – Режим доступа: <https://cuckoosandbox.org/>. – Дата доступа: 09.10.2019.
2. Quinlan, J. R. C4.5: Programs for Machine Learning / J. R. Quinlan // San Francisco, CA, USA. Morgan Kaufmann Publishers, Inc. – 1993.
3. Паклин, Н. Б. Бизнес-аналитика: от данных к знаниям(+CD): Учебное пособие. 2-е изд. / Н. Б. Паклин, В. И. Орешков // СПб: Питер, 2013.
4. Mitchell, T. C4.5: Machine Learning / T. Mitchell // McGraw-Hill Science/Engineering/Math. – 1997.
5. Лопез, П. М. Машинное обучение: алгоритмы для бизнеса. / П. М. Лопез // СПб: Питер, 2019.

РАСШИРЕНИЕ ВОЗМОЖНОСТЕЙ ЯЗЫКА РАЗМЕТКИ AIML ДЛЯ СОЗДАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ЧАТ-БОТОВ

Урбанович А. И., Кадан А. М.

Факультет математики и информатики, Кафедра системного программирования и компьютерной безопасности

Гродненский государственный университет имени Янки Купалы

Гродно, Республика Беларусь

E-mail: urbanovichwork@gmail.com, kadan@mf.grsu.by

Для задач создания интеллектуальных чат-ботов предлагаются средства расширения возможностей языка разметки AIML путем определения новых операторов и тегов. Даны практические рекомендации по использованию предложенных расширений. Приведены примеры их использования.

ВВЕДЕНИЕ

Создание чат-ботов – это тренд, в котором на данный момент еще нет разработки, которая смогла бы решить все поставленные в данной теме задачи. В последнее время многие компании разрабатывают чат-ботов, но в основном они служат в качестве помощников для обращения с продукцией компании. В то же время, ежегодно, множество людей создающих чат-ботов соревнуются между собой для получения премии Лёбнера, которая ежегодно является платформой для практического проведения теста Тьюринга, в котором люди, выступающие в роли судей, должны сказать общаются они с чат-ботом или человеком.

В основном для создания чат-ботов применяется два метода – это языки разметки [1] и нейросети, главное отличие которых в том, что языками разметки пользуются люди для того, чтобы вручную составить пару вопрос-ответ, по которому их чат-боты смогут отвечать, а в нейросети программа обучается на основе множества данных для построения взаимосвязей и делания выбора в сторону наиболее подходящего под конкретную ситуацию ответа [2]. В своей разработке мы пытаемся объединить оба варианта.

I. ОСНОВНЫЕ БАЗОВЫЕ ВОЗМОЖНОСТИ ЯЗЫКА AIML

AIML (Artificial Intelligence Markup Language) – язык разметки для искусственного интеллекта. Файл AIML – содержит набор категорий (category). Категория включает паттерны (pattern) и ответные реплики (template). Полноценный чат бот создается множеством категорий, в которых обозначено на что и как именно виртуальный собеседник будет отвечать. Принцип реализации можно продемонстрировать простым примером, где на фразу пользователя «Привет», чат-бот также ответит нам «Привет».

```
<category>
```

```
<pattern>ПРИВЕТ</pattern>
```

```
<template>Привет</template>
</category>
```

У паттернов есть возможность использовать различные операторы, при помощи которых, в рамках одной категории, можно увеличить покрытие понимаемых чат-ботом реплик. Например, оператор «*» будет расцениваться как одно или несколько любых слов, а оператор «()» служит для возможности указать несколько возможных слов, например (ДРУЗЬЯ КОЛЛЕГИ). В результате для паттерна «* ПРИВЕТ * (ДРУГ КОЛЛЕГА)» чат-бот сможет распознать и ответить на фразу «Пламенный привет, дорогой друг».

При этом ответную реплику также можно улучшить при помощи специальных AIML тегов. При использовании тега <random> и перечисления внутри него, через тег , нескольких вариантов ответа, чат-бот получит возможность случайным образом выбирать один из таких ответов, что может быть использовано для избавления от монотонных и однотипных ответов на одни и те же вопросы.

II. ОСОБЕННОСТИ СОЗДАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ ЧАТ-БОТОВ НА ОСНОВЕ AIML

Процесс обработки категорий и выдачи наиболее подходящего ответа выглядит следующим образом: изначально создается массив, индексами которого служат первые слова из всех паттернов, а элементами – новые массивы с индексами последующих слов. Таким образом выстраивается древовидная иерархия, где каждой последовательности слов соответствует цепочка из массивов, ведущая к наиболее подходящему ответу (помимо этого при помощи операторов паттернов можно задавать различную «важность паттернов»). После инициализации такой массив хранится в памяти устройства, что из-за большого количества категорий требует большого объема памяти. В используемой нами разработке изначально находилось более 50.000 категорий и ме-

сто, занимаемое ими, стремилось к 2 гигабайтам оперативной памяти.

Перед нами стояли задачи сокращения памяти, требуемой для инициализации массива. Это позволило бы запускать чат-бот на более широком списке устройств, ускорить обработку поиска ответа, облегчить и усовершенствовать написание паттернов и достичь сохранения контекста общения, что на данный момент является главной задачей всех разработчиков виртуальных собеседников.

III. ИНТЕГРАЦИЯ НОВЫХ ПАТТЕРНОВ

Возможность ускорение обработки ответа прямо пропорционально зависит от количества категорий, а количество категорий зависит от того, насколько просто и удобно будет задавать новые паттерны и покрывать множество вариантов диалогов, по которым пользователь может общаться с чат-ботом. Задача также усложняется особенностями русского языка, в предложениях которого допускается неоднозначный порядок слов, множество окончаний и смыслов у различных предложений, содержащих одинаковые слова.

Для улучшения паттернов были разработаны новые операторы. Например:

- Слов# – подходит под все слова, которые начинаются на «Слов» и продолжают как-нибудь окончанием;
- ^ екст – подходит под все слова, которые заканчиваются на «екст»;
- \$ - знак работает по такому же принципу, как * и _, но также учитывается, что вместо этого знака может и не стоять никаких слов. Таким образом мы предусматриваем возможность пропускать слова;
- [Слово1/Слово2/Слово3/..] – все слова в скобках разделенные слэшем («/») являются синонимами. При поиске введенного сообщения каждое из слов будет учитываться в этом месте предложения.

В результате, к примеру, может быть составлен следующий паттерн: * мам# \$ [мыл#/чист#/тер#] \$ рам# * Тогда, как не склоняй эту фразу: «Стиснув зубы, кряхтя и надрываясь, – моя милая мамочка продолжала тереть эту ужасную раму, а между тем – пот тек с её лба» – чат-бот все равно поймет правильно.

IV. РЕАЛИЗАЦИЯ НОВЫХ ТЕГОВ И УСОВЕРШЕНСТВОВАНИЕ СТАРЫХ

Помимо указанного выше, также нужно было внести разнообразие в ответы чат-бота, что было сделано при помощи реализации новых тегов. Например:

- <sticker> – тэг для отображения стикеров, имеющий атрибут «name», который может отображать одно из перечисленных состояний: Happy, Sad, Hilariously,

Laugh, VeryAngry, Angry, Cool, Love, Sweet, Smile, Cute, Cry, Neutral, Surprise, Kiss, Unhappy, Sick, Ups. Например: <sticker name="Cool"/> – отобразит смайлик, соответствующий тэгу Cool (эмоджи в очках);
- <male>, <m> и <female>, <f> – все, что находится внутри этих тэгов будет отображаться только если текущий пользователь девочка/мальчик (<male> и <m> отображает информацию для мальчиков, <female> и <f> – для девочек);
- /n – данные символы в <template> укажут боту, что слова слева и справа должны быть разделены на отдельные спич-баблы;
- <norepeat> – тэг исключающий повтор сообщения в рамках игровой сессии. Для его использования необходимо прописать уникальный идентификатор в атрибуте ID, который служит меткой для отслеживания повтора и может использоваться в рамках многих категорий.

Одна из важных особенностей чат-бота – это возможность запоминать информацию о которой ему говорит пользователь (имя, город, возраст и др.). Для этого существуют теги <get> и <set>. Но не было никакой возможности построить в категории ответ в зависимости от значения параметра, поэтому был расширен тег <condition>. Теперь внутри одного <condition> возможно прописать разное поведение в зависимости от значения в параметре и выглядит это следующим образом:

```
<condition name="USER_NAME">  
<li value=" $">Ты не сказал как тебя зовут</li>  
<li>Тебя зовут <get name="USER_NAME"/>  
</li>  
</condition>
```

V. ЗАКЛЮЧЕНИЕ

Благодаря данным разработкам, количество категорий сократилось до 20.000, соответственно уменьшилась и занимаемая ими память. При этом, данные 20.000 категорий, благодаря внесенным изменениям, покрывают большее количество различных фраз, понимаемых чат-ботом, и расширяют его возможности ответов. Главная задача, которая стоит перед нами сейчас и требует решения – это дальнейшее обучение чат-бота и добавление возможности сохранения контекста. Достичь этого мы планируем при помощи комбинирования технологий AIML с нейросетью.

СПИСОК ЛИТЕРАТУРЫ

1. Проватар А.И., Ключко К.А. Особенности и проблемы виртуального общения с помощью чат-ботов // Научные труды Винницкого национального технического университета. 2013. №3. С. 2.
2. Vinyals O., Quoc Le. A Neural Conversational Model // arXiv preprint arXiv:1506.05869, 22 Jul 2015.

АНАЛИЗ СИНХРОНИЗАЦИИ НЕЙРОННЫХ СЕТЕЙ В ПРИКЛАДНОЙ КРИПТОГРАФИИ

Урбанович П. П., Бирюк И. А., Плонковски М. Д.

Кафедра информационных систем и технологий, Белорусский государственный технологический университет, Минск, Республика Беларусь

Люблинский католический университет Яна Павла II, Люблин, Польша

prav.urb@yandex.by, plomyk@kul.pl

E-mail:

Рассматриваются важные аспекты использования нейросетевых технологий в задачах согласования криптографических ключей и вычисления хеш-функций. Архитектура сетей строится на основе известной древовидной машины четности Кантера-Кинцеля. Авторами доклада ранее предложена идея расширения используемой алгебры: действительные числа дополнены комплексными, а также кватернионами и октонионами. Это значительно повышает криптостойкость системы, но усложняет процесс взаимного обучения сетей. Приведены экспериментальные результаты и параметры этого процесса.

ВВЕДЕНИЕ

Современные информационные системы с повышенным уровнем безопасности строятся, в основном, на основе асимметричной криптографии и хеш-преобразований с требуемым уровнем необратимости. Однако главным недостатком асимметричных криптосистем является сложность математических операций, что значительно увеличивает время выполнения вычислений. Поэтому, по-прежнему все еще достаточно часто используются симметричные криптосистемы. Проблема создания быстрых протоколов согласования ключей до сих пор остается актуальной. Указанные обстоятельства вызвали интерес к разработке новых криптографических методов, не использующих в своей конструкции теорию чисел.

Использование в хеш-алгоритме Кессак входа и выхода переменной длины может применяться для генерации симметричных ключей из паролей, из протоколов согласования ключа по асимметричным алгоритмам.

Использование искусственных нейронных сетей (ИНС) для решения задач защиты информации впервые предложено И. Кантером и В. Кинцелем [1]. Развитием общей идеи стала известная архитектура ТРМ (англ. Tree Parity Machine, древовидная машина четности). При этом изначально функционирование ТРМ предусматривало использование целых действительных чисел как поля для описания и анализа процессов в сети.

Целью данного доклада является анализ процесса сходимости (синхронизации) ИНС на основе расширенной алгебры.

I. ХАРАКТЕРИСТИКА И ПАРАМЕТРЫ АРХИТЕКТУРЫ ТРМ

Архитектура ТРМ состоит из K независимых перцептронов, каждый из которых характеризуется N -элементным вектором весов $([w_{k1}, w_{k2}, \dots, w_{kN}]$, где $1 \leq k \leq K$).

Коэффициенты этих векторов – это целые числа из интервала $[-L, L]$. Входы перцептронов составляет K N -элементных векторов $([x_{k1}, x_{k2}, \dots, x_{kN}]$, где $x_{ki} \in [-1, 1]$).

Выходы нейронов составляют K значений $([z_1, z_2, \dots, z_k]$, которые равны сумме произведений элементов векторов входных значений на соответствующие элементы векторов весов [2].

Для решения задачи согласования ключевой информации между двумя сторонами применяются две идентичные сети. Все структурные элементы (как и в других криптографических системах) известны. Секретными элементами, на которых основывается криптостойкость системы – это начальные состояния векторов весов. Процесс взаимного обучения сетей или синхронизации начинается с инициализации векторов весов обеих сетей. Их начальное состояние, сгенерированное случайным образом, остается секретным на протяжении всего процесса обучения. Каждый шаг синхронизации начинается с подачи на входы обеих сетей определенного двоичного вектора, вычисления выходного значения сети и обмена выходными значениями между двумя сетями. В состоянии синхронизации значения весовых коэффициентов одинаковы и могут использоваться как совместный ключ обеих сторон (A и B). Основная проблема заключается в том, что третья сторона (C , интруз) может попытаться также достичь состояния синхронизации (с двумя сетями A и B).

II. РАСШИРЕНИЕ АЛГЕБРЫ ИСПОЛЪЗУЕМЫХ ЧИСЕЛ

Использование конструкции Кэли–Диксона (Cayley–Dickson) позволяет создавать расширения поля действительных чисел. Эта процедура дает возможность построить из действительных чисел последовательно их расширения: комплексные числа, кватернионы, октонионы, септенионы и т.д. Для криптографических применений разработаны модификации ТРМ с ис-

пользованием комплексных чисел (TPCM, Tree Parity Complex Machine), кватернионов (TPQM, TP Quaternion Machine) и октонионов (TROM, TP Octonion Machine) [2–4]. Сама архитектура указанных сетей, как и идея их функционирования, схожи с ТРМ. Изменения касаются методов применения правила обучения и модификации функции знака, относящегося к выходному параметру сетей A и B . Так, например, для TPCM выходы нейронов – это четырехвалентные комплексные числа, принадлежащие множеству $(1, 0), (0, 1), (-1, 0), (0, -1)$, а величины векторов весов – это комплексные числа, заключенные в квадрате $[-L, L] \times [-L, L]$.

Архитектура TPQM функционирует на основе кольца кватернионов. Правильное определение действий и изоморфизм с любым расширением тела действительных чисел, подтверждается теоремой Фробениуса. Кватернионы являются заключительным расширением тела действительных чисел, выполняющим условие ассоциативности операции умножения. Подтверждена эффективность использования модели TROM для вычисления хеш-функций [5].

В [6] проанализирована криптостойкость системы на основе ТРМ. Для защиты протокола при геометрической атаке, как минимум, необходимо значительное увеличение синаптической глубины, что также приводит к увеличению примерно на порядок времени, необходимого для наступления синхронизации. Таким образом, для обеспечения безопасности протокола при указанном способе атаки необходимы альтернативные способы защиты. Как показали исследования, проведенные авторами настоящего доклада (см. также [7]), имеется зависимость между наступлением процесса синхронизации сетей A и B и параметрами этих сетей. Ниже представлены новые результаты, касающиеся этого аспекта.

III. МОДЕЛИРОВАНИЕ И АНАЛИЗ ПРОЦЕССА СИНХРОНИЗАЦИИ ИНС

В нашем эксперименте две НС синхронизировали свои весовые коэффициенты обмениваясь выходными величинами на основе протокола TCP, причем пересылалась не явная информация, а хеш выходного значения на основе алгоритма SHA-512. Для сохранения результатов в разработанном приложении был использован EntityFramework и СУБД MS SQL Server. Весь процесс выполнялся с использованием авторского программного продукта, который позволял:

- выбирать конфигурацию сетей;
- подсчитывать количество проведенных опытов с данной конфигурацией;
- подсчитывать количество успешных опытов (сети синхронизованы);
- минимальное, максимальное и среднее количество итераций процесса синхронизации сетей;

Кроме того, мы получали отклонение полученных распределений от нормального распределения, а также индекс (метрика) Брея-Кертиса и расстояние по Хеллингеру. Для двух сетей с одинаковыми фиксированными параметрами проводилось не менее 1 тысячи опытов (попыток синхронизации). В таблице приведены некоторые результаты эксперимента. Здесь в четвертом столбце указано процентное отношение (u) числа синхронизаций к общему числу попыток синхронизации, в пятом столбце указано среднеквадратическое отклонение (СКО) и в шестом – индекс Брея-Кертиса (ИБК).

Таблица 1 – Результаты синхронизации весовых коэффициентов двух ИНС

Тип ИНС	K	N	$\pm L$	u	СКО	ИБК
TPCM	5	5	5	84,5	80,7	0,20
TPCM	6	7	6	92,9	68,6	0,17
TPQM	7	7	7	99,7	284,9	0,19
TROM	7	7	7	99,6	2779	0,23

Установлено, что распределение количества синхронизировавшихся НС по числу операций (итераций) обучения близко к нормальному. В диапазон «трех сигм» попадает 98.72 успешно синхронизировавшихся НС. Это можно использовать как параметр прерывания процесса синхронизации для снижения потенциальной возможности третьей стороны синхронизировать свои весовые коэффициенты с сетями A и B .

1. Kinzel, W. Interacting neural networks and cryptography / W. Kinzel, I. Kanter // *Advances in Solid State Physics*. – 2005. – Vol. 42. – P. 383–392.
2. Плонковский, М. Криптографическое преобразование информации на основе нейросетевых технологий / М. Плонковский, П. П. Урбанович // *Труды БГТУ. Сер. VI. Физико-математические науки и информатика*. – Минск: БГТУ. – 2005. – С. 161–164.
3. Plonkowski, M. The use of quaternions in the cryptographic key agreement protocol based on the architectures of the TPQM neural networks / M. Plonkowski, P. Urbanowicz, E. Lisitsa // *Przegląd elektrotechniczny*. – 2010. – Vol. 86, № 7. – P. 90–91.
4. Plonkowski, M. Split-complex numbers in neural cryptography / M. Plonkowski, P. Urbanowicz // *Przegląd elektrotechniczny*. – 2012. – Vol. 88, № 11b. – P. 340–341.
5. Urbanovich, P. The appearance of conflict when using the chaos function for calculating the hash code / P. Urbanovich, M. Plonkowski, K. Churikov // *Przegląd elektrotechniczny*. – 2012. – Vol. 88, № 11b. – P. 346–347.
6. Klimov, A. Analysis of Neural Cryptography / A. Klimov, A. Mityaguine, A. Shamir // *Advances in Cryptology – ASIACRYPT 2002: 8th International Conference on the Theory and Application of Cryptology and Information Security Queenstown, New Zealand, December 1–5, 2002, Proceedings*. P.288–298.
7. Урбанович, П. П. Моделирование и анализ процесса синхронизации нейронных сетей для обмена критической информацией / П. П. Урбанович, М. Долецки // *Материалы XVII МНТК «Комплексная защита информации. Безопасность информационных технологий»*, 18.05.2012, Суздаль. – 2012. – С. 255–257.

ПОСТРОЕНИЕ СКОРРЕЛИРОВАННЫХ ВИНЕРОВСКИХ ПРОЦЕССОВ

Фролов И. И.

Кафедра электронных вычислительных машин, Белорусский государственный университет информатики и радиоэлектроники
220013, П. Бровки, 6, Минск, Республика Беларусь
E-mail: frolov@bsuir.by

В работе рассматриваются особенности моделирования процессов рынка финансовых деривативов. Описаны вопросы построения скоррелированных винеровских процессов, используемых для моделирования цен финансовых инструментов. Приводится порядок использования алгоритма EWMA для расчета волатильности и последующего расчета коэффициентов корреляции, используемых при построении стохастических процессов.

ВВЕДЕНИЕ

Винеровские процессы являются важной составляющей стохастических моделей в финансовой математике. Зачастую характер движения цен нескольких интересующих финансовых инструментов демонстрирует определенную связанность между ними, обусловленную взаимным влиянием, настроением инвесторов, геополитическими факторами, оказывающими определенное влияние на цены. С математической точки зрения такую связанность принято выражать в виде коэффициентов корреляции, играющих весьма значимую роль при описании поведения финансовых инструментов.

I. СКОРРЕЛИРОВАННЫЕ ВИНЕРОВСКИЕ ПРОЦЕССЫ

В практике используется, как правило, стандартный подход построения скоррелированных Винеровских процессов W_1 и W_2 [1]. Пусть процессы Z_1 и Z_2 являются независимыми Винеровскими процессами. Тогда

$$dW_1(t) = dZ_1(t)$$

$$dW_2(t) = \rho dZ_1(t) + \sqrt{1 - \rho^2} dZ_2(t)$$

Нетрудно убедиться, что корреляция между dW_1 и dW_2 имеют дисперсию dt и коэффициент корреляции ρ :

$$D[dW_1(t)] = D[dZ_1(t)] = dt$$

$$D[dW_2(t)] = D[dZ_1(t)] + D[dZ_2(t)] =$$

$$= \rho^2 dt + (1 - \rho^2) dt = dt$$

$$Cov[dW_1(t)dW_2(t)] =$$

$$= \rho dZ_1 dZ_2 + \sqrt{1 - \rho^2} dZ_1 dZ_2 = \rho dt$$

Учитывая, что процессы Z_1 и Z_2 являются независимыми, то справедливо выражение $dZ_1 dZ_2 = 0$. Тогда коэффициент корреляции между dW_1 и dW_2 равен:

$$\rho = \frac{Cov[dZ_1, dZ_2]}{\sqrt{D[dZ_1]D[dZ_2]}} = \frac{\rho dt}{\sqrt{dt} \sqrt{dt}} = \rho$$

Аналогичным образом выводятся выражения и для многомерных случайных скоррелированных процессов [2]. Однако и в данном случае одним из значимых факторов является коэффициент корреляции ρ .

II. ОЦЕНКА ВОЛАТИЛЬНОСТИ С ИСПОЛЬЗОВАНИЕМ EWMA

Мерой отклонения, или колебания, цены актива на финансовом рынке выступает фактор волатильности, таким образом являющийся одновременно и общепринятым фактором меры риска. Для определения волатильности, как правило, используются разные источники входных данных, в зависимости от ее вида. Выделяют основные три вида волатильности [3]: историческая волатильность – рассчитывается на основании именно реальных исторических данных; ожидаемая, или подразумеваемая, волатильность – оценка выполняется на основании текущей стоимости торгуемых рыночных инструментов в предположении, что возможные риски учитываются в данном случае; историческая ожидаемая волатильность – является, по сути, зафиксированным списком исторических «расчитанных» значений ожидаемой волатильности.

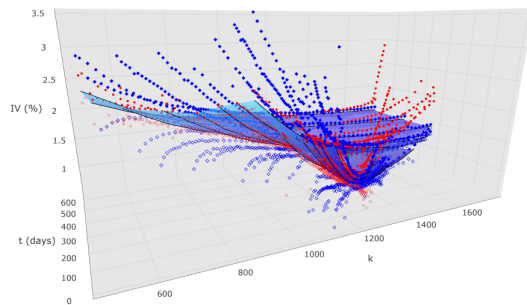


Рис. 1 – Пример поверхности ожидаемой волатильности для акций компании GOOGLE

При построении и последующей оценке стохастического процесса в симуляции более важ-

ным и представляющим интерес являются именно изменения характеристик финансового актива, нежели фиксированные состояния в определенный момент времени.

Расчет волатильности заключается в пошаговой оценке последовательных изменений цены исследуемого актива за выбранный период, и может быть представлен следующей последовательностью действий:

Шаг 1. Изначально нужно выбрать исследуемый период для выполнения расчетов, например, 1 год. На данном периоде необходимо выполнить «сбор данных»: изменения цены исследуемого актива на ежедневных интервалах. Для, например, акций или курса валют, такие ежедневные изменения (daily returns) выражаются, как правило, в логарифмической форме:

$$dailyLnRet = \ln \left(\frac{Price_i}{Price_{i-1}} \right)$$

где $Price_i$ представляет собой ежедневную цену закрытия для актива.

Шаг 2. Следующее действие – вычислить квадрат суммы ежедневных изменений за исследуемый интервал. На самом деле расчет сводится к вычислению простой дисперсии или волатильности, представленной следующей формулой:

$$\sigma_n^2 = \frac{1}{m} \sum_{i=1}^m u_{n-i}^2$$

где u представляет собой ежедневные изменения, и m обозначает число дней в исследуемом периоде.

Шаг 3. На данном этапе необходимо назначить веса таким образом, чтобы более свежие данные имели больший вес, чем более старые данные. Для этого вводится постоянный, сглаживающий, параметр λ . Веса назначаются как $(1 - \lambda)\lambda^0$. Т.е. параметр λ всегда должен быть меньше единицы. Как правило, общепринятой практикой является использование параметра $\lambda = 94\%$. Таким образом, первый вес будет равным $(1 - 0.94) = 6\%$, второй вес $6\% \times 0.94 = 5.64\%$ и т.д.

Шаг 4. Сумма перемноженных возведенных в квадрат ежедневных изменений цены $dailyLnRet's$ на соответствующие им веса w_i

$$sum = \sum_{i=1}^m dailyLnRet_i^2 \times w_i$$

Данное значение представляет собой конечную дисперсию σ^2 , тогда как волатильность σ будет равна $\sqrt{\sigma^2}$.

Обобщенная форма EWMA для вычисления волатильности может быть представлена в виде следующей рекурсивной формулы:

$$\sigma_n^2(EWMA) = \lambda\sigma_n^2 + (1 - \lambda)u_{n-1}^2$$

III. РАСЧЕТ КОВАРИАЦИИ

Согласно [2] коэффициент корреляции ρ может быть вычислен как

$$\rho = \frac{Cov_{1,2}}{\sigma_1\sigma_2} \quad (1)$$

где $Cov_{1,2}$ – ковариация между рядами изменений тех же активов, для которых рассчитываются и значения волатильности, вычисляемая по формуле

$$Cov_{1,2} = \sum_{i=1}^m dailyLnRet_{1,i} \times dailyLnRet_{2,i} \times w_i$$

Вычислив значения ковариации и волатильностей для исследуемых активов можно рассчитать искомое значение корреляции по формуле (1), и после – приступить к моделированию исходных скоррелированных стохастических процессов.

ЗАКЛЮЧЕНИЕ

Использование EWMA-подхода при вычислении исторической волатильности и последующего расчета коэффициентов корреляции позволяет оценить финансовые активы с точки зрения реальной меры риска, т.к. исторические данные уже фактически зафиксировали произошедшие события и цены. Такое сопоставление исторических оценок с прогнозированием позволяет получить более реалистичные оценки.

СПИСОК ЛИТЕРАТУРЫ

1. Tavella, D. Quantitative Methods in derivatives pricing. An Introduction to Computational Finance. / D. Tavella // New Jersey: John Wiley & Sons, Inc. – 2002. – 304 p.
2. Ширяев, А. Н. Основы стохастической финансовой математики: В 2 томах. Т. 1: Факты, модели // Москва: МЦНМО. – 2016. – 440 с.
3. Exploring the Exponentially Weighted Moving Average [Electronic resource] / Ed. M. Grant, D. R. Harper. – 2019. – Mode of access: <https://www.investopedia.com/articles/07/ewma.asp>. – Date of access: 07.10.2019.

ОБЪЕКТИВНАЯ ОЦЕНКА ЗАБОЛЕВАНИЙ ГОРТАНИ НА ОСНОВЕ АКУСТИЧЕСКОГО АНАЛИЗА ГОЛОСА

Федосеев Д. С., Шилин Л. Ю.

Кафедра информационных радиотехнологий, кафедра теоретических основ электротехники, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: precisionaudiosystems@gmail.com, dekfitu@bsuir.by

I. ВВЕДЕНИЕ

В настоящее время много внимания уделяется акустическому анализу голоса, позволяющему объективно оценить состояние голосовой функции гортани, что особенно важно для более точной постановки правильного диагноза с последующей оценкой восстановления голоса вследствие проведенного лечения. Следует учесть, что данных, характеризующих фонаторную функцию гортани при ее изменениях, которые связаны с различными типами функциональной дисфонии, встречается мало и они, как правило, отражают проблемы исключительно вокального голоса.

Для акустического анализ голоса у больных с функциональной дисфонией по гипотонусному типу, не являющимися профессионалами голоса, авторами предлагается использовать программно-аппаратный комплекс на базе персонального компьютера со встроенной звуковой картой и микрофоном, а также программой "Sound Forge", базирующийся на применении быстрого преобразования Фурье с различными типами усреднения, при помощи которой производился спектральный анализ голоса.

II. ИССЛЕДОВАНИЯ

Перед началом исследования определялся тоновой диапазон голоса пациента, т.е. расстояние от самой низкой до самой высокой ноты, которые может воспроизвести исследуемый[1]. Высота тона задавалась с помощью генератора звуковых частот в определённой последовательности, с интервалом в терцию, в пределах выявленного диапазона. У музыкально не ориентированных лиц, он, как правило, лежит в пределах 1 октавы в тесситуре, характерной для данного голоса. Далее пациент фонировал слог "мал" на forte и piano для определения динамического диапазона его голоса, т.е. разница силы звучания голоса при громком и тихом звукоизвлечении. Серьёзной проблемой, возникающей при записи, являлось либо отсутствие, либо неудовлетворительное развитие у исследуемого должного музыкального слуха, а также затруднения, возникшие с воссозданием заданного на генераторе звуковых частот тона. Данную проблему удалось практически полностью решить после адаптации слуха пациента к новой акустической сре-

де. Другим решением этой проблемы является «ступенчатое подведение» исследуемого к заданному тону посредством применения простых интервалов[2].

Алгоритм спектрального анализа голоса :

1. Находясь на строго заданном расстоянии от микрофона исследуемым на максимальной громкости поётся выбранная нота из середины рабочего диапазона.
2. На forte и piano последовательно поётся каждая нота, предварительно выбранная из ряда генератора звуковых частот, удерживая звук на 4/4 на слог «мал», с интервалом в одну терцию и ограниченная тональным диапазоном.
3. На примарных тонах на 4/4 фонировались слоги «мал», «мел», «мил», «мол» и «мул».

Полученные при проведённом исследовании результаты фиксируются в виде спектро-, соно- и фонограмм для дальнейшего анализа соответствующими специалистами.

Спектрограмма – это изменяющаяся в масштабе реального времени кривая, где по оси X откладывается частота звука, а по оси Y – соответствующая ей интенсивность. При анализе спектрограммы делается акцент на следующие её характеристики : количество и амплитуду пиков обертонов, а также правильность их формы и регулярность заполнения, значение чистоты межпиковой структуры, частотную границу, за которой обертоны перестают регистрироваться, наличие и количество заполненных обертонами зон формантного усиления, их устойчивость, интенсивность шумового компонента[3].

III. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТА

Проведенный авторами анализ показал, что спектрограммы у исследуемых со стойкими функциональными нарушениями голоса были различными в сравнении с контрольной группой. В свою очередь для каждой из групп были характерны определённые спектрографические кривые, из которых более выраженные нарушения были отмечены у пациентов со рецидивирующей функциональной афонией.

На Рис. 1 и Рис. 2 показаны спектрограммы на forte и piano соответственно пациента с диагнозом «Гипотонусная дисфония».

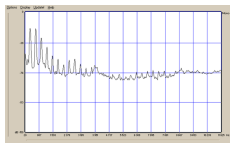


Рис.1

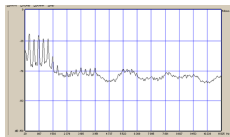


Рис.2

На Рис. 3 и Рис. 4 показаны спектрограммы на forte и piano соответственно пациента с диагнозом «Рецидивирующая функциональная афония».

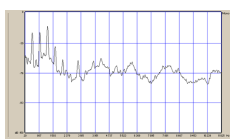


Рис.3

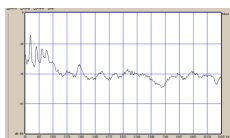


Рис.4

На Рис. 5 и Рис. 6 показаны спектрограммы на forte и piano соответственно пациента с диагнозом «Фонастения».

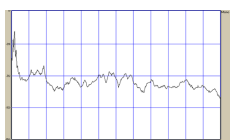


Рис.5

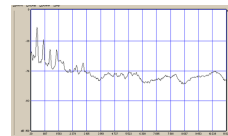


Рис.6

ЗАКЛЮЧЕНИЕ

Как видно из приведенных выше спектрограмм, у пациентов с диагнозом «Гипотонусная дисфония» отмечается уменьшение голоса на 10 дБ, а с диагнозом «Фонастения» - 15 дБ.

Речевой диапазон у исследуемых со стойкой функциональной гипотонусной дисфонией в среднем составлял 2 тона, а с диагнозом «Фонастения» - 1 тон.

Значения динамического диапазона у пациентов с диагнозом «Гипотонусная дисфония» лежали в пределах 7-8 дБ, а с «Фонастения» - 5 дБ.

Сужение динамического диапазона находилось в корреляционной зависимости от выраженности нарушения голосовой функции.

Исследование характеристик голоса больных с стойкими функциональными нарушениями голоса показало, что каждое функциональное нарушение гортани характеризуется своей оригинальной спектральной картиной, что в свою очередь, проявляется изменением обертонового состава, уменьшением речевого и сужением динамического диапазонов, практическим отсутствием формантных областей, уменьшением интенсивности голоса в среднем на 10-15 дБ.

СПИСОК ЛИТЕРАТУРЫ

1. Василенко, Ю. С. Голос. Фонологические аспекты. М 2002.
2. Wuyts, F. L. De Bodt, M. S Molenberghs, G. The dysphonia severity index: an objective measure of vocal quality based on a multiparameter approach. *Speech Lang Hear Res* 2000; 43: 3: 796-809.
3. Heylen, L. G. Wuyts, F. L Mertens, F. W. Pattyn, J. F. *Phonetography in voice diagnoses. Acta Otorhinolaryngol* 1996; 50: 4: 299-308.

THE STATISTICAL STABILITY OF CONSENSUS INDEPENDENT COMPONENT ANALYSIS FOR RNA-SEQ DATA IN CANCER RESEARCH

Chepeleva M., Yatskou M., Nazarov P.

Department of System Analysis and Computer Modeling, Belarusian State University
 Multiomics Data Science Research Group, Quantitative Biology Unit, Luxembourg Institute of Health
 Minsk, Belarus; Strassen, Luxembourg

E-mail: maryna.chepeleva@gmail.com, yatskou@bsu.by, petr.nazarov@lih.lu

Independent component analysis (ICA) became a part of the standard machine learning pipeline for genomics data analysis. The approach allows to correct technical biases and batch effects in transcriptomics datasets. Separated signals are successfully used to characterize biological functions, their weights might be used for diagnostics (cancer subtypes classification) and prognostics (survival prediction). Using weights of independent components as features for downstream analysis requires high reproducibility of decomposition. Here we investigated the stability of extracted components depending on ICA parameters and validated the optimal number of parallel consensus ICA runs that provided reproducible deconvolution. Also, we estimated the effect of parallel runs on the quality of lung cancer type classification (LUSC/LUAD) and gene enrichment analysis results. Finally, we estimated the boundary values for the number of components that allows detecting biologically relevant signals in smaller patient cohorts.

INTRODUCTION

Independent component analysis (ICA) allows decomposing heterogeneous transcriptomics data and extracting relevant transcriptional signals that correspond either to relevant biological processes or to technical biases [1]. Using independent components as features for downstream analysis requires high reproducibility of decomposition. Here we investigated the stability of ICA and tested reproducibility of its results for single and multiple runs, and in the case of reduced number of samples.

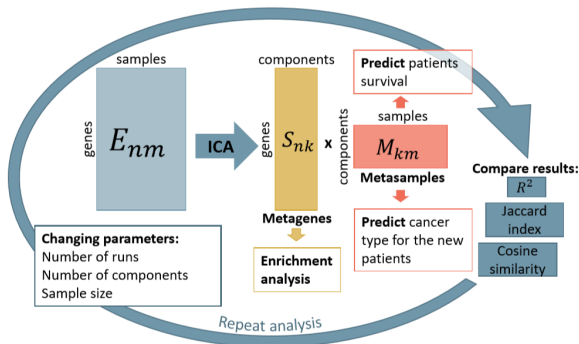


Рис. 1 – ICA decomposes gene expression matrix into meaningful signals S and weights M. Biological processes can be found in S, while M could be linked to patient cancer groups and patient survival. Changing input data and ICA parameters results can be compared using correlation and similarity metrics.

METHOD

We applied the developed parallel consensus Independent Component Analysis (ICA) algorithm [2] to TCGA [3] RNA-seq gene expression data on patients with skin cutaneous melanoma (SKCM) and non-small cell lung cancers: squamous cell carcinoma (LUSC) and adenocarcinoma

(LUAD). ICA finds a robust decomposition of an expression matrix: $E = S \times M$, where S is a matrix of statistically independent and biologically meaningful signals (metagenes) and M – their weights (metasamples). Functional annotation of components was performed by enrichment analysis using R package topGO [4] (biological processes are considered).

Two predicting models were used to classify the patients based on ICA results: random forest [5] and xgboost [6] from corresponding R packages. Metasamples (M-matrix), the most significant differentially expressed genes (by limma package [7]) and all the genes were used as input features to classifiers. To estimate patient survival, Cox-regression model was trained using ICA-based risk score, as in [1]. To explore how the stability of ICA depends on the parameters, we repeated the analysis on subsampled data sets and carried out pairwise comparison of calculated metagenes (S-matrix) or enriched gene ontologies (GOs) corresponding to each component. Independent components from different ICA runs were matched by maximum correlation. GO lists were compared using Jaccard index and cosine similarity between ranks of GO-term significance (Fig. 1).

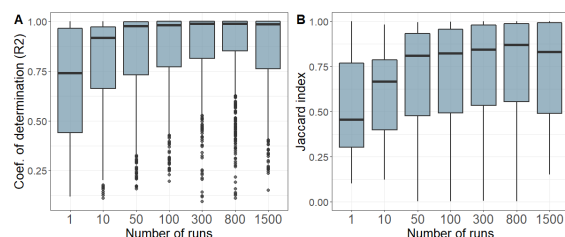


Рис. 2 – Stability of independent components (A) and significant gene signatures (B).

RESULTS

Multiple runs. Exploring dependence of the stability on the number of consensus ICA runs, we observed a strong increase of a squared correlation R^2 between corresponding independent signals (columns of S) and Jaccard indexes between contributing genes signatures (FDR < 0.05) as is shown in Fig. 2 A, B. GO similarity showed growing trend as well (Fig. 3A). For lung cancers 18 of components did not have enriched GOs (FDR < 0.05). These components may be linked to technical artefacts. Based on the presented profiles (Fig 2), the number of consensus parallel runs > 100 was enough to provide high reproducibility of ICA.

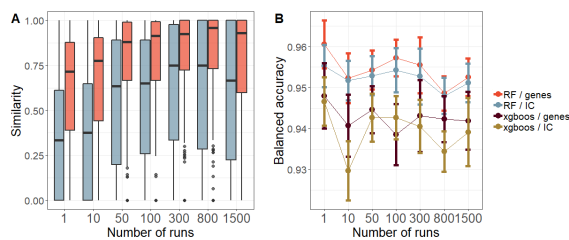


FIG. 3 – Stability of independent components (A) and significant gene signatures (B).

To validate the increase of reproducibility we performed two additional verifications. First, elements of each row in the data matrix were permuted, thus preserving distribution for each gene. Then ICA was performed on the permuted data. Second, we repeated permutations before each ICA. Slight increase was observed in the first case and no significant increase in the second (data not shown).

Average balanced accuracy did not have a considerable variation (Fig 3B). Therefore increasing number of ICA runs raises the stability but have no influence on classification accuracy.

Sample size. To investigate the dependency of ICA stability on the sample size, we fixed the number of components, as it can not be larger than the number of the samples. We selected equal number of LUSC and LUAD patients and a smaller number of normal samples from the dataset. The number of normal patients did not show any effect on the stability and classification accuracy. Fig. 4, 5 present the required sample size in order to achieve the median stability R^2 above 0.5 ICA with a small number of components (≤ 30) required less samples to reach plateau in the stability. However ICA with the low number of components may be not sensitive enough to detect all important biological signals and technical artefacts. Thus it is necessary to keep a balance between the number of components and the sample size. Interestingly, after a certain number of components, there is no improvement in classification accuracy with their increase.

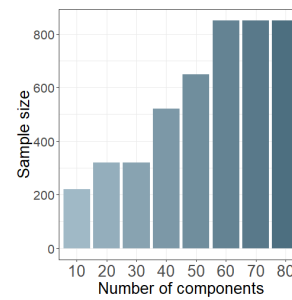


FIG. 4 – Required sample size to get median 0.5 stability (R^2) of metagenes for fixed number of components.

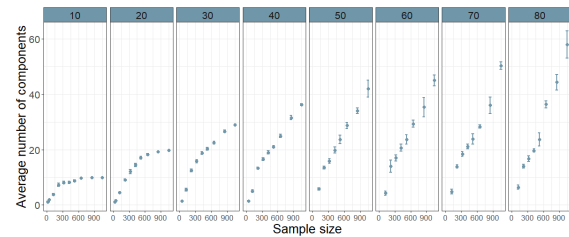


FIG. 5 – An average number of components with $R^2 > 0.5$ depending on the sample sizes.

CONCLUSION

To guarantee a high reproducibility of ICA 100 runs is sufficient. Random forest provides the highest accuracy on the significant genes but this approach loses in working time and interpretability. In order to detect more independent biological signals, more components should be used in ICA. However it requires a larger dataset.

ACKNOWLEDGEMENT

This work was supported by the Luxembourg National Research Fund (C17/BM/11664971/DEMICS).

1. Deconvolution of transcriptomes and miRNomes by independent component analysis provides insights into biological processes and clinical outcomes of melanoma patients / P. V. Nazarov [et al.] // BMC Med Genomics. – 2019. – V. 12, № 1. – P. 132–149.
2. Independent Component Analysis for Unraveling the Complexity of Cancer Omics Datasets / N. Sompairac [et al.] // Int J Mol Sci. – 2019. – V. 20, № 18. – P. 4414–4441.
3. Tomczak, K. The Cancer Genome Atlas (TCGA): an immeasurable source of knowledge / K. Tomczak, P. Czerwińska, M. Wiznerowicz // Contemp Oncol (Pozn). – 2015. – V. 19, № A1. – P. 68–77.
4. Alexa, A. Enrichment Analysis for Gene Ontology with topGO [Electronic resource] / A. Alexa, J. Rahnenfuhrer // Bioconductor. – Mode of access: <https://bioconductor.org/packages/release/bioc/html/topGO.html> – Date of access: 4.10.2019.
5. Liaw, A. Classification and Regression by randomForest / A. Liaw, M. Wiener // R News. – 2002. – V. 2, № 3. P. 18–22.
6. Chen, T. XGBoost: A Scalable Tree Boosting System / T. Chen, C. Guestrin // KDD '16 Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, California, USA, 2016. – P. 785–794.
7. Ritchie, M. E. Limma powers differential expression analyses for RNA-sequencing and microarray studies / M. E. Ritchie, B. Phipson // Nucleic Acids Research. – 2015. – Vol. 43. – P. e47.

СРАВНИТЕЛЬНАЯ ХАРАКТЕРИСТИКА МЕТОДОВ ВЫЯВЛЕНИЯ И ИДЕНТИФИКАЦИИ РИСКОВ

Черняк А. И., Поттосина С. А.

Кафедра экономической информатики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: alexanderchernyak1995@gmail.com, s.pottosina@gmail.com

Риск является неотъемлемой частью всякого проекта или процесса. Несмотря на то, что проекты могут быть связаны с большим или меньшим числом рисков, не существует ни одного проекта, полностью свободного от них. Всегда помня о невозможности избежать рисков, профессионалы постоянно ищут пути компромиссов между рисками и потенциальными возможностями и не становятся на путь абсолютной минимизации риска ценой исключения всего остального.

ОПИСАНИЕ МЕТОДОВ ВЫЯВЛЕНИЯ И ИДЕНТИФИКАЦИИ РИСКОВ

Существуют следующие методы выявления и идентификации рисков:

Мозговой штурм. Целью мозгового штурма является создание подробного списка рисков проекта. Список рисков разрабатывается на собрании, в котором принимает участие 10-15 человек – члены команды проекта, часто совместно с участием экспертов из разных областей, не являющихся членами команды. Участники собрания называют риски, которые считают важными для проекта, при этом не допускается обсуждение выдвинутых рисков. Далее риски сортируют по категориям и уточняют.

Метод Дельфи. Аналогичен методу мозгового штурма, но его участники не знают друг друга. Ведущий с помощью списка вопросов для получения идей, касающихся рисков проекта, собирает ответы экспертов. Далее ответы экспертов анализируются, распределяются по категориям и возвращаются экспертам для дальнейших комментариев. Консенсус и список рисков получается через несколько циклов этого процесса. В методе Дельфи исключается давление со стороны коллег и боязнь неловкого положения при высказывании идеи.

Метод номинальных групп. Позволяет выявить и расположить риски в порядке их важности. Данный метод предполагает формирование группы из 7-10 экспертов. Каждый участник индивидуально и без обсуждений перечисляет видимые им риски проекта. Далее происходит совместное обсуждение всех выделенных рисков и повторное индивидуальное составление списка рисков в порядке их важности.

Карточки Кроуфорда. Обычно собирается группа из 7-10 экспертов. Ведущий сообщает, что задаст группе 10 вопросов, на каждый из которых участник письменно, на отдельном листе бумаги, должен дать ответ. Вопрос о том, какой из рисков является наиболее важным для проекта, ведущий задает несколько раз. Каждый участник вынужден обдумать десять различных рис-

ков проекта. Опросы экспертов с большим опытом работы над проектами [1].

Идентификация основной причины. Цель этого процесса: выявление наиболее существенных причин возникновения рисков проекта. Это позволяет дать более точные определения рискам и сгруппировать риски по причинам, их вызывающих. Реагирование на риски может быть эффективным только тогда, когда оно направлено на устранение основной причины возникновения риска.

Анализ сильных и слабых сторон, возможностей и угроз (анализ SWOT). Цель проведения анализа - оценить потенциал и окружение проекта. Потенциал проекта, выраженный в виде его сильных и слабых сторон, позволяет оценить разрыв между содержанием проекта и возможностями его выполнения. Оценка окружения проекта показывает, какие благоприятные возможности предоставляет и какими опасностями угрожает внешняя среда.

Анализ контрольных списков. Контрольные списки представляют собой перечни рисков, составленные на основе информации и знаний, которые были накоплены в ходе исполнения прежних аналогичных проектов. Метод аналогии. Для идентификации рисков этот метод использует накопленные знания и планы по управлению рисками других аналогичных проектов.

Опрос экспертов. Проведение опросов среди опытных сотрудников, принимающих участие в проекте, среди участников проекта и экспертов в этой области, может способствовать идентификации рисков. Результаты опросов являются одним из основных источников информации в процессе сбора данных об идентификации рисков.

Методы с использованием диаграмм. К методам отображения рисков в виде диаграмм относятся диаграммы причинно-следственных связей и блок-схемы процессов, которые позволяют проследить последовательность событий, происходящих в данном процессе [2].

В таблице 1 представлено сравнение методов выявления и идентификации рисков.

ЗАКЛЮЧЕНИЕ

По результатам сравнительного анализа, приведенного в таблице 1, следует отметить, что ни один из методов не лишен недостатков. В связи с этим, рекомендуется на практике использовать комбинацию из представленных в докладе методов [3]. Это позволяет снизить влияние недостатков каждого из используемых в комбинации методов и приводит к более конкретной и точной идентификации рисков. Например, если совместить любой из методов с методами с использованием диаграмм, то в результате получится не только текстовое представление возможных рисков, но еще и графическое.

Своевременное выявление и идентификация рисков способствуют скорейшему реагированию на данные риски, что позволяет минимизировать негативное влияние найденного риска на проект.

СПИСОК ЛИТЕРАТУРЫ

1. Новиков, Д. А. Управление проектами: организационные механизмы / Д. А. Новиков – М. :ПМСОФТ – 2007. – 140 с.
2. Арчибальд, Р. Д. Управление высокотехнологичными программами и проектами / Р. Д. Арчибальд – М. :ДМКПресс – 2002. – 472 с.
3. Заренков, В. А. Управление проектами / В. А. Заренков // Издательство: Ассоциации строительных вузов, СПбГАСУ – Москва. – 2006. – 312 с.

Таблица 1 – Сравнение методов выявления и идентификации рисков

Метод идентификации.	Преимущества	Недостатки
Опрос экспертов.	Используется прошлый опыт.	Предвзятость экспертов. Требуется много времени.
Мозговой штурм.	Способствует взаимодействию членов группы. Быстрый. Недорогой.	Может проявиться преобладание одной личности. Можно сосредотачиваться только в конкретных областях. Требуется опытного, сильного, объективного ко всем членам группы ведущего. Для оценки необходимо контролировать склонности группы.
Метод Delphi.	Нет доминирования одной личности. Может проводиться дистанционно, через электронную почту. Исключается проблема ранней оценки. Требуется участия каждого члена группы.	Занимает много времени. Высокая загрузка ведущего.
Метод номинальных групп.	Уменьшается эффект доминирующей личности. Обеспечивает взаимодействие участников. Дает упорядоченный список рисков.	Требуется много времени. Высокая загрузка ведущего.
Карточки Кроуфорда.	Быстрый. Легко реализуется. Должен участвовать каждый член группы. Вырабатывается большое количество идей. Можно проводить с группами большеобычного размера. Уменьшает эффект доминирующей личности.	Меньшее взаимодействие между участниками.
Контрольные списки.	Конкретный и упорядоченный. Легко использовать.	Предвзятость. Может не содержать конкретных элементов для данного проекта.
Метод аналогии.	Использует прошлый опыт для исключения проблем в будущем. Подобные проекты содержат много сходных черт.	Требуется много времени. Легко получить результаты, не подходящие для данного случая. Аналогия может быть некорректной.
Методы с использованием диаграмм.	Ясное представление участвующих процессов. Легкость построения. Для них имеется много компьютерных инструментов.	Иногда вводит в заблуждение. Может занимать много времени.

AN IMPROVED FORMING ALGORITHM OF ERROR-PATTERNS FOR TWO-DIMENSIONAL CODES

Ren Xunhuan, Ma Jun, Konopelko V. K.
 Department of Infocommunication Technologies ,BSUIR
 Minsk, Belarus
 E-mail: rxh1549417024@gmail.com

Developments of techniques for processing two-dimensional information have aroused interest in two-dimensional codes. [1] have investigated the two-dimensional codes called - $\beta\gamma$ array codes. Nonzero codewords of these codes are two-dimensional versions of M-sequences and have interesting properties that are essentially two-dimensional.[2] has made studies on product codes from a two-dimensional viewpoint and has shown that the product of an (n_1, k_1) cyclic code and an (n_2, k_2) cyclic code permits the correction of every two dimensional burst (or spot) of area $(n_1 - k_1)$ or (n_2, k_2) or less. In this paper, we analyze the methods for forming error vectors and the properties of error vectors, which are allowed to determining the type of error vector in two-dimensional coding of information.

INTRODUCTION

Error-correcting codes have their origin in Shannon's seminal publication from 1948, where he proved that nearly error-free discrete data transmission is possible over any noisy channel when the code rate is less than the channel capacity. This statement is nowadays called the (noisy) channel coding theorem. The channel capacity depends on the physical properties of the channel and it is an active research area to determine the capacity of non-trivial communication channels. However, the proof of the channel coding theorem is non - constructive and therefore it is not clear how to construct error - correcting codes which actually achieve the Shannon limit. Noise - resistant codes are designed for synchronous noise and noise control in real channels of digital information. Nowadays, they have become an inevitable attribute of almost all mobile telecommunication systems (*TCS*). The history of the rapid development of the theory and practice of error -correcting coding has a little more than 70 years and Hamming R. laid the foundations of noise-resistant coding, proposed the first real noise - resistant codes [3].

In this paper, we analyze the methods for classifying error vectors and the properties of error vectors, which are allowed to determining the type of error vector in two-dimensional coding of information. On the basis of [4], a new method and algorithm for classifying error vectors are proposed.

The rest of the paper is organized as follows. In Section 1, a brief description of product codes. Section 2, different algorithms for generating the library of $t = 2:6$ error patterns. The conclusion is given in Section 3.

I. PRINCIPLE OF PRODUCT CODE

Product code is a technique to form a long length code with higher ECC capabilities using small length constituent codes. Compared to plain long length codes, it has high performance from cross parity check and low circuitry overhead

since the constituent codewords are of low error correction capability. Let C_1 be a (n_1, k_1) linear code, and let C_2 be a linear code. Then, a $(n_1 n_2, k_1 k_2)$ linear code can be formed such that each codeword is a rectangular array of n_1 columns and n_2 rows in which every row is a code word of C_1 , and every column is a code word in C_2 , as shown in Figure 1. This two-dimensional code is called the direct product (or simply the product) of C_1 and C_2 . The $k_1 k_2$ digits in the upper right corner of the array are information symbols. The digits in the upper left corner of this array are computed from the parity-check rules for C_1 on rows, and the digits in the lower right corner are computed from the parity-check rules for C_2 of columns. Now should we compute the check digits in the lower-left corner by using the parity-check rules for C_2 on columns or the parity-check rules for C_1 on rows. It turns out that either way yields the same $(n_1 - k_1) \times (n_2 - k_2)$ check digits, and it is possible to have row code words in C_1 and all column code words in C_2 simultaneously [3].

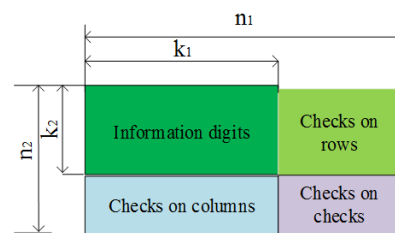


Рис. 1 – A typical two-dimensional product code

Here we take two Hamming codes as the code and illustrate the encoding process of the product code, for example, the $(7 \times 4) \times (7 \times 4)$ coding matrix is shown in the figure 2, where m represents the information bit, p indicates the corresponding parity bit, and the subscript i and j represent the number of rows respectively.

For example, firstly encoding the row (column) and then encoding the column (row), the row encoder encodes the first row of information bits

$(m_{11}, m_{21}, m_{31}, m_{41})$ according to the encoding rule of the $(7, 4)$ Hamming code, calculates the corresponding check bit (p_{51}, p_{61}, p_{71}) and adds it to the right of $(m_{11}, m_{21}, m_{31}, m_{41})$, and then the encoder turns to the next row. Every line performs the same encoding process until the fourth line of the information bit is completed, that's $(m_{14}, m_{24}, m_{34}, m_{44})$ encoding which can form a (4×7) matrix; then the column encoder also encodes the first line of information bits $(m_{11}, m_{21}, m_{31}, m_{41})$ according to the coding rules of the $(7, 4)$ Hamming code, calculate the corresponding check bits (p_{15}, p_{16}, p_{17}) and added under the information bits $(m_{14}, m_{24}, m_{34}, m_{44})$. The column encoder then moves to the next column to perform the same encoding process. After completion the information bit encoding in the fourth column, the column encoder continues to encode the third column of the check bits in the right to obtain the checks on checks bits, and finally completes the product code. When the code is transmitted in the channel, it will be transmitted serially according to the sequence $(m_{11}, m_{21}, m_{31}, m_{41}, p_{51}, p_{61}, p_{71}, m_{12}, m_{22}, \dots, p_{71})$. At the receiving end, the decoder will still arrange the received serial sequence according to the above way, convert it into a two-dimensional matrix, and then decode it according to the matrix structure. Product codes are usually encoded by simple system codes, and decoding is usually done by first column decoding. Therefore, the complexity of decoding increases linearly with the decoding complexity of its subcodes.

m_{11}	m_{21}	m_{31}	m_{41}	p_{51}	p_{61}	p_{71}
m_{12}	m_{22}	m_{32}	m_{42}	p_{52}	p_{62}	p_{72}
m_{13}	m_{23}	m_{33}	m_{43}	p_{53}	p_{63}	p_{73}
m_{14}	m_{24}	m_{34}	m_{44}	p_{54}	p_{64}	p_{74}
p_{15}	p_{25}	p_{35}	p_{45}	p_{55}	p_{65}	p_{75}
p_{16}	p_{26}	p_{36}	p_{46}	p_{56}	p_{66}	p_{76}
p_{17}	p_{27}	p_{37}	p_{47}	p_{57}	p_{67}	p_{77}

Рис. 2 – A schematic view of the (7×7) product code encoding

II. ALGORITHM FOR GENERATING ERROR PATTERN

The known classification algorithm of generation error vector [4] The error pattern can be equivalent to a matrix of $(t \times t)$. For each square matrix containing t error modes, according to the definition, we consider that the two matrices are equivalent if they can be transformed into each other through the elementary transformation. The rules which can generate the library error patterns [4].

1. Generating all of the original vectors V_o on length $n_1 = (t \times t)$;
2. From all of the vector V_o calculate the vector V_f ;

3. If there is a class whose vectors formed the same error pattern, the vector is excluded from consideration;
4. If there is no such class, the vector becomes the pattern of the new class ;
5. Generating all of the original vectors V_o on the length of n_1 ;
6. Compare the patterns of classes in order to generate typical patterns classes .

According to the algorithm of [4], we proposed an improved algorithm which uses the structure of trees. Firstly, based on the error pattern of t , we adding one random error and generating the error patterns of $t + 1$, finally, we get an error pattern library (Fig.3).

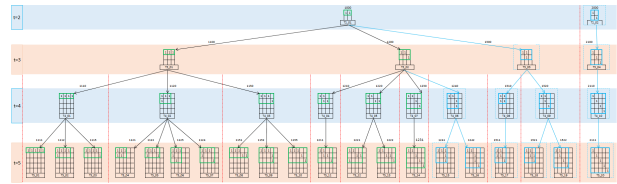


Рис. 3 – The Structure generating error patterns for $t = 2 : 5$

Таблица 1 – Performance comparison of the methods of forming the error vector

t	V_f	Fast method V_f	Acceleration factor
2	6	3	2
3	84	14	6
4	1820	52	35
5	53130	210	253
6	1947792	1054	3141

III. CONCLUSION

In this paper, we analyze different methods for generating the library of error patterns on the basis of two-dimension coding. The improved method has reduced the time of generating the error library. The experiment results indicate that the new method has a good performance.

1. Nomura, T. H. A theory of two-dimensional linear recurring arrays / T. H. Nomura, I. Miyakawa, A. Fukuda //IEEE Trans. Inform. –Nov.1972. – № IT-18. – C. 775–785.
2. Elspas, B. Notes on multidimensional burst-error correction / B. Elspas //IEEE Int. Svmm. Information Theorv. San Remo.Italy. – 1967.
3. Daniel, J, C. Error Control Coding, Second Edition / J. C. Danie,l, SHU //(ISBN 0-13-042672-5). – 2004. – . C. 44–89.
4. Smolyakov, O. G. Classification of error vectors in two-dimensional information coding / O. G.O. GSmolyakova ,V. K. Konopelko // – 2008. – . C. 19–28.
5. Smolyakova, .O. G. Correction of errors and erasures in two-dimensional coding of information / O. G.O. GSmolyakova ,V. K. Konopelko // – 2008. – . C. 142–153.
6. Sung, W. Y. Two-Dimensional Error-Pattern-Correcting Codes / W. Y. Sung, M. Jaekyun //IEEE Trans. Inf.Theory, vol. 63 . – Aug.2015. – . C. 2725–2740.

КЛАССИФИКАЦИЯ МОЛЕКУЛ РНК

Яцков Н. Н., Климук И. В., Скакун В. В., Гринев В. В.

Кафедра системного анализа и компьютерного моделирования, кафедра генетики, Белорусский государственный университет
Минск, Республика Беларусь

E-mail: yatskou@bsu.by, grinev_vv@bsu.by, skakun@bsu.by, ivanklimuk96@gmail.com

Разработаны три модели классификации молекул РНК, на основе: 1) алгоритмов векторизации молекул РНК и классификации случайного леса; 2) определения открытых рамок считывания в молекулах РНК; 3) кодирования единиц («one hot encoding») с последующим использованием одномерной сверточной нейронной сети. Проверка работоспособности разработанных алгоритмов выполнена на примере набора РНК человека из базы данных NCBI RefSeq. Точность классификации молекул с использованием разработанных моделей варьируется от 90,4 до 99,8%.

ВВЕДЕНИЕ

Транскрипция генома приводит к образованию матричных РНК, а также разнообразных малых и длинных некодирующих РНК [1]. Длинные некодирующие РНК тесно связаны со многими биологическими процессами, такими как многоуровневая регуляция экспрессии генов. Их структура во многом схожа со структурой матричной РНК, что значительно осложняет задачу точного определения вида молекул, основываясь только на нуклеотидной последовательности. Для решения данной задачи требуется разработка классификационной модели, позволяющей эффективно определять кодирующие и некодирующие молекулы РНК. В работе [2] представлена классификационная модель на основе к-меров и сверточных нейронных сетей, точность классификации которых варьируется от 87,97 до 99,63% для молекул РНК мышей и кур соответственно, для организма же человека точность равна 98,72%. Однако данная модель имеет существенные ограничения, к которым можно отнести ресурс-затратные процедуры определения к-меров, сложность статистического анализа в определении значимости к-меров, определённые требования к вычислительным ресурсам.

Целью данной работы является разработка эффективных математических моделей классификации кодирующих и некодирующих молекул РНК человека, устраняющих недостатки модели [2]. Разработаны три классификационные модели на основе следующих алгоритмов: 1) векторизации молекул РНК и классификации случайного леса; 2) определения открытых рамок считывания (ОРС) молекул РНК; 3) кодирования единиц с последующим использованием одномерной сверточной нейронной сети. Проверка работоспособности разработанных алгоритмов выполнена на примере набора РНК человека из базы данных NCBI RefSeq.

I. МОДЕЛИ КЛАССИФИКАЦИИ

Модель 1. Включает алгоритмы векторизации [3] и случайного леса [4]. Векторизация нук-

леотидных последовательностей произведена в 104 признака (частоты моно-, ди- и тринуклеотидов [3], параметры модели Вао [5], корреляционные факторы нуклеотидов [6], длины последовательностей). Обучение модели производится на выборках кодирующих и некодирующих молекул РНК. Очевидными достоинствами модели являются простота, в том числе алгоритмической и программной реализации, минимальные требования к вычислительным ресурсам, интерпретируемость параметров векторизации.

Модель 2. В основе алгоритма классификации лежит модель 1, обученная на выборках истинных и ложных ОРС, полученных из кодирующих и некодирующих молекул, и алгоритм определение ОРС. Если в молекуле определена ОРС, то молекула считается кодирующей, иначе некодирующей. Преимущества модели: 1) простота программной реализации; 2) умеренные требования к вычислительным ресурсам; 3) возможность прямого использования для решения важной задачи молекулярной биологии – определение ОРС молекул.

Модель 3. Модель представляет двухэтапный алгоритм. На первом этапе осуществляется так называемое кодирование единицей («one hot encoding») молекул РНК [7]. На втором этапе используется одномерная сверточная нейронная сеть [7]. Обучение модели производится на выборках кодирующих и некодирующих молекул РНК. Данная модель фактически является улучшенным вариантом модели [2], однако вместо выполнения процедуры параметризации в векторное пространство к-меров выполняется кодирование единицей, эффективность которого, для решения подобной задачи, подтверждена в [7]. Очевидным недостатком данной модели является требование значительных вычислительных ресурсов.

II. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ И ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

В ходе анализа моделей 1-3 рассмотрены 4235 некодирующих молекул РНК и 5000 случайно отобранных кодирующих молекул РНК

из базы данных NCBI RefSeq. Для модели 2 класс псевдо ОРС-последовательностей содержит 109230 нуклеотидных фрагментов, полученных из 4235 некодирующих молекул РНК. Класс истинных ОРС-последовательностей включает 108654 реальных ОРС молекул РНК. Обучающая выборка моделей 1 и 2 двух типов молекул включала 75% исходных данных, тестируемая – 25%. Обучающая выборка модели 3 – 90% исходных данных, тестируемая – 10%. Для модели 3 выбрана обучающая выборка наибольшего размера с целью увеличения точности классификатора (в задачах классификации генетических данных нейронные сети требуют обучающие выборки больших размеров) [9]. Процентная оценка точности классификации производится как отношение числа верно классифицированных молекул к общему числу.

III. РЕЗУЛЬТАТЫ

Вычислительные алгоритмы моделей 1 и 2 реализованы на языках программирования R и C++ с использованием открытых библиотек R-функций проектов Bioconductor и CRAN. Анализ данных выполнен на вычислительном сервере, основные характеристики которого – 12 ядерный процессор Intel i9 (3.9 GHz), 64 Gb RAM, 8 Tb HDD. Время вычислений – 2-5 минут для модели 1 и 20-30 мин для модели 2.

Алгоритм модели 3 реализован на языке программирования Python с использованием открытых библиотек нейронных сетей глубокого обучения PyTorch и анализа данных scikit-learn. Анализ данных выполнен на 24-ядерной виртуальной машине в облачном сервисе Google Cloud. Время вычислений – 10 часов.

Модель 1. Точность классификации кодирующих и некодирующих РНК – 90,35%. Оценена информативность признаков векторизации молекул с использованием критерия на основе индекса Джини [8], встроенного в алгоритм случайного леса. Наиболее информативными признаками являются признаки модели Вао и частоты появления некоторых тринуклеотидов. Менее информативными признаками являются корреляционные факторы нуклеотидов.

Модель 2. Успешно выполнен анализ молекул РНК. Точность классификации – 96,67%. Точность классификации сопоставима с точностью модели [2], однако скорость вычислений существенно выше, требования к использованию вычислительных ресурсов ниже.

Модель 3. Точность классификации нейронной сети с 9-тью одномерными сверточными слоями и 2-3 эпохами обучения классификации – 99,80%, наивысшая среди рассматриваемых моделей. Однако системные требования существенные – одна эпоха обучения длится около 1 часа. Для конвейерного анализа молекул РНК, дан-

ные по которым генерируются с помощью транскриптомного секвенирования, требуются огромные системные ресурсы. Еще одним недостатком модели является известная склонность алгоритмов к переобучению, что может привести к ложной классификации молекул РНК в ходе проведения новых экспериментальных исследований.

IV. ЗАКЛЮЧЕНИЕ

Разработаны три модели классификации кодирующих и некодирующих молекул РНК, точность которых варьируется от 90,4 до 99,8%. Определён набор наиболее информативных признаков молекул РНК – это признаки модели Вао и набор частот тринуклеотидов.

Модели на основе алгоритмов векторизации молекул и классификации случайного леса практически не уступают в точности опубликованной модели [2], однако существенно превосходят по вычислительной производительности, статистической значимости и интерпретируемости параметров, требуют меньше вычислительных ресурсов.

Классификатор на основе одномерной сверточной нейронной сети является наилучшим, однако требует значительные вычислительные ресурсы (включающие мощные графические карты или многопроцессорные вычислительные серверы (более 24-х ядер)) и имеет склонность к переобучению.

СПИСОК ЛИТЕРАТУРЫ

1. Djebali, S. Landscape of transcription in human cells / S. Djebali, C. A. Davis, A. Merkel [et al.] // *Nature*. – 2012. – Vol. 489, – P. 101–108.
2. Wen, J. Classification model for lncRNA and mRNA based on k-mers and a convolutional neural network / J. Wen, Y. Liu, Y. Shi, H. Huang [et al.] // *BMC Bioinformatics*. – 2019. – Vol. 20, № 1:469. – P. 1–14.
3. Разработка алгоритмов и программных средств классификации кодирующих и некодирующих нуклеотидных последовательностей / В. П. Закирова [и др.] // *Информатика*. – 2019. – Т. 16, № 2. – С. 111–120.
4. Breiman, L. Random forest / L. Breiman // *Machine Learning*. – 2001. – Vol. 45, № 1. – P. 5–32.
5. Bao, J. An improved alignment-free model for DNA sequence similarity metric / J. Bao, R. Yuan, Z. Bao // *BMC Bioinformatics*. – 2014. – Vol. 15, № 321. – P. 1–15.
6. Comparative analyses between retained introns and constitutively spliced introns in arabidopsos thaliana using random forest and support vector machine / R. Mao [et al.] // *PLoS One*. – 2014. – Vol. 9, № 8. – P. 1–12.
7. Al-Ajlan, A. CNN-MGP: Convolutional Neural Networks for Metagenomics Gene Prediction / A. Al-Ajlan, A. El Allali // *Interdiscip Sci*. – 2018. – DOI: 10.1007/s12539-018-0313-4.
8. Интеллектуальный анализ данных / Н. Н. Яцков – Минск: БГУ, 2014. – 151 с.
9. Chen, D. Deep learning and alternative learning strategies for retrospective real-world clinical data / D. Chen, S. Liu, P. Kingsbury [et al.] // *npj Digital Medicine*. – 2019. – Vol. 2, № 43. – P. 1–5.

АВТОМАТИЧЕСКОЕ ОПРЕДЕЛЕНИЕ ОТКРЫТЫХ РАМОК СЧИТЫВАНИЯ В МОЛЕКУЛАХ РНК ЧЕЛОВЕКА С ИСПОЛЬЗОВАНИЕМ АЛГОРИТМОВ ВЕКТОРИЗАЦИИ И КЛАССИФИКАЦИИ

Яцков Н. Н., Скакун В. В., Гринев В. В.

Кафедра системного анализа и компьютерного моделирования, кафедра генетики, Белорусский государственный университет

Минск, Республика Беларусь

E-mail: yatskou@bsu.by, grinev_vv@bsu.by, skakun@bsu.by

Разработан вычислительный подход для автоматического определения открытых рамок считывания в большом наборе молекул РНК на основе алгоритмов векторизации нуклеотидных последовательностей и классификации (случайного леса). Проверка работоспособности алгоритмов выполнена на двух наборах молекул РНК из баз данных NCBI RefSeq и Ensembl. Точность определения открытых рамок считывания составляет 98,14%.

ВВЕДЕНИЕ

Современные методы полнотранскриптомного секвенирования [1, 2] позволяют точно установить нуклеотидные последовательности молекул РНК, присутствующих в клетке, а также определить их количественное содержание. При этом кодирующий потенциал таких молекул может быть оценен с помощью алгоритмов определения открытых рамок считывания (ОРС), реализованных в программных пакетах NCBI ORFfinder [3] или CPC2 [4]. Однако данные алгоритмы предназначены для одномолекулярного анализа и не позволяют сделать обоснованный выбор одной из открытых рамок считывания в случае множественности таковых в изучаемой молекуле РНК.

В настоящей работе предложен вычислительный подход для автоматического определения ОРС в большом наборе молекул РНК на основе алгоритмов векторизации нуклеотидных последовательностей и классификации (наиболее вероятной ОРС).

I. ВЫЧИСЛИТЕЛЬНЫЙ ПОДХОД ДЛЯ ОПРЕДЕЛЕНИЯ ОРС

Вычислительный подход включает алгоритмы векторизации [5] и случайного леса [6]. Векторизация нуклеотидных последовательностей произведена в 104 признака (частоты моно-, ди- и тринуклеотидов [5], параметры модели Вао [7], корреляционные факторы нуклеотидов [8], длины последовательностей). Этапы анализа.

1. Формирование наборов данных для обучения, представляющих классы истинных (кодирующих) и псевдо (некодирующих) ОРС-кандидатов.

2. Векторизация фрагментов нуклеотидных последовательностей молекул в 104 признака.

3. Обучение метода случайного леса на эталонном наборе данных. Оценка точности (ошибки) классификации на тестируемом наборе дан-

ных. Экспорт классификационной модели для определения ОРС молекул РНК.

4. Анализ исследуемых молекул РНК с целью точного определения ОРС: i) нахождение всевозможных ОРС-кандидатов в молекуле; ii) точное определение ОРС с использованием классификационной модели п. 3.

II. ЭКСПЕРИМЕНТАЛЬНЫЕ ДАННЫЕ И ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

В ходе анализа рассмотрены 4235 некодирующих молекул РНК, не имеющих ОРС, и 113063 кодирующих молекул РНК из базы данных NCBI RefSeq. Класс псевдо ОРС-последовательностей содержит 109230 нуклеотидных фрагмента, полученных из 4235 некодирующих молекул РНК. Класс истинных ОРС-последовательностей включает 108654 реальных ОРС из молекул РНК. Оценка точности определения ОРС произведена на полном наборе кодирующих молекул РНК. Для оценки точности определения ОРС используются координаты ОРС молекул, представленные в базе данных NCBI RefSeq. Для дополнительного подтверждения разработанной классификационной модели рассмотрены 63832 кодирующих молекул РНК с точными координатами ОРС из базы данных Ensembl. Процентная оценка точности определения ОРС молекул производится как отношение числа верно классифицированных ОРС молекул к общему числу рассматриваемых молекул.

III. РЕЗУЛЬТАТЫ

Вычислительные алгоритмы реализованы на языках программирования R и C++ с использованием открытых библиотек R-функций проектов Bioconductor и CRAN. Анализ данных выполнен на вычислительном сервере, основные характеристики которого – 12 ядерный процессор Intel i9 (3.9 GHz), 64 Gb RAM, 8 Tb HDD. Время вычислений – 14 часов.

Визуализация результатов векторизации ОРС-последовательностей с использованием метода главных компонент [9] представлена на рис. 1. Два класса ложных и истинных ОРС последовательностей разделяются.

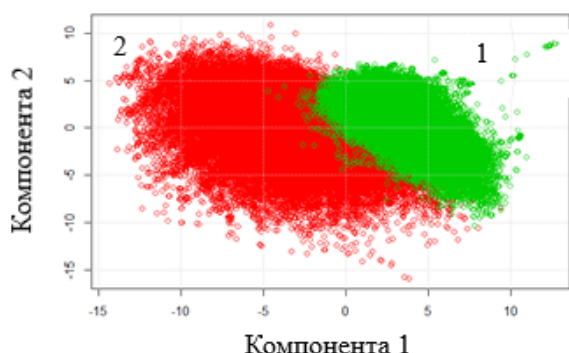


Рис. 1 – Результаты применения метода главных компонент к векторизованному набору данных: ОРС-кандидаты кодирующих (1) и не кодирующих (2) молекул РНК в пространстве первых двух главных компонент

Успешно выполнен анализ молекул РНК с целью точного определения ОРС. Обучающая выборка ОРС-кандидатов двух типов включала 75% исходных данных, тестируемая – 25%. Точность классификации истинных (кодирующих) и псевдо (не кодирующих) ОРС-кандидатов – 99,35%. Оценена информативность признаков фрагментов нуклеотидных последовательностей молекул с использованием критерия на основе индекса Джини [9], встроенного в алгоритм случайного леса. Наиболее информативными признаками являются признаки модели Вао и два варианта вычисления длины ОРС (в количестве нуклеотидов и с использованием логарифмирования). Менее информативными признаками являются частоты различных комбинаций нуклеотидов и корреляционные факторы нуклеотидов. Для точной классификации фрагментов нуклеотидных последовательностей достаточно использовать 25 наиболее информативных признака. Разработанный классификатор применен для нахождения ОРС 113063 кодирующих молекул РНК (с известными ОРС). Точность нахождения – 98,14% (точно определены ОРС 110959 молекул). Дополнительно проверена работоспособность классификатора на наборе 63832 кодирующих молекул РНК из базы данных Ensembl – точность определения ОРС – 98,14%.

Следует отметить важный сопутствующий вывод по результатам нашей работы, который позволяет предположить о невысокой значимости консенсусных последовательностей Козак [10] при определении координат ОРС и расстояний старт- и стоп-кодона от 5' и 3' начала и конца последовательностей, которые не учтены в разработанной модели векторизации фрагментов ОРС.

Разработан и успешно проверен на большом наборе экспериментальных данных эффективный вычислительный подход к определению ОРС кодирующих молекул РНК на основе алгоритмов векторизации и случайного леса, обученного на ложных ОРС не кодирующих РНК и истинных ОРС кодирующих РНК. Определён набор наиболее информативных признаков фрагментов нуклеотидных последовательностей молекул – это признаки модели Вао и два параметра оценки длины ОРС. Отметим, что наши результаты позволяют предположить о невысокой значимости последовательностей Козак при определении координат ОРС и параметров расстояний старт- и стоп-кодона от 5' и 3' начала и конца последовательностей. Точность определения ОРС в рассмотренных молекулах РНК из баз данных NCBI RefSeq и Ensembl составляет 98,14%.

Предложенный вычислительный подход может быть использован в прикладных биомедицинских исследованиях, нацеленных на совершенствование дифференциальной диагностики заболеваний человека генетической природы (включая онкологические заболевания) и для улучшения качества построения прогностических моделей течения подобных заболеваний (включая прогнозирование ответа пациента на лечебную терапию).

СПИСОК ЛИТЕРАТУРЫ

- Mardis, E. R. DNA sequencing technologies: 2006-2016 / E. R. Mardis // *Nat. Protoc.* – 2017. – Vol. 12, № 2. – P. 213–218.
- Reuter, J. A. High-throughput sequencing technologies / J. A. Reuter, D. V. Spacek, M. P. Snyder // *Mol. Cell.* – 2015. – Vol. 58, № 4. – P. 586–597.
- Sayers, E. W. Database resources of the National Center for Biotechnology Information / E. W. Sayers, R. Agarwala, E. E. Bolton [et al.] // *Nucleic Acids Res.* – 2019. – Vol. 47, № D1. – P. D23–D28.
- Kang, Y. J. CPC2: a fast and accurate coding potential calculator based on sequence intrinsic features / Y. J. Kang, D. C. Yang, L. Kong [et al.] // *Nucleic Acids Res.* – 2017. – Vol. 45, № W1. – P. W12–W16.
- Разработка алгоритмов и программных средств классификации кодирующих и не кодирующих нуклеотидных последовательностей / В. П. Закирова [и др.] // *Информатика.* – 2019. – Т. 16, № 2. – С. 111–120.
- Breiman, L. Random forest / L. Breiman // *Machine Learning.* – 2001. – Vol. 45, № 1. – P. 5–32.
- Bao, J. An improved alignment-free model for DNA sequence similarity metric / J. Bao, R. Yuan, Z. Bao // *BMC Bioinformatics.* – 2014. – Vol. 15, № 321. – P. 1–15.
- Comparative analyses between retained introns and constitutively spliced introns in *arabidopsis thaliana* using random forest and support vector machine / R. Mao [et al.] // *PLoS One.* – 2014. – Vol. 9, № 8. – P. 1–12.
- Интеллектуальный анализ данных / Н. Н. Яцков – Минск: БГУ, 2014. – 151 с.
- Kozak, M. Point mutations define a sequence flanking the AUG initiator codon that modulates translation by eukaryotic ribosomes / M. Kozak // *Cell.* – 1986. – Vol. 44, № 2. – P. 283–292.

Секция «Первый шаг в науку»

Section «First step»

АЛГОРИТМЫ РАСПРЕДЕЛЕНИЯ ЗАДАНИЙ СОТРУДНИКОВ ОТДЕЛА

Бачило Е. Г.

Факультет информационных технологий и управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: e_bachilo@mail.ru

Объектами исследования являются методы и алгоритмы распределения заданий среди сотрудников отдела. В данной статье рассматриваются существующие методы делегирования заданий, ролей и ответственностей сотрудникам, а также виды календарного планирования. Цель состоит в том, чтобы проанализировать существующие методы делегирования и виды календарного планирования для разработки на их основе улучшенного алгоритма распределения заданий и получения календарного плана.

ВВЕДЕНИЕ

Делегирование – это такая организация работы, при которой руководитель распределяет между своими подчинёнными конкретные задания; это также передача подчинённому задачи или действия из сферы действий руководителя вместе с необходимыми полномочиями. Со временем повышается ценность только одного ресурса – человеческого. Ценность сотрудников может расти в зависимости от того, как ими руководят и как используют их способности. Делегирование полномочий побуждает сотрудников стремиться достичь большего, добиваться весомых результатов и пользы для компании [1].

I. МЕТОДЫ ДЕЛЕГИРОВАНИЯ

Рассмотрим пять основных методов делегирования заданий сотрудникам отдела.

Матрица Эйзенхауэра относится скорее не к методам делегирования, а к методам классификации задач для делегирования. Однако, на основании такой классификации задач уже можно сказать, как и что лучше делегировать. Матрица Эйзенхауэра, представленная на рисунке 1, представляет собой четыре квадранта, основанием которых служат две оси – ось важности (по вертикали) и ось срочности (по горизонтали). Смысл матрицы Эйзенхауэра заключается в том, чтобы научиться грамотно распределять все дела, отличать важное от срочного, несрочное от наименее важного, а также по максимуму сокращать время на занятие любыми делами, выполнение которых не даёт никаких существенных результатов [2].

	СРОЧНО	НЕ СРОЧНО
ВАЖНО	A	B
НЕ ВАЖНО	C	D

Рис. 1 – Матрица Эйзенхауэра

Упрощённая версия алгоритма делегирования состоит из следующих шагов:

Шаг №1. Руководитель делает всё сам;

Шаг №2. Руководитель проводит предварительную подготовку сотрудника;

Шаг №3. Сотрудник выполняет знакомые действия частично самостоятельно;

Шаг №4. Руководитель разрешает возникающие проблемы, по итогам разбора дополняет правила и примеры;

Шаг №5. Руководитель оценивает: может ли в дальнейшем сотрудник выполнять этот этап самостоятельно? Если «Нет», то перейти на «Шаг №2» или «Шаг №3». Если «Да», то перейти на «Шаг №6»;

Шаг №6. Сотрудник выполняет этап самостоятельно;

Шаг №7. Плановый контроль со стороны руководителя и сбор информации по возникающим проблемам и отклонениям.

Режиссёрская версия алгоритма делегирования представляет собой расширенную версию предыдущего метода с добавлением понятия «класс событий» [3].

Пятиступенчатый метод применяется в случаях сложного и нового задания и имеет следующие ступени:

1. Подготовить сотрудника;
2. Объяснить задачу;
3. Показать, как делать работу;
4. Доверить сотруднику выполнение работы под наблюдением и корректировать его;
5. Передать сотруднику работу целиком и контролировать его.

Схема пяти шагов применяется, если сотрудник новый, т.е. неизвестны его способности. Он состоит из следующих пяти шагов:

1. Смотри, как делаю я;
2. Спроси, как сделать, и делай под моим контролем;
3. Предложи, как сделать, и делай под моим контролем;
4. Предложи план, сделай, отчитайся;
5. Сделай, отчитайся.

Помимо методов делегирования заданий существуют методы делегирования ролей и ответственностей, которые также необходимо учитывать. Рассмотрим некоторые из них.

Методологию RACI можно описать с помощью таблицы (таблица 1).

Таблица 1 – Методология RACI

«R» Исполнитель (Responsible)	Лежит ответственность за выполнение поставленной задачи
«A» Утверждающий (Accountable)	Перед ним производится отчет о полученном результате, имеются полномочия, как принимать, так и отвергать предложения, накладывая на них вето
«C» Консультант (Consulted)	Консультация и согласование принимаемых решений, характеризуется двусторонней связью между подразделениями
«I» Информированный (Informed)	Поступает конечная информация о проделанной работе, характеризуется односторонней связью

Методология RAEI также представлена таблицей (таблица 2). Его автор Ицхак Адизес выделяет четыре управленческие роли.

Таблица 2 – Методология RAEI

«P» Производитель результатов	Получение результатов
«A» Администратор	Слежение за соблюдением установленных правил, организация процесса выполнения задания, поддержание порядка
«E» Предприниматель	Активное создание нового и развитие существующего
«I» Интегратор	Умение формировать общие цели и ценности по ключевым вопросам

Методология «Функционер» и «Развитийщик» основана на двух классификациях. Различают следующие типы исполнителей: «генератор идей», «развитийщик»; «функционер», «человек процедур»; смешанный тип; универсал. Среди задач различают следующие задачи: «функциональная»; инновационная, «новая»; задача, требующая модернизации. Руководителю важно достичь баланса между сотрудниками «развитийщиками» и «функционерами».

II. ВИДЫ КАЛЕНДАРНОГО ПЛАНИРОВАНИЯ

Задачами календарного планирования являются составление и корректировка расписа-

ния, в котором работы, выполненные различными организациями, увязываются во времени между собой и с возможностями их обеспечения различными видами материально-технических и трудовых ресурсов [4].

Календарный план по *диаграммам Ганта* определяет продолжительность работ, даты (сроки) их начала и окончания, резервы времени и величины ресурсов, необходимых для выполнения работ и проекта в целом.

Планирование по срокам является не отдельным методом календарного планирования, а частью планирования по диаграммам Ганта.

Объемно-календарный план строится путем последовательного нанесения на график соответствующих упорядоченных работ по отдельным операциям с фиксацией их длительности. Данное планирование не требует построения сетевого графика, однако, для него необходима интегрированная система ERP, показывающая взаимосвязь работ [5].

ЗАКЛЮЧЕНИЕ

На основании рассмотренного материала предполагается построить алгоритм делегирования, который будет являться синтезом метода делегирования заданий и метода делегирования ролей и ответственностей. С помощью диаграмм Ганта будет построен календарный план как наиболее наглядный и информативный вид календарного планирования. После чего с помощью теории массового обслуживания будет сформулирована задача для имитации. Эффективность предлагаемого автором алгоритма предполагается проверить с помощью имитационного моделирования с использованием системы моделирования дискретных процессов GPSS World.

СПИСОК ЛИТЕРАТУРЫ

- Трейси, Б. Делегирование и управление / Б. Трейси // Издательство: Манн, Иванов и Фербер, 2014. — 144 с.
- Гений, А. Высокоэффективный тайм-менеджмент по Матрице Эйзенхауэра / А. Гений // Издательство: АСТ, 2016. — 320 с.
- Пошаговый алгоритм делегирования задач подчиненным от А до Я [Электронный ресурс] / Открытая студия. — Москва, 2016. — Режим доступа: <http://openstud.ru/blog/for-businessmen/delegation-algorithm/> — Дата доступа: 26.06.2019.
- Киселев Э. В. Календарное планирование в управлении качеством: Учебное пособие / Э. В. Киселев // Издательство: РГТА, 2003. — 71 с.
- Методы планирования материальных потребностей [Электронный ресурс] / LearnLogistic. — Екатеринбург, 2009. — Режим доступа: <http://learnlogistic.ru/metody-planirovaniya-materialnyh-potrebnostej/> — Дата доступа: 13.06.2019.

АЛГОРИТМЫ АНАЛИЗА АВТОМАТИЗАЦИИ БИЗНЕС-ПРОЦЕССОВ

Бибик О. Н., Белый А. И., Гуринович А. Б.

Факультет информационных технологий и управления, Кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: volha.bibik@gmail.com, alex.bely@gmail.com, gurinovich@bsuir.by

Автоматизация — приоритетное и актуальное направление в сфере теории и практики управления бизнес-процессами. Ее цель — повышение социально-экономической эффективности предприятия путем целенаправленного внедрения программно-аппаратных систем, повышающих скорость и точность реализации бизнес-процессов, минимизирующих при этом ошибочные действия и решения, возникающие под влиянием случайных факторов.

I. ОПРЕДЕЛЕНИЕ БИЗНЕС-ПРОЦЕССОВ

Существующая практика построения систем управления включает в себя несколько подходов. Основные методы организации систем управления: системы, построенные на управлении функциями и управлении бизнес-процессами организации.

Бизнес-процесс — последовательность действий (подпроцессов), направленная на получение заданного результата, ценного для организации[1].

Системы управления, построенные на принципах управления функциями, представляют собой иерархическую пирамидальную структуру подразделений, сгруппированных по выполняемым функциям. Под функциональным подразделением можно понимать группу экспертов в данной функциональной области. В организациях, построенных по данному принципу, управление осуществляется на административно-командных принципах.

Другим подходом построения систем управления является управление потоками работ или процессами, составляющими деятельность предприятия. Процессное подразделение включает в себя координатора — владельца процесса и исполнителей из различных функциональных областей, сгруппированных по принципу единства результата бизнес-процесса. Подобные системы часто называют «горизонтальные», подразумевая под «вертикальным» управлением иерархию функциональных подразделений и руководителей в стандартной системе управления, построенной по функциональному принципу.

Процессный подход к управлению предприятием можно считать базовым при интегрировании предприятия. При этом подходе объектом управления выступает определенная деятельность на предприятии — процесс, который можно определить как ряд взаимосвязанных видов деятельности, характеризующихся потреблением ресурсов (вход процесса) и дающих определенный результат (выход процесса). Процессы проходят через все подразделения, вовлекая все

службы предприятия, и ориентируются на конечный результат — увеличение стоимости бизнеса. Управляя процессами, имеющими свои цели, можно добиться высокой эффективной деятельности с помощью хорошо налаженных горизонтальных связей в вертикальной структуре управления предприятием.

Концептуальная схема управления процессом представлена на Рис. 1.

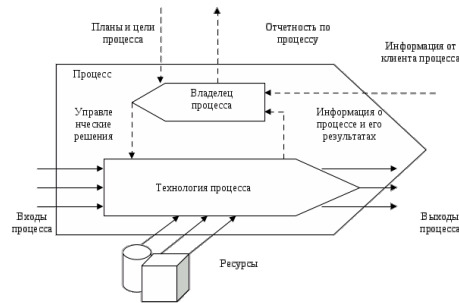


Рис. 1 – Концептуальная схема управления процессом

II. ОПИСАНИЕ БИЗНЕС-ПРОЦЕССОВ

Чтобы понять, как функционирует система и сохранить накопленный опыт, сначала записывают, как процесс реально функционирует сейчас. Нужно помнить, что целью описания является выявление связей между предпринимаемыми действиями, а не фиксирование мельчайших подробностей. Поэтому описание бизнес-процессов рекомендуется стандартизовать, используя стандартные формы и карты процесса.

Описывать бизнес-процесс рекомендуется методом последовательных приближений. После завершения описания бизнес-процесса рекомендуется провести работу по его улучшению (повторить цикл действий до получения приемлемого результата).

В описании бизнес-процесса можно выделить следующие разделы:

1. Стандартные формы бизнес-процесса;
2. Карта бизнес-процесса;
3. Маршруты бизнес-процесса;
4. Блок-схемы бизнес-процесса;

5. Развернутое описание бизнес-процесса;
6. Документирование бизнес-процесса.

III. АНАЛИЗ БИЗНЕС-ПРОЦЕССОВ

Анализ — это деятельность, предпринимаемая для установления пригодности, адекватности и результативности рассматриваемого объекта для достижения поставленных целей[2].

Условно все виды анализа бизнес-процессов можно разбить на два вида: качественные и количественные. Методы, применяемые для анализа процесса с точки зрения его составных частей, элементов и способа реализации, относятся к качественным. Методы, используемые для оценки процесса в цифровом выражении (скорость выполнения, количество выпускаемой продукции, стоимость реализации и т.д.) при измерении каких-либо показателей эффективности его выполнения, — к количественным.

Анализ бизнес-процессов следует понимать в широком смысле: в него включается не только работа с графическими схемами, но и анализ всей доступной информации по процессам, измерения их показателей, сравнительный анализ и т. д. Классификация видов анализа процессов[3] приводится на рисунке 2.

Можно выделить несколько методик субъективной оценки процессов. Во многом такие методики были разработаны в трудах основоположников и последователей методологии реинжиниринга бизнес-процессов, например у Хаммера и Чампи, Робсона и Уллаха и т. д. Кроме того, для качественного анализа процессов могут

быть использованы общеизвестные методы анализа: SWOT-анализ, анализ при помощи Бостонской матрицы и другие.

Методы количественного анализа процессов более подробно разработаны в мировой практике. Большая их часть основана на сборе, обработке и анализе статистической информации о процессах. Фактически методы статистического анализа процессов разрабатывались как инструменты, используемые при внедрении систем менеджмента качества. В настоящее время широкое распространение получили такие методы количественного анализа, как имитационное моделирование процессов и ABC-анализ процессов (операционный анализ затрат).

IV. Выводы

Важнейшими этапами при проектировании автоматизированных систем являются описание и анализ бизнес-процессов предприятия, последующая формализация требований к разрабатываемой системе на основании проведенного анализа. Данные этапы относятся к предпроектному исследованию и составляют фундамент для последующих этапов жизненного цикла автоматизированной системы.

- 1 Марка, Д. Методология структурного анализа и проектирования / Д. Марка, К. МакГоуэн // Москва. – 2003. – С. 240.
- 2 Елиферов, В. Г. Бизнес-процессы: регламентация и управление / В. Г. Елиферов, В. В. Репин // ИНФРА-М. – 2005. – С. 319.
- 3 Шеер, А. Бизнес-процессы: основные понятия, теории, методы / А. Шеер // Москва. – 1999. – С. 368.

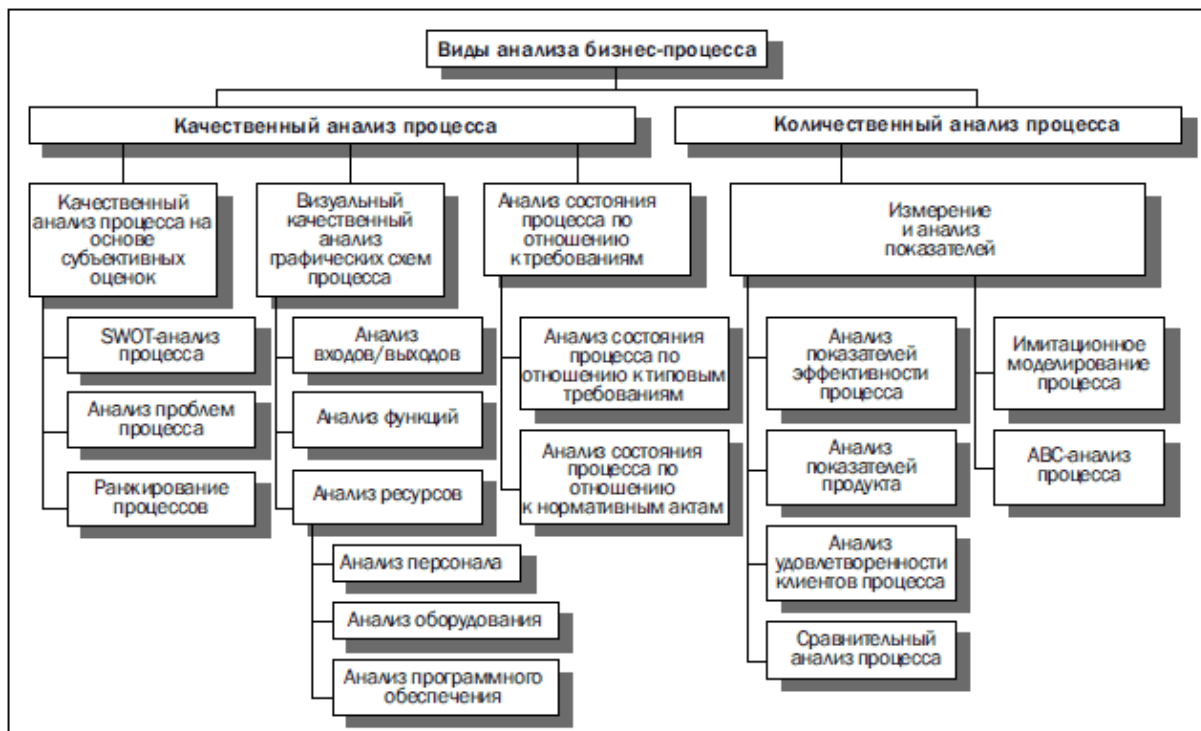


Рис. 2 – Классификация видов анализа бизнес-процессов

НОВЫЕ ПОДХОДЫ ОПТИМИЗАЦИИ ЗАПРОСОВ В БАЗАХ ДАННЫХ

Гобрик О.Д., Гуринович А.Б.

Кафедра информационных технологий автоматизированных систем, Кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь

E-mail: oleg.gobrik@gmail.com, gurinovich@bsuir.by

В работе исследуется проблема обеспечения эффективности функционирования систем управления базами данных (СУБД) с помощью оптимизации запросов. Показываются наиболее распространенные виды параллельных реляционных СУБД. Причиной наибольших ошибок оптимизатора запросов является оценка стоимости запроса.

ВВЕДЕНИЕ

Рост многообразия моделей данных и способов их представления на физических носителях (т.е. типов БД), и многообразия средств взаимодействия с БД (т.е. СУБД) приводит к появлению новых методов решения оптимизации запросов. Отметим, что ядром БД является модель данных, т.е. совокупность структур данных и операций их обработки. Кроме того, СУБД по своему назначению делятся на операционные и предназначенные для работы с хранилищами данных, содержащими очень большой объем информации, подготовка представления которой занимает значительное время. Несмотря на растущую сложность реализации и большое обилие алгоритмов, обеспечивающих большую производительность, по-прежнему существует некоторое число проблем, связанных с производительностью запросов. В исследовании планируется описать подходы оптимизации запросов баз данных.

I. ПОДХОДЫ ОПТИМИЗАЦИИ ЗАПРОСОВ

На данный момент существуют 3 направления исследований.

1. Построение параллельных вычислений на основе использования графических процессоров (GPU). В состав обычного CPU входят от 1 до 32 ядер, а в состав GPU – несколько тысяч простых процессоров, например, GeForce RTX 2080 обладает 2944 ядрами CUDA. Под «простыми» понимаются процессоры, не содержащие компонент проверки условий, кэш и т.д. Имея функцию, выполняющую некоторую работу, разработчик назначает ее исполнение на тот или иной GPU. Модель программирования GPU описывает 6 типов областей памяти, различных по скорости доступа, прав на запись-чтение, объему. Наличие большого числа процессоров и быстрой памяти обеспечивает эффективность выборки данных со сложным условием, так как каждый поток исполняется независимо от других, оперируя только с малой частью обрабатываемых данных. В результате существенно снижается различие между временем загрузки данных с медленно-

го носителя в оперативную память и временем, затрачиваемым на исполнения запроса и возвращения результата[3].

2. Разработка моделей и методов обработки и хранения больших данных. Под термином «большие данные» понимают электронные данные, характеризующиеся большим объемом, разнообразием и скоростью, с которой структурированные и неструктурированные данные поступают по сетям передачи в процессоры и хранилища, а также наличием эффективных процессов переработки данных в требуемую информацию. Именно актуальность разработки средств эффективной обработки больших данных привела к появлению концепции NoSQL[4]. Ее суть состоит в проектировании архитектуры, обладающей свойством адаптации к возрастающим объемам данных. При этом сохранение эффективности процессов обработки данных обеспечивается за счет высокой пропускной способности и потенциально неограниченного горизонтального масштабирования.

Также для обработки данных применяют облачные технологии. Для оптимизации в облаке используется 2 способа. Первый способ реализует параллельную обработку на уровне запросов. В этом случае алгебраическое выражение запроса разбивается на подвыражения, которые могут выполняться одновременно на разных процессорах или ядрах одного многоядерного процессора. Второй способ основан на параллельной реализации операций, которые составляют запрос. Далее отдельно рассмотрен способ оптимизации путем использования машинного обучения.

II. ПРИМЕНЕНИЕ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ

Для оптимизации запросов в реляционных СУБД применяется стоимостной оптимизатор. При стоимостной оптимизации оптимальность плана запроса определяется значением, которое именуется стоимостью плана. Данное значение рассчитывается как совокупная стоимость каждой операции в запросе и определяется в зависимости от планируемого числа записей, которые

будут извлечены из базы (значение cardinality). Для решения проблемы производительности при взаимодействии с данными, которые имеют зависимость между собой, применяются алгоритмы, позволяющие подбирать наилучший план. Данная реализация называется "Adaptive Query Optimization". Она позволяет выбирать наиболее производительный план при повторном выполнении запросов одного типа, обучаясь после каждого запроса[5].

Таким образом, решается проблема актуальности статистических данных для осуществления расчётов системой машинного обучения. Теперь, произведя всю цепочку описанных действий, можно получить оценку стоимости каждой вершины плана и выбрать наиболее оптимальную с точки зрения объёмов затрачиваемых на выполнение последних ресурсов. Технологии машинного обучения позволяют автоматизировать и оптимизировать работу с формализованными данными, собирая при этом актуальную статистику и самосовершенствуясь.

Не менее важным вопросом, является предоставление наиболее точных исходных данных. Оптимального размещения файлов реляционной БД по узлам компьютерной сети требует ряда информационных массивов исходных данных, значительная часть которых может быть получена лишь в усредненном или пониженном виде. Это такие характеристики, как интенсивности запросов, время пересылки и обработки запросов, объёмы запросов и ответов на запросы. Точность собранной статической информации будет решающим образом влиять на конечный результат реализации выбранной математической модели и, следовательно, на производительность системы, работающей с реляционной БД[6].

В PostgreSQL существует реализация адаптивной оптимизации. Здесь используется аналогичный метод подбора значения cardinality для схожих запросов. В качестве алгоритма используется модифицированный метод KNN (К-ближайших соседей), способный обрабатывать ситуацию изменения данных в таблице.

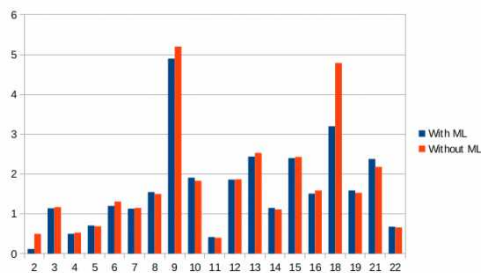


Рис. 1 – Диаграмма скорости выполнения запросов

Хотя данная реализация способна справляться с различными изменениями в данных, все же рекомендуется реинициализировать алгоритм после внесения обширных изменений в данные (в контексте KNN это означает удаление созданных объектов с информацией о cardinality). Данный подход ускоряет время работы СУБД. Для тестирования использовался бенчмарк TPC-H. На одном из типов запросов бенчмарка ускорение составляет 30-45% (2-ой и 18-ый запросы).

Тем не менее, данная реализация имеет несколько недостатков. В качестве признаков здесь используются значения селективности (процент кортежей, удовлетворяющих конкретному условию), извлекаемые из статистики, что может работать не совсем корректно, так как неэквивалентные условия в запросе могут превращаться в эквивалентные по значению результаты предсказания. Так же существует проблема с работой данного плагина на репликах, поскольку параметры алгоритма хранятся в таблице в базе.

В условиях стремительно растущей потребности в эффективных способах хранения и манипуляций с большими объёмами данных поиск новых решений является необходимым.

III. ЗАКЛЮЧЕНИЕ

Существующие подходы к проблеме оптимизации запросов обладают рядом недостатков, для решения которых предлагается 3 метода: построение параллельных вычислений на основе использования графических процессоров, разработка моделей и методов обработки и хранения больших данных, и применение алгоритмов машинного обучения. Установлено, что использование машинного обучения ведет к улучшению работы планировщика и, соответственно, ускорению работы СУБД, а также отмечены проблемы в алгоритме оценки выборочности. В дальнейшем, при использовании более сложных алгоритмов машинного обучения, возможно ускорить обработку запросов и работу СУБД в целом.

СПИСОК ЛИТЕРАТУРЫ

1. А. Ю. Васильев, Работа с PostgreSQL: настройка и масштабирование, 2015 - 203 с.
2. <http://dspace.nbuv.gov.ua/bitstream/handle/123456789/1638/12-Shchetinin.pdf?sequence=1>
3. <https://www.academia.edu/8903054/>
4. Медетов А.А. Термин Big Data и способы его применения // Молодой ученый, 2016.207-210
5. http://www.libeldoc.bsuir.by/bitstream/123456789/32182/1/Sharayev_optimizatsiya.pdf
6. Маркин С.Д., Головинов А.О. Интеграция методов машинного обучения в планировщик SQL-запросов // Научное сообщество студентов XXI столетия. Технические науки: сб. ст. по мат. LIII междунар. студ. науч.-практ. конф. № 5(52).

МОДЕЛЬ МНОГОМЕРНОГО ПРЕДСТАВЛЕНИЯ ДАННЫХ В ХРАНИЛИЩАХ ДАННЫХ

Геврасёва И. П.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: irina_gevraseva@mail.ru

Рассматривается многомерная модель данных, ее основные понятия и схемы, используемые при проектировании хранилища данных.

ВВЕДЕНИЕ

Типичная структура хранилища данных существенно отличается от структуры обычной операционной базы данных. Как правило, в базах данных используется реляционная модель данных, но для хранилищ данных более эффективным является применение многомерной, или пространственной, модели данных, основанной на использовании таблиц фактов и измерений.

I. ОСОВНЫЕ ПОНЯТИЯ ДЛЯ МНОГОМЕРНОЙ МОДЕЛИ ДАННЫХ

Многомерное моделирование - это метод логического проектирования данных в стандартной интуитивно понятной структуре для высокопроизводительного доступа, состоящей из одной таблицы с многокомпонентным ключом, называемой таблицей фактов, и набора меньших таблиц, называемых таблицами измерений.

По сравнению с реляционной моделью многомерная организация данных обладает более высокой наглядностью и информативностью.

Основными понятиями для многомерных моделей данных являются:

- *Агрегируемость* данных означает рассмотрение информации на различных уровнях ее обобщения;
- *Историчность* данных предполагает обеспечение высокого уровня статичности данных и их взаимосвязей, а также обязательность привязки данных ко времени;
- *Прогнозируемость* данных подразумевает задание функций прогнозирования и применение их к различным временным интервалам.

Многомерность модели данных означает не многомерность визуализации цифровых данных, а многомерное логическое представление структуры информации при описании и в операциях манипулирования данными.

Основным достоинством многомерной модели данных является удобство и эффективность аналитической обработки больших объемов данных, связанных со временем.

Недостатком многомерной модели данных является ее громоздкость для простейших задач обычной оперативной обработки информации.

II. СТРУКТУРА МНОГОМЕРНОЙ МОДЕЛИ ДАННЫХ

Данные в многомерной модели организованы не по третьей нормальной форме, а с использованием двух типов таблиц – таблиц фактов и измерений:

Таблицы Фактов. Таблица фактов – основная таблица хранилища данных. Она содержит сведения об объектах или событиях, совокупность которых будет в дальнейшем анализироваться.

Таблица фактов, как правило, содержит уникальный составной ключ, объединяющий первичные ключи таблиц измерений. Чаще всего это целочисленные значения либо значения типа «дата/время». Помимо этого, таблица фактов содержит одно или несколько числовых полей-фактов, на основании которых в дальнейшем будут получены агрегатные данные.

Для многомерного анализа пригодны таблицы фактов, содержащие как можно более подробные данные, т.е. данные без агрегации. Обработка, хранящихся в таблицах фактов данных, происходит в дальнейшем.

Таблицы Измерений. Таблицы измерений содержат неизменяемые или редко изменяемые данные. Эти таблицы содержат описательного характера данные, относительно которых будут анализироваться собранные в таблице фактов данные. Скорость роста таблиц измерений должна быть незначительной по сравнению со скоростью роста таблицы фактов.

В зависимости от связей между таблицами фактов и таблицами измерений существует три схемы организации многомерной базы данных – «звезд», «снежинка» и «галактика».

III. СХЕМЫ ОРГАНИЗАЦИИ МНОГОМЕРНОЙ МОДЕЛИ ДАННЫХ

Схема «Звезда». На рисунке 1 представлена реализация схемы звезда.

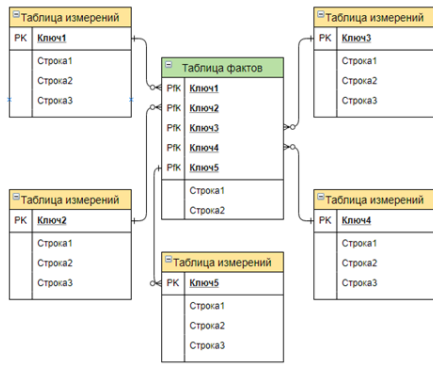


Рис. 1 – Схема «звезда»

В схеме «звезда» таблица фактов, содержащая данные для анализа, является центральной. От таблицы фактов идут связи к таблицам измерений, содержащим описательную информацию. Таблица фактов и таблицы измерений связаны идентифицирующими связями, при этом первичные ключи таблицы измерений мигрируют в таблицу фактов в качестве внешних ключей. Первичный ключ таблицы фактов целиком состоит из первичных ключей всех таблиц измерений. Как правило, таблицы измерений денормализованы и не имеют связей с другими таблицами измерений.

Преимуществом рассматриваемой схемы является то, что благодаря денормализации таблиц измерений упрощается восприятие структуры данных и формулировка запросов, уменьшается количество операций соединения таблиц при обработке запроса, что, в свою очередь, повышает производительность обработки данных.

Недостатком данной схемы в связи с денормализацией таблиц измерений является появление избыточности данных, которое влечет за собой увеличение объема памяти, необходимой для хранения данных.

Схема «Снежинка». Реализация схемы «снежинка» представлена на рисунке 2.

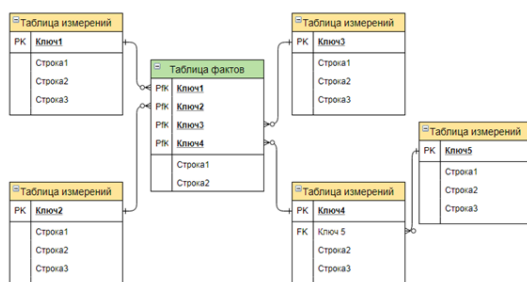


Рис. 2 – Схема «снежинка»

Схема «снежинка» имеет схожие отличительные черты со схемой «звезда». В схеме «снежинка» так же, как и в схеме «звезда», центральной таблицей является таблица фактов, имеющая связи с описательными таблицами измерений. Отличительной чертой рассматриваемой схемы является то, что таблицы измерений имеют более нормализованную структуру и могут быть связаны с другими таблицами измерений, что не характерно для схемы «звезда».

Схема «Галактика». Это более сложная структура, имеющая несколько таблиц фактов, которые могут использовать общую таблицу измерений. В целом ее можно представить, как усложненную версию схемы «звезда». Соответственно, преимущества и недостатки данной схемы такие же, как и у схемы «звезда». Однако более сложная структура данной схемы влечет за собой увеличения сложности запросов для обработки данных.

Нормализация таблиц измерений позволяет минимизировать избыточность данных и более эффективно выполнять запросы, связанные со структурой значений измерений, что является существенным преимуществом схемы ««снежинка»». При этом же более нормализованная структура таблиц измерений влечет за собой потерю производительности при выполнении запросов.

Для организации хранилища данных в основном используют схему «звезда», так как производительность является более приоритетным преимуществом.

СПИСОК ЛИТЕРАТУРЫ

1. Kimball, R. The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, Third Edition / R. Kimball, M. Ross // Wiley Publishing, Inc. – 2013. – С. 564.
2. Inmon, W. H. Building the Data Warehouse, Fourth Edition / W. H. Inmon // Wiley Publishing, Inc. – 2005. – С. 576.
3. Han, J. Data mining Concepts and Techniques, Third edition / J. Han, M. Kamber, J. Pei // Morgan Kaufmann, Inc. – 2011. – С.744

ОБЗОР ПОДХОДОВ К РЕШЕНИЮ ЗАДАЧИ ПЕРЕНОСА СТИЛЯ ТЕКСТА С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

Евтушенко Е. Ю.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: lizasixsixsix@tut.by

В данной работе даётся обзор новейших подходов к решению задачи изменения стиля текста с оригинального на целевой с сохранением его смыслового содержания с помощью искусственных нейронных сетей.

ВВЕДЕНИЕ

Задача переноса стиля в различных областях творческой деятельности человека – в текстах, в изображениях, в музыке – сегодня вызывает большой интерес, и не только благодаря практической применимости, но и как показатель способности нейронных сетей творить.

Эта задача имеет дополнительную сложность: отсутствие чётких метрик для оценки качества сгенерированного контента и его соответствия желаемому результату. Для преодоления этой сложности были разработаны различные архитектуры нейронных сетей.

Работа [1] показала потрясающие результаты в области переноса стиля изображений – комбинация двух входных изображений – одного как источника содержания и другого как источника стиля – преобразовывалась сетью в убедительный результат. На основе этой сети в 2016 году было создано мобильное приложение для обработки изображений Prisma [2].

К сожалению, область естественного языка пока не может похвастаться столь же впечатляющими результатами. Это связано с гораздо более сложной семантикой текстов в сравнении с изображениями – генерация осмысленного текста является трудной задачей.

Кроме того, понятие стиля применительно к текстам – понятие, которое может трактоваться широко – как настроение текста, или же как возраст и пол автора.

Рассматриваемые далее работы решают задачу переноса стиля для текстов на английском языке.

I. ОТДЕЛЕНИЕ СОДЕРЖИМОГО ОТ СТИЛЯ

До недавнего времени решения задачи переноса стиля текста пытались копировать подход, применяемый в упомянутом выше примере переноса стиля изображений в том смысле, что содержимое и стиль рассматривались как компоненты, которые необходимо разделить, чтобы затем собрать из их новых сочетаний новый текст. В данной задаче единицей обычно является одно предложение.

В этом случае входное предложение преобразуется моделью в два вектора, один из которых

(z) принадлежит пространству скрытого представления содержимого (смысла), а другой (s) – пространству скрытого представления стиля.

Скрытое представление содержания z должно быть одинаковым для текстов, отличающихся лишь стилем. Это говорит о том, что удалось очистить содержимое от стиливой информации.

Вектор представления стиля s , в отличие от случая работы с изображениями, не может быть составлен на основании только одного предложения, а извлекается из целых корпусов текстов, и задача скорее определяется как перенос стиля из одного домена в другой. Это требует некоторых заранее подготовленных стилистически однородных наборов данных.

В работе [3] рассматривается модель, построенная на основе вариационного автокодировщика. Задача переноса стиля здесь понимается как изменение отрицательных отзывов посетителей различных заведений на положительные.

Пространство скрытого представления предложений h разделяется на два пространства s и c – для стиля и содержания, соответственно. Для каждого из этих пространств применяется своя пара функций потерь.

Одна из каждой пары функций обеспечивает то, что соответствующее пространство содержит необходимую информацию. Другая функция в паре, наоборот, стремится очистить каждое пространство от лишней информации, чтобы нельзя было предсказать стиль по вектору содержимого и наоборот.

Что интересно в этой модели – содержимое тоже может быть предсказано. Для этого используется аппроксимация информации о содержании текста BOW-атрибутами (BOW – bag-of-words – «мешок слов»). В данном случае BOW-атрибут входной последовательности – это вектор, каждый элемент которого показывает вероятность того, что данное слово (исключая заранее определённые эмоционально окрашенные слова) встретится в предложении.

В работе [4] к той же задаче применяется иной подход, первоначально применявшийся в машинном переводе – специализированный декодировщик.

Здесь модель похожа на автокодировщик, но при этом кодировщик e принимает на вход как предложения из домена оригинального стиля S , так и из целевого домена T и пытается научиться создавать скрытое представление без стилистической информации. Декодировщик d преобразует это скрытое представление в предложение в целевом домене T (независимо от домена его происхождения). Таким образом, модель обучается создавать только одно скрытое представление z – для содержимого.

Для оценки качества переноса используется популярная сейчас архитектура GAN (Generative Adversarial Network – генеративно-сопоставительная сеть), впервые описанная в работе [5]. Кодировщик и декодировщик представляют собой генератор $G = d \circ e$, а дискриминатор D – это отдельный компонент, который предназначен для различения перенесённых предложений \hat{t} и оригинальных t , взятых непосредственно из T . Цель обучения генератора при этом – «обман» дискриминатора.

II. НОВЫЕ ПОДХОДЫ – БЕЗ ОТДЕЛЕНИЯ СОДЕРЖАНИЯ ОТ СТИЛЯ

Последние работы отходят от идеи явно-го разделения стиля и содержимого для решения задачи переноса стиля текста. Тому есть несколько причин: во-первых, такого разделения невозможно полностью достичь; во-вторых, скрытое представление может оказаться неспособным полностью сохранить содержание.

Работа [6] показала, что отделение вообще необязательно.

В данной работе авторы перезаписывают настроение, пол и возрастную группу автора отзыва, то есть, сразу несколько атрибутов стиля.

Рассматривается тренировочный набор данных из n предложений $x^i \in X$ в паре со значениями атрибутов y^i . $y \in Y$ – это набор из m значений атрибутов $y = (y_1, \dots, y_m)$. Каждое значение атрибута y_k – это дискретное значение в множестве Y_k возможных значений атрибута k , например, $Y_k = \{\text{bad, neutral, good}\}$ для y_k , обозначающего общую оценку в отзыве. Задачей является обучение модели $F : X \times Y \rightarrow X$, которая отображает каждую пару (x, \hat{y}) входного предложения x (чьим настоящим набором атрибутов является y) и нового набора m значений атрибутов \hat{y} на новое предложение \hat{x} , которое обладает обозначенными атрибутами \hat{y} , при условии как можно большего сохранения первоначального содержания x .

Рассматриваемая архитектура производит отображение с помощью автокодировщика, который сперва кодирует x в скрытое представление $z = e(x)$, затем декодирует (z, \hat{y}) в $\hat{x} = d(z, \hat{y})$.

Скрытое представление z может быть очень запутанным в смысле стиля и содержания, однако требованием является только «перезапись»

декодером первоначальной информации об атрибутах стиля желаемыми атрибутами.

Для оценки переноса стиля использовалась техника обратного перевода (back-translation): берутся входные данные (x, y) и кодируются x в z , а затем декодируются, используя другой набор атрибутов \hat{y} , возвращая реконструированное \hat{x} . Затем \hat{x} используется как входные данные кодировщика и декодируются с использованием первоначальных атрибутов y , пытаясь, в идеале, получить первоначальное x . Таким образом, модель обучается отображать (\hat{x}, y) в x .

Таким образом, в этой модели стиль перезаписывается декодером «поверх» оригинального, и нет нужды отделять стиль от содержимого.

Механизм внимания (взвешенный вектор важности элементов последовательности), впервые описанный в работе [7], не мог использоваться вместе с концепцией скрытого представления из-за того, что длина вектора скрытого представления фиксирована. Отказ от этого подхода позволил применить механизм внимания в работе [8].

Модель, получившая название Style Transformer, имеет генеративно-сопоставительную архитектуру с двумя функциями потерь – одна представляет собой рассмотренный выше «обратный перевод», а другая реализована как дискриминатор.

Здесь, как и в первых двух работах, перезаписывается только один атрибут – настроение отзыва.

ЗАКЛЮЧЕНИЕ

Перенос стиля текста – актуальная, хотя и сложная задача, в которой с каждым годом применяются новые подходы из разных областей машинного обучения, дающие всё более убедительные результаты.

1. Gatys, L. A. A Neural Algorithm of Artistic Style / L. A. Gatys, A. S. Ecker, M. Bethge // arXiv:1508.06576 – 2015.
2. Prisma Labs [Electronic resource] / Mode of access: <https://prisma-ai.com> – Date of access: 29.09.2019.
3. John, V. Disentangled Representation Learning for Non-Parallel Text Style Transfer / V. John, L. Mou, H. Bahuleyan, O. Vechtomova // arXiv:1808.04339 – 2018.
4. Ashual, O. Deep Text Style Transfer / O. Ashual, T. Jurgenson, D. Grinberg // 2017.
5. Goodfellow, Ian J. Generative Adversarial Networks / I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio // arXiv:1406.2661 – 2014.
6. Lample, G. Multiple-Attribute Text Rewriting / G. Lample, S. Subramanian, E. Smith, L. Denoyer, M. A. Ranzato, Y-L. Boureau // International Conference on Learning Representations – 2019.
7. Dai, N. Neural Machine Translation by Jointly Learning to Align and Translate / N. Dai, J. Liang, X. Qiu, X. Huang // arXiv:1409.0473 – 2019.
8. Dai, N. Style Transformer: Unpaired Text Style Transfer without Disentangled Latent Representation / N. Dai, J. Liang, X. Qiu, X. Huang // arXiv:1905.05621 – 2019.

О РАЗРАБОТКЕ ВЕБ-СЕРВИСА ДЛЯ СЕМАНТИЧЕСКОГО ПОИСКА ФИЛЬМОВ

Заблоцкий В. В., Кита М. А., Рудикова Л. В.

Кафедра современных технологий программирования, Гродненский государственный университет имени Янки Купалы

Гродно, Республика Беларусь

E-mail: viktorzablotsky@gmail.com, kitaetoya@gmail.com, rudikowa@gmail.com

В работе излагаются общие подходы к проектированию и реализации веб-сервиса, предназначенного для поиска информации о фильмах на основе семантики их названия и описания.

ВВЕДЕНИЕ

В современном мире существует огромный спрос на различную продукцию киноиндустрии, в частности фильмы, сериалы, мюзиклы и пр. В этой сфере задействованы не только огромные людские ресурсы, но и бюджеты, сопоставимые с ВВП не самых бедных стран. Так, согласно англоязычной Википедии, годовой оборот киноиндустрии в 2018 году оценивается от 41 до 136 миллиардов долларов. В связи со значительно возросшим уровнем информационных технологий, большую популярность приобретают Интернет-ресурсы, предназначенные для сохранения информации о фильмах и удобного поиска. Такие ресурсы позволяют найти всю необходимую информацию, связанную с каким-либо фильмом, а также изучить оценки и отзывы критиков и зрителей, что позволяет принять взвешенное решение о просмотре фильма. Среди таких ресурсов можно выделить Интернет-базу данных фильмов IMDb и российский аналог – Кинопоиск. Эти ресурсы содержат информацию о миллионах фильмов, которые были созданы с самого зарождения кинематографа и создаются по сей день.

Разрабатываемое приложение преследует те же цели, что и перечисленные выше поисковые базы, однако также позволяет вести поиск не только по прямому совпадению слов в названиях фильмов, но и по семантическому значению поискового запроса. Информация о фильмах, используемая в приложении, получена из общедоступных датасетов сайта IMDb.

I. ЭТАПЫ РАЗРАБОТКИ ВЕБ-СЕРВИСА

Работу над реализацией приложения можно разделить на несколько этапов.

Первый этап – анализ предметной области и определение функциональных требований к сервису. На данном этапе были выделены несколько источников данных, пригодных для реализации доменной области, проанализирована структура данных и на ее основе выделены основные направления функциональности, которые должен предоставлять сервис.

Второй этап – проектирование базы данных. Здесь были проанализированы и выбраны

оптимальные средства работы с базами данных, а также разработана физическая модель данных приложения.

Третий этап – проектирование архитектуры приложения. В результате были выбраны технологии и средства реализации приложения, а также в качестве рабочей архитектуры была выбрана многоуровневая архитектура клиент-сервер.

Заключительный этап – реализация серверной и клиентской частей приложения, а также их тестирование и развертка в облачной инфраструктуре.

II. ОБЩИЕ ПОДХОДЫ К РЕАЛИЗАЦИИ ВЕБ-СЕРВИСА

Для реализации веб-приложения используется клиент-серверная архитектура с применением принципов REST. Серверная часть реализована с помощью языка программирования Kotlin с применением фреймворков Spring Boot и Hibernate. В качестве СУБД используется MySQL и хостинг Amazon RDS. Клиентская часть выполнена средствами библиотеки React с применением паттерна Flux (Redux) для управления состоянием приложения. Также была разработана собственная библиотека компонентов с использованием препроцессора стилей SASS. В качестве языка программирования клиентской части выбран Typescript. Веб-приложение разворачивается в изолированной среде с помощью технологии контейнеризации Docker и в среде Amazon Web Services. Сервис состоит из нескольких слабо связанных микросервисов, среди которых можно выделить следующие: сервис авторизации и регистрации, сервис поиска, сервис разработки рекомендаций и сервис обновления данных. Данные сервисы практически не взаимодействуют друг с другом, что позволяет добиться большей автономности в их работе и реализации, однако они используют общую базу данных.

В разрабатываемой поисковой системе можно выделить ключевые архитектурные компоненты и уровни: клиентский уровень (состоит из множества компонентов отображения, компонентов управления состоянием и сервисов, объединенных в модули), прикладной уровень, уровень

бизнес-логики, уровень доступа к данным, уровень хранения данных.

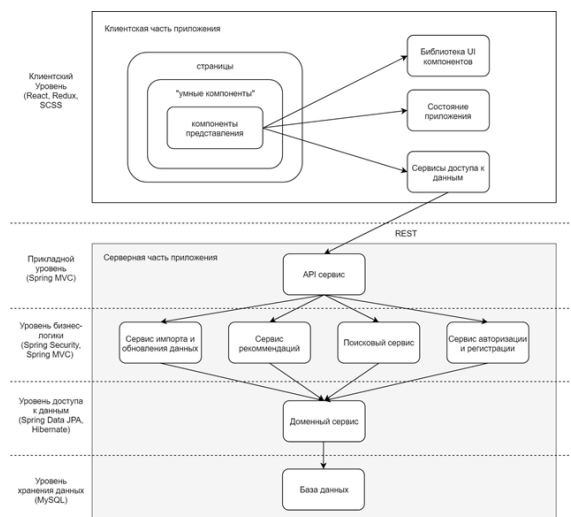


Рис. 1 – Архитектура веб-сервиса

Приложение реализовано по принципам многоуровневой архитектуры и разбито на несколько слоев. Каждый слой является абстракцией над более низкими слоями, предоставляя свой API верхним слоям. Благодаря такому разделению реализуется принцип ограниченной ответственности компонентов и упрощается их параллельная разработка и модификация.

На самом нижнем уровне находится база данных, в которой хранится вся импортированная из датасетов информация, а также пользовательские данные.

Доменный уровень представлен набором сущностей, моделей, репозиторий и сервисов, реализующих доступ к данным и их отображение в объектно-ориентированном стиле.

На уровне бизнес логики находятся сервисы импорта, обновления и поиска данных, а также сервис авторизации и регистрации пользователей.

Прикладной уровень содержит набор конечных точек (endpoints), через которые клиентские приложения могут осуществить доступ к данным, используя HTTP запросы. Таким образом, приложение придерживается RESTful архитектуры, предоставляя API клиентскому приложению.

Важной составляющей частью данного веб-сервиса является семантический поиск. Семантический поиск – способ и технология поиска информации, основанная на использовании контекстного значения запрашиваемых фраз, вместо словарных значений отдельных слов или выражений при поисковом запросе. При семантическом поиске учитывается информационный контекст, местонахождение и цель поиска пользователя, словесные вариации, синонимы, обобщенные и специализированные запросы, язык запроса, а также другие особенности, позволяющие

получить соответствующий результат. В данной работе, используется уже готовый инструмент word2vec.

Работа с технологией word2vec осуществляется следующим образом: word2vec принимает большой текстовый корпус в качестве входных данных и сопоставляет каждому слову вектор, выдавая координаты слов на выходе. Сначала он создает словарь, «обучаясь» на входных текстовых данных, а затем вычисляет векторное представление слов. Векторное представление основывается на контекстной близости: слова, встречающиеся в тексте рядом с одинаковыми словами (а, следовательно, имеющие схожий смысл), в векторном представлении будут иметь близкие координаты векторов-слов. Полученные векторы-слова могут быть использованы для обработки естественного языка и машинного обучения.

В word2vec существуют два основных алгоритма обучения: CBOW (Continuous Bag of Words) и Skip-gram. CBOW – «непрерывный мешок со словами» модельная архитектура, которая предсказывает текущее слово, исходя из окружающего его контекста. Архитектура типа Skip-gram действует иначе: она использует текущее слово, чтобы предугадывать окружающие его слова. Пользователь word2vec имеет возможность переключаться и выбирать между алгоритмами. Порядок слов контекста не оказывает влияния на результат ни в одном из этих алгоритмов. Получаемые на выходе координатные представления векторов-слов позволяют вычислять «семантическое расстояние» между словами. И, именно основываясь на контекстной близости этих слов, технология word2vec совершает свои предсказания. Так как инструмент word2vec основан на обучении нейронной сети, чтобы добиться его наиболее эффективной работы, необходимо использовать большие корпуса для его обучения. Это позволяет повысить качество предсказаний. В данной работе информация о фильмах кодируется в векторы, которые затем используются для предоставления пользователю возможностей семантического поиска фильмов по их сюжетам.

III. ЗАКЛЮЧЕНИЕ

Разработанное приложение представляет собой пользовательский интерфейс для поиска контента, предоставленного датасетами сервиса IMDb. Приложение позволяет просматривать фильмы по категориям, искать и фильтровать по различным критериям фильмы, а также просматривать детализированную информацию о фильмах и личностях. В дальнейшем приложение можно развивать в сторону увеличения отображаемого контента, а также расширения поисковых возможностей и социальной составляющей.

АНАЛИЗ МЕТОДОВ И АЛГОРИТМОВ ПОИСКА ПЛАГИАТА В ИСХОДНОМ КОДЕ

Иванов И. П.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: com.ilya.ivanov@gmail.by

В тезисах рассматриваются классификации методов поиска плагиата исходного кода, анализируются преимущества и недостатки некоторых современных алгоритмов поиска плагиата в исходном коде.

ВВЕДЕНИЕ

Вопрос плагиата очень стар и до сих пор актуален. Сама задача поиска плагиата довольно широка и сильно зависит от предметной области. Важная разница между плагиатом исходного кода и свободным текстом является то, что методы, используемые для их обнаружения различаются. Обнаружение копирования исходного кода является хорошо изученной областью и считается проще, чем обнаружение плагиата в свободном тексте, поскольку в исходном коде набор слов, который можно использовать, ограничен и строго регламентирован.

I. КЛАССИФИКАЦИЯ МЕТОДОВ

Принято выделять два метода в оценке уровня заимствования в исходном коде [1]:

- атрибутивные методы;
- структурные методы.

Первыми в данной области появились атрибутивные методы. Сути их заключается в представлении исходного текста в виде точки в n -мерном пространстве, где каждая координата является некоторой количественной характеристикой программы. Примером такой характеристики может служить количество строк кода, количество операторов ветвления и циклов, количество переменных и т.д.

Две программы могут считаться похожими, если соответствующие точки в пространстве признаков близки или совпадают.

Недостатками метода является:

- высокий уровень ложных срабатываний;
- поверхностное изменение кода (без изменения логики программы) может скрыть факт заимствования;
- метод практически не работает, если задачей стоит поиск заимствования частей программы.

Другой, более современный метод состоит в сравнении программ с учетом их структуры. Эта процедура более сложная, чем сравнение отдельных атрибутов программы. Очевидно, что это такие алгоритмы имеют высокую сложность и требуют больше ресурсов. Классическим примером структурного подхода является построение абстрактного дерева программы с последую-

щим сравнением деревьев для разных программ. Сложность такого метода кубическая, что на практике не даёт возможности эффективно применять его для большого числа длинных программ. Сложность реализации структурных алгоритмов является платой за их точность.

Алгоритмы, относящиеся к структурному методу, мы и рассмотрим далее.

II. СТРУКТУРНЫЕ АЛГОРИТМЫ

Здесь мы рассмотрим несколько современных структурных алгоритмов для поиска плагиата с исходном коде, имеющие под собой готовые реализации.

Прежде чем начать рассмотрение, давайте познакомимся с основной моделью представления программы в подобных алгоритмах, а именно: токенов.

Пусть у нас есть две строки кода $for(int i = n; i >= 0; i --)$ и $for(int k = n; k > -1; k --)$. Для многих очевидно, что обе строки кода эквивалентны, однако это не так очевидно для ЭВМ. Чтобы бороться с такими попытками сокрытия плагиата было придумана модель *токенизованного представления кода*. Суть заключается в игнорировании несущественных (легко модифицируемых) деталей программного кода, таких как названия переменных, функций и т.д. Процедуру токенизации можно описать следующим образом:

- каждому оператору языка приписывается заранее известный код;
- порядок следования кодов соответствует порядку следования операторов в исходном коде.

ЖАДНОЕ СТРОКОВОЕ ЗАМОЩЕНИЕ

Рассмотрим подробнее эвристический алгоритм получения жадного строкового замощения (Greedy String Tiling [2]). На вход алгоритма поступает две строки символов над фиксированным алфавитом (токенизированное представление кода), а на выходе выдает набор общих для двух строк непересекающихся подстрок, близкий к оптимальному. В алгоритме задействованы две эвристики:

- примем, что более длинные последовательные пересечения лучше, чем множество меньших и непоследовательных, в том числе, если в сумме длина последних больше;
- алгоритм игнорирует пересечения, длины которых меньше заданного порога.

Обе эвристики способствуют потере оптимального замощения, но результаты работы алгоритма, как правило, достаточно близки к нему, чтобы свидетельствовать о факте плагиата. Также вторая эвристика способствует фильтрации шумов, то есть маленьких участков кода, которые случайно совпали в текстах исходных кодов и не являются плагиатом. Сложность худшего случая для этого алгоритма является $O(n^3)$, но на практике она ниже $O(n^2)$, где n – длина токенизованного представления исходного кода программы. Также можно отметить высокую устойчивость алгоритма ко вставке операторов. Этот алгоритм успешно используется в программном средстве JPlag [3].

МЕТОД ИДЕНТИФИКАЦИОННЫХ МЕТОК

Данный метод предоставляет возможность находить копии и частичные копии в текстовой базе большого объема. Для этого каждому файлу сопоставляется его более краткое представление – идентификационные метки (fingerprints).

Алгоритм локальной дактилоскопии можно представить так:

1. На вход алгоритма подается токенизованное представление исходного кода;
2. Текст разбивается на токены по k -грам алгоритму [4];
3. От каждого такого отрезка берется хэш-функция. На практике часто используется хэш функция из алгоритма Рабина-Карпа [5];
4. Последовательность значений, полученная на предыдущем шаге, разбивается на последовательность пересекающихся окон (отрезков) длиной w . Значение w получается как $w = t - k + 1$, где t – порог гарантии (если у двух программ есть общая подстрока длиной как минимум t , то она будет найдена), k – порог шума (минимальная длина рассматриваемой подстроки), так чтобы $k \leq t$. Оба значения выбираются пользователем.
5. Происходит выборка значений хэш-функции для каждого окна методом просеивания [6]. В каждом окне выбирается минимальное значение хэша. Если таких значений несколько, то выбирается самое правое.

Полученный набор значений называется отпечатком документа. Если у двух документов совпадают значения отпечатков, то, с большой

долей вероятности, в них есть совпадающие подстроки, что может свидетельствовать о плагиате.

К достоинствам алгоритма можно отнести:

- по итогам все отпечатки можно занести в базу данных, которая ускорит проверку один-против-всех;
- общие подстроки длиной меньше, чем порог шума, игнорируются;
- общие подстроки длиной больше, чем порог гарантии, гарантированно будут выявлены;
- алгоритм не чувствителен к перестановкам больших фрагментов кода;
- разбиение участка совпадения вставкой уникального блока, перестановка небольшого количества операторов, а также переименование переменных и функций слабо изменят значение функции схожести.

Недостатками же можно отметить:

- невысоких шанс совпадения отпечатков программ, но отсутствие совпадения в исходных кодах;
- для нахождения непосредственно совпадающих участков требуется применение других алгоритмов (например, алгоритм суффиксного дерева [7]);
- замена оператора на схожий (например, while на for) в середине совпадающего блока, каждый длиной меньше k может привести к полному игнорированию совпадения.

Данный алгоритм успешно применяется в программном средстве MOSS [8].

СПИСОК ЛИТЕРАТУРЫ

1. T. Lancaster, F. Culwin, “Classifications of Plagiarism Detection Engines”, *Italics*, Vol. 4 (2), 2005.
2. Wise M.J. String similarity via greedy string tiling and running Karp-Rabin matching. // Dept. of CS, University of Sydney. December 1993.
3. Institute for Program Structures and Data Organization [Электронный ресурс]. – Электронные данные. – Режим доступа : <https://jplag.ipd.kit.edu> – Дата доступа: 21.09.2019.
4. N-gram Language Models [Электронный ресурс] : Article / Stanford University. – Электронные данные. – Режим доступа : <https://web.stanford.edu/~jurafsky/slp3/3.pdf> – Дата доступа: 20.09.2019.
5. R. Rarp, M. Rabin, Pattern-matching algorithms. IBM Journal of Research and Development, 21(2): 249-260, 1987.
6. Aiken A., Schleimer S., Wikerson D. Winnowing: local algorithms for document fingerprinting. // Proceedings of ACM SIGMOD Int. Conference on Management of Data. San Diego. 2003. P. 76–85. ACM Press. New York, USA. 2003.
7. Document Overlap Detection System for Distributed Digital Libraries [Электронный ресурс]. – Электронные данные. – Режим доступа : https://www.researchgate.net/publication/221347616_Document_overlap_detection_system_for_distributed_digital_libraries – Дата доступа: 21.09.2019.
8. Plagiarism Detection [Электронный ресурс]. – Электронные данные. – Режим доступа : <https://theory.stanford.edu/~aiken/moss> – Дата доступа: 21.09.2019.

ПРОФЕССИОНАЛЬНЫЕ КОМПЕТЕНЦИИ СПЕЦИАЛИСТОВ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Комличенко В. Н., Салапура М. Н., Богданова Е. А.

Кафедра экономической информатики, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: v.komlitchenko@gmail.com, marinasalapura@gmail.com, bogdanova.competentum@gmail.com

Решение задачи качественной подготовки специалистов для ИТ-индустрии актуально в современных условиях и предполагает творческое объединение специалистов высшей школы, бизнеса, работодателей и обучающихся. Динамичность ИТ-сферы требует оптимального сочетания и постоянного взаимодействия теории и практики.

ВВЕДЕНИЕ

В настоящее время во всем мире сохраняется крайне высокая потребность в квалифицированных специалистах в области ИТ. Республика Беларусь входит в десятку стран мира по проведению дружественных к ИТ-бизнесу реформ по оценке Всемирного банка. ИТ-индустрия занимает третье место в стране по количеству вакансий. 60,5% ИТ-специалистов заняты в аутсорсинговых услугах.

Белстат опубликовал статистические данные на 2018/2019 учебный год по ВУЗам, которые свидетельствуют о том, что количество студентов в ВУЗах Беларуси за последние 8 лет также сократилось. В 2010 году белорусские вузы приняли 100,5 тыс. студентов, а в 2018 году на первый курс было зачислено всего 58,8 тыс., в 2010/2011 учебном году в белорусских университетах числилось 442,9 тыс. студентов, в 2018/2019 — 268,1 тыс., что в 1,6 раза меньше.

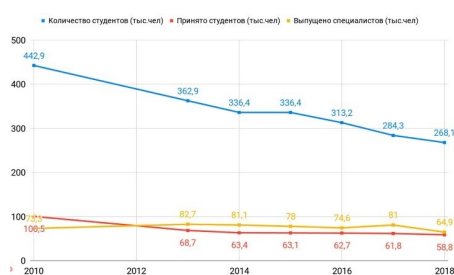


Рис. 1 – Динамика численности студентов в ВУЗах Беларуси

ПРОФКОМПЕТЕНЦИИ ИТ-СПЕЦИАЛИСТОВ

Учитывая демографическую ситуацию в Республике Беларусь в ближайшие годы можно говорить о снижении численности студентов, в том числе и в сфере ИТ. При снижении численности ИТ-студентов без принятия специальных мер по существенному изменению их доли в общей численности обучающихся, ситуация по удовлетворению потребности в ИТ-кадрах может стать неудовлетворительной.

Не менее остро стоит проблема отбора содержания образования. Вопрос обучения специа-

листов, занимающихся разработкой и внедрением информационных технологий, дискутируется на протяжении ряда лет [1]. В тоже время представители бизнеса считают, что в области обеспечения индустрии ИТ высококвалифицированными кадрами накопился целый пласт проблем.

Во-первых, обучение должно быть поставлено таким образом, чтобы ИТ-специалисты обладали фундаментальными знаниями в области computer science, которые позволяли бы постоянно повышать свою квалификацию, т.к. ситуация в сфере ИТ постоянно и динамично развивается. Во-вторых, специалисты должны иметь практические навыки по созданию программных продуктов с широкими функциональными возможностями, удовлетворяющими многообразным потребностям. Сфера образования должна готовить специалистов, которые могли бы приступать к производственной деятельности сразу после получения диплома.

В настоящий момент в Республике Беларусь действует двухступенчатая модель получения высшего образования, принятая за основу в болонской декларации. На первой ступени осуществляется подготовка на квалификацию бакалавра (как правило, 4 года обучения) и, на второй – на квалификацию магистра (еще не менее 2 лет). По мнению заведующего лабораторией открытых информационных технологий факультета ВМК МГУ имени М.В. Ломоносова, профессора, доктора технических наук, академика Академии информатизации образования В.А. Сухомлина «для области знаний, важнейшей тенденцией развития которой является быстрое расширение ее научных и прикладных горизонтов, двухступенчатая модель более логична».

Другой точки зрения придерживаются представители бизнеса. По мнению директора профильной Ассоциации «ИТ Украины» Виктора Валева, озвученному на заседании рабочей группы по подготовке ИТ-специалистов при Министерстве образования и науки, программы подготовки ИТ-специалистов должны унифицироваться, ориентируясь на наиболее типичные для отрасли специальности.

Результаты исследования показывают, что одним из важнейших критериев при приеме на работу является наличие практического опыта. Чтобы поставлять на рынок труда программистов, отвечающих запросам работодателей, система их подготовки должна быть не просто модульной, но и многоступенчатой, дающей как специализированные знания, так и практический опыт. Вместе ступени этой системы составляют единый цикл. После прохождения которой каждой человек должен иметь возможность стать полноценным специалистом, найти работу и, в то же время (при желании) продолжить образование.

Первая ступень – это техникум или колледж при вузе. Младшего специалиста нужно обучать тем же дисциплинам и на том же уровне, что и бакалавра, просто в меньшем объеме. Должно быть больше практики, развитие навыков программирования; теория может составлять не более половины объема часов на обучение. В хорошо построенном бизнесе требуется до 50% ИТ-специалистов именно такой квалификации и такого уровня образования. После того, как младший специалист приобретет практический опыт в программировании (от 2 до 5 лет), он получает возможность продолжить обучение на следующей ступени. Все это время предполагается, что будущий программист включен в проекты, тренинги, курсы, конкретную технологию. В дальнейшем он принимает решение: продолжить ли обучение и стать проектным менеджером или дальше совершенствоваться в своей области. В случае изменения технологий такого специалиста можно достаточно быстро переучить.

Вторая ступень – бакалавриат, специальный уровень. На курс обучения отводится уже 4 года. Специалист с дипломом бакалавра должен уметь выполнять работу более сложного, чем младший специалист, уровня, требующую основательной теоретической подготовки. Он должен понимать разные технологии, сценарии взаимодействия систем и работы пользователей, уметь разрабатывать архитектуру решений, модели баз данных и т.п.

Третья ступень – магистратура, управленческая подготовка для специалистов, которые смогут самостоятельно вести сложные проекты, понимать организационные моменты бизнеса, работу смежных отделов и подразделений, основы бюджетной практики и управления людьми. Обучение на этом уровне включает такие курсы, как управление проектами, ресурсами и знаниями, т.е. к имеющимся специальным знаниям добавляется бизнес-образование. На подготовку магистра отводится еще 1,5 – 2 года. А работа по распределению могла бы рассматриваться как определенная практика по приобретению навыков работы по профессии, что позволило бы

уже на базе этого опыта определять и проходить следующий этап обучения в очно-заочной форме обучения.

Такая система помогла бы многим людям совершить меньше ошибок в выборе профессии, реализовать, в случае возникновения необходимости, изменение профессиональной траектории. Нет никаких гарантий, что всем, кто сегодня получил образование по конкретным узким направлениям, завтра найдется рабочее место. Встает задача подготовки студентов так, чтобы они быстро могли переучиваться и получать востребованную профессию. Для этого необходимо построить такую модель обучения специалистов для ИТ-индустрии, при которой они «растут», получают знания и формируют навыки, проходя все этапы реализации проекта, осваивая все проектные роли. По мнению профессионалов, не менее 20% учебного времени студенты должны заниматься выполнением реальных проектов (2–3 проекта на протяжении каждого семестра) [2].

Отечественная система высшего образования работала достаточно эффективно и была оправдана тогда, когда специалиста и «заказывало», и «потребляло» государство, оплачивавшее процесс подготовки. Сейчас специалисты «заказывает» государство в лице Министерства образования, а «потребляет», в основном, бизнес. Формулировки профессиональных компетенций, прописанные в образовательных стандартах, должны быть основаны на реальных потребностях индустрии. В области информационных технологий востребована система обучения, которая позволяла бы: выдерживать стандарты подготовки, достигать взаимозаменяемости специалистов. Компетенции выпускников должны соответствовать квалификационным стандартам, складывающимся на практике, а также отраслевым стандартам. Для того чтобы снова утвердить определяющую роль диплома, ВУЗы обязаны максимум своих выпускников выводить на уровень требований, зафиксированных в стандарте; оперативно готовить необходимое количество специалистов разного уровня квалификации, умеющих быстро обучаться новым технологиям. Решение таких важных задач в области подготовки специалистов для ИТ-индустрии не только актуально и важно в современных условиях, а предполагает творческое объединение ВУЗа, бизнеса и работодателя в этом важном и значимом вопросе.

1. Жужжалов В.Е. Специфика обучения программированию при подготовке студентов-информатиков // Вестник МГПУ. Сер. «Информатика и информатизация образования». – М. – 2004. – № 1 (2). – С. 56–61
2. Сухомлин В.А. Концепция нового образовательного направления [Электронный ресурс] // Открытые системы. 2003. №2. URL: <http://www.osp.ru/os/2003/02/182628/> (дата обращения 04.10.2019)

3D-МОДЕЛЬ ЧЕЛОВЕЧЕСКОГО ТЕЛА И ИЗМЕРЕНИЕ ЕГО ПОКАЗАТЕЛЕЙ ПРИ ПОМОЩИ КАМЕРЫ

Кулыба В. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем, Кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: kulyba.vadim@gmail.com, gurinovich@bsuir.by

В данной статье исследуются алгоритмы построения 3d моделей человеческого тела и замеров его ключевых показателей (обхват в области плеч, обхват в области талии, форма лица, положение глаз). Для решения проблемы использованы алгоритмы и модели глубокого обучения. Ключевые слова: 3d модель, машинное обучение, камера с датчиком глубины, преобразование изображений и видео, датасет. Компьютерное зрение — теория и технология создания машин, которые могут производить обнаружение, отслеживание и классификацию объектов.

ВВЕДЕНИЕ

В настоящее время многие крупные компании работают над задачей построение точных 3d моделей человека. Существует несколько причин заинтересованности в построение 3d моделей: первая — заинтересованность в увеличении объема online продаж одежды; вторая причина — это предполагаемый AR/VR/XR рынок.

I. СНЯТИЕ ПОКАЗАТЕЛЕЙ ТЕЛА ЧЕЛОВЕКА

В первой части статьи рассмотрим простой способ измерения параметров тела при помощи как обычной камеры, так и глубинной. При условиях близких к идеальным, то есть человек стоит ровно и смотрит в камеру можно использовать обычную камеру. Суть метода заключается в использовании методологии применяемой в глубоком обучении под название pose estimations.

Pose estimations - это одна из распространенных задач глубокого обучения, по построению модели костей человека, по его фотографии, чтобы оценить его позиционирование на изображении или видео.

На данный момент уже существует множество предобученных моделей для задачи pose estimations разного качества. На выходе их отличается лишь точность и датасет на основе которого мы получаем список точек, предполагаемого скелета. Основные датасеты для этой задачи: MPII Human Pose Dataset, COCO dataset.

При помощи этих точек на выходе модели, при условии, что человек стоит ровно можно построить зависимости опираясь на коэффициенты при помощи которых вычислить параметры тела человека. В случае же если человек стоит не ровно, стоит использовать камеру с датчиком глубины, при помощи этой камеры можно вычислить углы наклона и по схожей методике вычислить параметры тела с поправкой на углы.

Однако стоит отметить, что подход является эффективным для снятия только 2d параметров человека.

II. ПОДХОДЫ ДЛЯ ПОСТРОЕНИЯ 3D МОДЕЛЕЙ ЧЕЛОВЕКА

Для построения 3d моделей человека будем опираться на камеры с датчиком глубины, для того чтобы эффективно строить 3d сцены вокруг человека. У таких камер по стандарту есть sdk для работы с различными видами отображений в 3d, таких как: point cloud, triangle mesh и другие. На схеме ниже представлен стек технологий и алгоритмов, которые нужны для построения 3d моделей (Рис. 1).

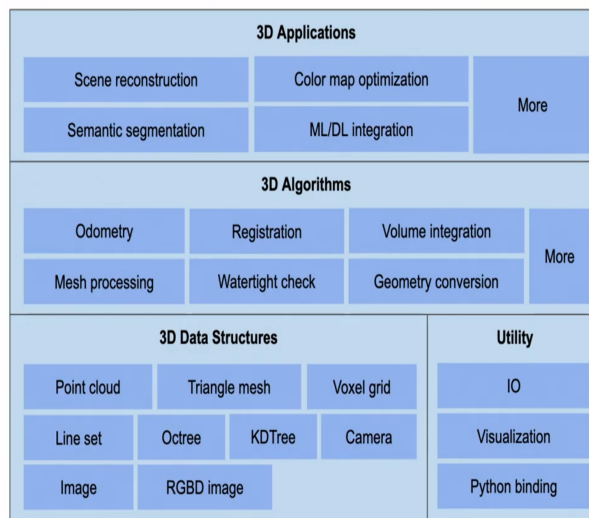


Рис. 1 – Стек технологий при построении 3d модели

Введем основные понятия блоков представленных на схеме:

Сегментация изображений (Semantic segmentation) — это одна из важнейших задач компьютерного зрения, которая представляет собой процесс разделения цифрового изображения на несколько сегментов. Цель сегментации заключается в упрощении и/или изменении представления изображения, чтобы его было проще и легче анализировать.

Реконструкция сцены (Scene reconstruction) — это процесс частичной реконструкции 3d пространства из RGB (RGBD) изображений.

Одометрия — это процесс соединения последовательности изображений в единое пространство.

Рассмотрим упрощенную последовательность действий для построения 3d модели человека. Изначально при помощи камеры получаем последовательность изображений. Следующим шагом является сегментация человека на изображениях при помощи моделей глубокого обучения. После чего зная, где находится человек на изображениях, мы можем удалить шумы с изображений. Далее при помощи библиотеки использующей одометрию мы можем объединить изображения в 3d модель человека (Рис 2).

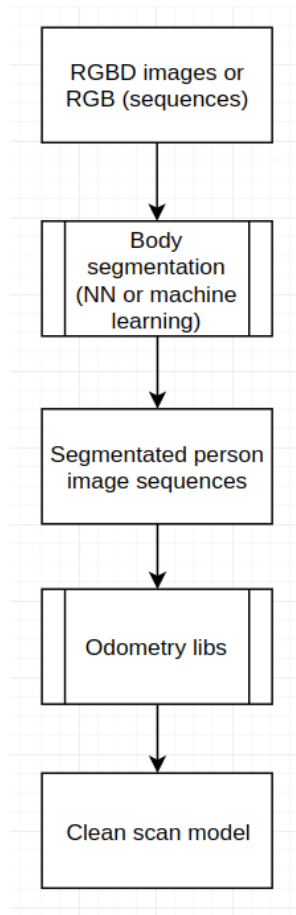


Рис. 2 – Упрощенная последовательность действий для построения 3d модели человека

В схеме были опущены некоторые пункты, которые зависят от камеры, такие как: соединение изображений с нескольких камер, так как одной будет недостаточно для детального сканирования и процесс работы с одометрией реализуется в многих библиотеках по разному при помощи многих видов преобразований.

III. КАМЕРЫ ГЛУБИНЫ

Камеры глубины, которые снимают видео, в каждом пикселе которого хранится не только цвет, а расстояние до объекта в этой точке. Такие камеры существуют уже больше 20 лет, однако в последние годы скорость их развития выросла многократно и уже можно говорить про революцию.

Самыми известными решениями на рынке камер глубины являются:

- Microsoft Kinect;
- Intel RealSense;
- Sony DepthSensing;
- ZED Stereo Camera.

Технологий измерения глубины развиваются в следующих направлениях:

- Structured Light камеры, или камеры структурного света, когда есть проектор (часто инфракрасный) и камера, снимающая структурный свет проектора;
- Time of Flight камеры, или камеры, основанные на измерении задержки отраженного света; Depth from Stereo камеры — классическое и, пожалуй, наиболее известное направление построения глубины из стерео;
- Light Field Camera — они же камеры светового поля или пленоптические камеры;
- Камеры, основанные на Lidar-технологиях, особенно свежие Solid State Lidars, которые работают без отказа примерно в 100 раз дольше обычных лидаров и выдают привычную прямоугольную картинку.

IV. ЗАКЛЮЧЕНИЕ

Таким образом, опираясь на пункты описанные выше можно полагать, что задачи построения качественных 3d моделей человека будут оставаться актуальными еще несколько лет, так как остаются нерешенными задачи:

- Построение моделей человеческого лица;
- Снятие параметров с 3d модели напрямую;
- Построение моделей без использования камер с дополнительными датчиками;
- Моделирование мелких деталей человеческого тела.

СПИСОК ЛИТЕРАТУРЫ

1. Камеры глубины — тихая революция (когда роботы будут видеть) <https://habr.com/ru/post/457524/> Часть 1.
2. Камеры глубины — тихая революция (когда роботы будут видеть) <https://habr.com/ru/post/458458/> Часть 2.
3. Semantic Segmentation — <https://medium.com/nanonets/how-to-do-image-segmentation-using-deep-learning-c673cc5862ef>.

ПОСТАНОВКА ЗАДАЧИ РАЗРАБОТКИ CRM-СИСТЕМЫ

Кукин Д. П., Свито И. Л., Свито А. И., Гриневич Я. Г.

Кафедра вычислительных методов и программирования, кафедра теоретических основ электротехники,
Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: {kukin, svito,ja.grinevich}@bsuir.by, alexandervirk@gmail.com

Целью настоящей работы является постановка задачи разработки CRM-системы управления взаимоотношениями с клиентами (от англ. Customer Relationship Management) с поддержкой принятия решений DSS. Система управления взаимоотношениями с клиентами должна облегчать привлечение новых клиентов, сбор и анализ информации о взаимодействии с клиентом, настройку бизнес-процессов и последующий анализ результатов.

ВВЕДЕНИЕ

Система поддержки принятия решений (СППР) (от англ. Decision Support System, DSS) – компьютерная автоматизированная система, целью которой является помощь людям, принимающим решение в сложных условиях для полного и объективного анализа предметной деятельности. Растущее количество пользователей смартфонов и количества мобильных приложений делает мобильную разработку одним из самых динамичных и конкурентных секторов, по темпам роста опережая разработку приложений для настольного ПК и микроконтроллеров [1,2]. Компаниям, занимающимся разработкой мобильных приложений, необходим инструмент для управления риском не получить выплату магазина приложений в срок и понести убытки. На финансовом рынке уже есть инструмент для управления подобными рисками – это факторинг. Однако современные банки отказываются сотрудничать с IT-компаниями, в связи с тем, что потенциальную прибыль таких компаний сложно оценить. Для поддержания отношений с потенциальными заёмщиками, математической оценке рисков, сбора и анализа клиентских трат необходимо спроектировать соответствующее программное обеспечение. Подобные системы называются системами управления взаимоотношениями с клиентами. Основная сложность факторинга заключается в том, что рынок интернет-маркетинга динамичен и постоянно развивается. Более того, для грамотного анализа теоретической прибыли необходимо собирать огромное количество данных во всевозможных разбивках: по стране, по платформе и по полу/возрасту. Такой анализ сложно проводить вручную, без использования программных средств, поэтому необходимость подобной автоматической программной системы очевидна. У текущих программных решений есть определённые недостатки, такие как: ограниченное количество интеграций с магазинами и рекламными сетями, сложный интерфейс и отсутствие средств аналитики и предупреждения аномальной активности. Поэтому разумно разработать

систему, не обладающую данными недостатками. Для этого необходимо решить следующие задачи:

1. Изучить рынок и основные программные аналоги, определить их достоинства и недостатки. Проанализировать предметную область.
2. Спроектировать архитектуру приложения, определить основные сущности и связи между ними.
3. Реализовать программные обеспечения для функционирования системы.
4. Произвести анализ качества получившегося программного продукта, протестировать производительность.

I. ОБЗОР СУЩЕСТВУЮЩИХ АНАЛОГОВ

Существует множество различных CRM-систем. Как правило, большая их часть ориентирована на специфичную отрасль, например, на управление продажами. Salesforce, ProfitWell и LighterCapital ориентированы на работу с финансами. Данные решения особенно популярны среди мелкого и среднего бизнеса, ввиду неспецифичности области деятельности и общих алгоритмов взаимодействия с клиентами. Однако крупные компании разрабатывают собственные решения, основанные на особенностях делопроизводства. К таким системам можно отнести системы операторов связи и логистических центров супермаркетов. Самой крупной платформой для управления взаимоотношениями с клиентами является Salesforce.com. Данная система, помимо своих базовых функций, предлагает облачную систему управления базами данных, реестр всех подключенных компаний и различные средства для групповой работы, такие как почтовые сервисы, корпоративные чаты и системы управления задачами.

Основным компонентом Salesforce является Sales Cloud – функциональная часть CRM-системы для поддержки процессов продаж. Это основной модуль системы, непосредственно с которым работают администраторы и менеджеры. К недостаткам данного модуля можно отнести многоуровневую систему подписок. Полная функциональность и работа без подключения к сети интернет доступна лишь по самой дорогой подписке, остальные же не предоставля-

ют полной функциональности. Вторым модулем является модуль для конструирования клиентских сайтов и поддержки баз данных – Service Cloud. Данный модуль предназначен для поддержки своих клиентов и предоставляя им возможности самообслуживания.

Для конструирования собственных решений и модулей в системе существует платформа Force.com. Использование данной системы требует знания языка программирования Apex и реляционных баз данных. Также Salesforce предлагает облачную систему баз данных Database.com, являющуюся промежуточным слоем между Force.com и пользовательским интерфейсом. Данная система работает на технологической основе Oracle Database и предоставляет эластичную обработку данных в том смысле, что при неожиданном росте нагрузки на базу данных, ей будут предоставлены необходимые вычислительные средства для их обработки.

В системе существуют решения для организации внутрикорпоративной работы – Chatter и бизнес-каталог с информацией о предприятиях – Data.com.

К недостаткам данной платформы можно отнести её размеры – система предоставляет множество решений для маркетинга и управления персоналом, которые не являются решающими для компании, ведущей деятельность по факторингу. Более того, обслуживание системы становится дорогим при загрузке в неё большого количества данных. Система предоставляет огромное количество инструментов и решений, большая часть которых не нужна клиенту, но которые требуют оплаты и от которых невозможно отказаться. Также, интерфейс работы с системой со стороны клиента достаточно сложен и не дружелюбен для пользователя: страницы интерфейса содержат огромное количество управляющих элементов, большинство из которых редко используется.

Ещё одной платформой, осуществляющей аналитику и сбор данных для мобильных приложений, является ProfitWell. Данная платформа осуществляет сбор данных с различных рекламных сетей и проводит аналитику – считает рекламные метрики и визуализирует данные. Система работает с пятью рекламными сетями Stripe, Braintree, Zuora, Chargebee, Recurly и позволяет следить за тратами, заработками, удержанием и вовлечённостью аудитории. Система

не работает с гигантами рекламного рынка – Facebook и Google и не предоставляет более детальной информации о приложениях – количестве подписок, пробных периодов, возвратов средств и прочего. ProfitWell собирает информацию лишь с рекламных сетей, игнорируя данные с магазинов приложений. Более того, интерфейс для пользователя неудобен – на главном экране отображается лишь базовая аналитика, такая как графики отмен подписок и доходов. Графики для просмотра более сложных метрик, например, удержания аудитории, находятся во вложенных вкладках и требуют большого количества кликов.

II. ФОРМУЛИРОВКА ЗАДАНИЯ

Таким образом, необходимо разработать систему для анализа данных, приходящих с рекламных сетей и каналов на рекламу, и осуществлять помощь в факторинге клиентов на основе имеющихся данных. Минимальными требованиями к реализации программной системы будут являться следующие:

1. Система должна предоставлять интерфейс для клиента, чтобы присоединять свои приложения и отслеживать для них всевозможную техническую (количество возвратов средств, продлений подписок и т.д.) и финансовую информацию (заработки и траты).
2. Система должна реагировать на добавления приложений и рекламных сетей и регулярно подгружать данные с доступных источников.
3. Администратор должен иметь возможность просматривать финансовую информацию клиента и на его основе определять размер факторинга.
4. После осуществления факторинга администратор должен иметь возможность назначать платежи и следить за трансфером суммы.
5. Оповещать об аномальных действиях со стороны клиента путём электронной почты: о смене банковских счётов, паролей от рекламных сетей.

СПИСОК ЛИТЕРАТУРЫ

1. Система поддержки принятия решений [Electronic resource] / Система поддержки принятия решений – 2019. – Mode of access: <https://ru.wikipedia.org/wiki/> . – Date of access: 09.10.2019.
2. Тенденции и перспективы рынка мобильных приложений [Электронный ресурс] / Тенденции и перспективы рынка мобильных приложений . – Минск, 2019. – Режим доступа: <https://habr.com/ru/company/alconost/blog/323020/>. – Дата доступа: 09.10.2019.
3. Stickel C., Ebner M., Holzinger A. The XAOS Metric – Understanding Visual Complexity as measure of usability. – Work Learning, Life Leisure, Springer, 2010, – с. 278-290

ЭФФЕКТИВНАЯ РАБОТА С ПАМЯТЬЮ ЭВМ

Лукьянов В. Н.

Кафедра вычислительных методов и программирования, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: uladzislau.luk@gmail.com

В введении объясняется необходимость большого объема ОЗУ. В первом и втором разделах рассказывается о важности локальности, кэше и времени доступа к памяти ЭВМ. В третьем разделе предлагается альтернатива стандартному динамическому выделению памяти. Заключение.

ВВЕДЕНИЕ

Решаемые разработчиками задачи, по объемам обрабатываемых данных, можно разделить на два типа:

- задачи с большим количеством обрабатываемой информации;
- задачи с малым объемом обрабатываемых данных.

В первом случае, в следствии огромного объема данных, информация в адресном пространстве находится на значительном удалении друг от друга и крайне редко модифицируется, что затрудняет ее обработку процессором, ввиду низкой скорости операций запроса данных из оперативной памяти. Во втором случае объем обрабатываемой информации мал, что позволяет производить меньше обращений к ОЗУ и хранить данные в кэше процессора. В настоящее время количество задач требующих обработки малого объема информации постоянно снижается. Однако, следует помнить, что если стоит выбор между машиной с очень маленьким объемом ОЗУ, но с огромной ее скоростью и машиной с относительно большим объемом ОЗУ, вторая машина всегда выигрывает, если размер обрабатываемой информации превышает небольшой объем оперативной памяти первой машины.

I. Локальность

SRAM используется для создания временных копий данных в оперативной памяти, которые, с большой вероятностью, в скором времени будут использованы процессором. Это возможно, благодаря тому, что код программы и данные имеют временную и пространственную локальность. Это означает, что в течение короткого периода времени существует большая вероятность повторного использования одного и того же кода или данных. Для кода это означает, что в нем, скорее всего, есть циклы, так что один и тот же программный код будет выполняться снова и снова (идеальный вариант для пространственной локализации). Доступ к данным также в идеале ограничен небольшими регионами. Даже если память, используемая в течение короткого периода времени, не находится близко друг к другу, существует большая вероятность того,

что одни и те же данные будут использованы повторно в течение длительного времени (временная локальность). Для кода это означает, например, что в цикле производится вызов функции и эта функция находится в другом месте адресного пространства. Функция может находиться на значительном удалении в памяти, но вызовы этой функции будут близки во времени. Для данных это означает, что общий объем используемой одновременно памяти (размер рабочего блока) в целом ограничен, но использованная память, в результате произвольного доступа к ней, находится далеко друг от друга. Понимание того, что локальность существует, является ключом к пониманию концепции кэша процессора.

II. Кэш

Вскоре после внедрения кэша система усложнилась. Разница в скорости между кэшем и основной памятью снова увеличилась, до такой степени, что был добавлен другой уровень кэша, больше и медленнее, нежели первый. Исключительное увеличение размера кэша первого уровня не было возможным по экономическим причинам. Уже сегодня существуют ЭВМ с тремя уровнями кэша (см. рис. 1).

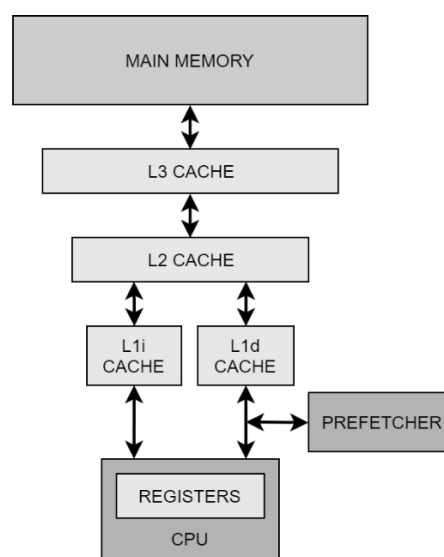


Рис. 1 – Схема ЦПУ с тремя уровнями кэша

Записи, хранящиеся в кэше, представляют собой не отдельные слова, а «строки» из множе-

ства последовательных слов. В ранних кэшах эти «строки», чаще их называют кэш-линиями, были 32 байта длиной; в настоящее время их размер составляет 64 байта. Если ширина шины памяти составляет 64 бита, то это означает, что на одну кэш линию приходится восемь операций передачи данных [1]. Оценочная стоимость во времени этой и других операций отражена в таблице 1.

Таблица 1 – Время выполнения операций с памятью

Операция	Время, нс
Доступ к кэшу данных первого уровня	0,5
Доступ к кэшу инструкций первого уровня	5
Доступ к кэшу второго уровня	7
Доступ к оперативной памяти	100
Последовательное считывание 1 Мбайта данных из памяти	250000

III. РАСПРЕДЕЛИТЕЛЬ ПАМЯТИ

Динамическое выделение памяти с помощью `malloc()` или глобального оператора `new` из C++ является крайне медленной операцией, ввиду дополнительных служебных операций на каждый вызов, использования кучи и фрагментации виртуальной памяти. Для предотвращения такого поведения рекомендуется использовать разработанные для конкретной цели распределители памяти. В системах с большим количеством операций считывания из памяти и записи в память, память, необходимая на протяжении всей жизни приложения, выделяется на этапе инициализации программного комплекса. Дальнейшие манипуляции с ней производятся через специальные распределители памяти, также называемые аллокаторами [2–3]. Рассмотрим основные типы аллокаторов:

1. Линейный аллокатор. Критически важным для этого аллокатора является то, что он не добавляет никаких дополнительных служебных операций при выделении и не изменяет предварительно выделенный буфер памяти. Не все распределители памяти будут обладать этими свойствами. Это делает его идеальным для низкоуровневых систем или для работы с памятью, доступной только для чтения;
2. Аллокатор, основанный на стеке. Чрезвычайно эффективен, когда необходимо выполнить несколько операций распределения памяти, и все они могут быть выполнены в обратном порядке. Кроме того, он также может быть использован для хранения данных, которые должны оставаться постоянными в течение некоторого времени с постоянно меняющимися данными в дополнение к ним;
3. Аллокатор, основанный на списке освобождения. Является «аварийным» аллокатором. Он предназначен для использования

в качестве стратегии резервирования в ситуациях с малым объемом памяти, когда память недоступна для структур управления, необходимых другим распределителям памяти. В таких ситуациях аллокатор свободного списка гарантирует, что память не будет потеряна, но с несколькими недостатками:

- сложность операций в свободном списке пропорциональна количеству свободных блоков;
 - структуры данных обладают низкой локальностью и, следовательно, потенциально низкой производительностью кэш-памяти.
4. Пул памяти. Выделяет область памяти один раз и разделяет эту область на фрагменты, которые точно соответствуют M объектам размером N. Пул памяти крайне полезен при выделении или освобождении памяти для объектов одного размера и с необходимостью постоянного динамического инстанцирования или уничтожения. Большинство из этих объектов создаются и уничтожаются в совершенно случайном порядке, в силу их динамической природы. Поэтому желательно иметь возможность выделять и освобождать память с минимальной фрагментарностью [4–5].

ЗАКЛЮЧЕНИЕ

Что стоит помнить и использовать в работе:

- временная и пространственная локальность крайне важна;
- чем больше данных хранится в кэше, тем быстрее они обрабатываются;
- чем ближе находятся данные, тем больше информации добавится в кэш за одну операцию чтения;
- стандартные способы динамического выделения памяти медленны;
- под каждую задачу существует соответствующий распределитель памяти.

СПИСОК ЛИТЕРАТУРЫ

1. What Every Programmer Should Know About Memory [Electronic resource] / U. Drepper. – Raleigh, N.C., 2007. – Mode of access: <https://people.freebsd.org/~lstewart/articles/>. – Date of access: 01.10.2019.
2. Blunden, B. Memory management: algorithms and implementation in C/C++ / B. Blunden // Plano. Wordware Publishing, Inc., 2002. – 360 p.
3. Gregory, J. Game Engine Architecture / J. Gregory // Boca Raton. CRC Press, 2018. – 1240 p.
4. Building your own memory manager for C/C++ projects [Electronic resource] / A. Sen, R. Kardam. – Armonk, N.Y., 2008. – Mode of access: <https://developer.ibm.com/tutorials/>. – Date of access: 02.10.2019.
5. Jacob, B. Memory Systems: Cache, DRAM, Disk / B. Jacob, S. W. Ng, D. T. Wang // Burlington. Morgan Kaufmann, 2007. – 900 p.

АВТОМАТИЗАЦИЯ ПРОЦЕССОВ ПРИНЯТИЯ РЕШЕНИЙ ПРИЁМНОЙ КОМИССИИ

Макаров А.

Факультет информационных технологий и управления, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: arhangel.makarov.1994@bk.ru

Рассматриваются вопросы разработки информационной системы (ИС) управления вузом. Приведена разработанная информационная система, обеспечивающая сквозную автоматизацию организации и управления работы приемной комиссии. Описано функционирование автоматизированной подсистемы приемной комиссии (ПК) вуза. ИС управления рассмотрена как человеко-машинная система с автоматизированной технологией получения результатов организации, документационного оформления и управления.

ВВЕДЕНИЕ

В последние годы наблюдается тенденция повышения престижа высшего образования, ориентации выпускников школ на поступление в высшие учебные заведения. Об этом свидетельствует неуклонно повышающийся конкурс абитуриентов на большинство гуманитарных, естественнонаучных и технических специальностей. Использование информационных технологий в приемных кампаниях высших образовательных учреждений обуславливается потребностью повысить эффективность управления документооборотом, сделать вуз преуспевающей, экономически выгодной организацией, реализующей важную социальную функцию по подготовке высококвалифицированных дипломированных специалистов и научных кадров. Высшие учебные заведения представляют собой сложную систему, имеющей целью удовлетворение потребностей личности, государства и общества в образовательных услугах высокого качества.

Содержание образовательной деятельности определяется вузом на основе требований соответствующих образовательных стандартов. Вуз как система подвержена воздействию внешней и внутренней сред и зависит от многих факторов организации ее деятельности. При этом общая деятельность вуза многогранна и охватывает учебную, учебно-методическую, научно-исследовательскую, социальную, производственно-хозяйственную и др. работы.

Высшим учебным заведениям как система и объект управления характерны следующие особенности:

1. результаты деятельности вуза определяются от внутренних и внешних связей и их динамичности;
2. наличие составляющих элементов по направлениям деятельности по горизонтали и уровню иерархии;
3. функционирование и объединение элементов системы для достижения поставленной цели и т.д.

I. АВТОМАТИЗАЦИЯ ПРОЦЕССА ПРИНЯТИЯ РЕШЕНИЯ

Сущность автоматизация процесса принятия решения сводится к распределению множества кандидатов на множество вакансий. Каждый из кандидатов претендует на некоторую совокупность вакансий из множества. Претендент упорядочивает вакансии из множества в соответствии со своими предпочтениями. Каждой группе вакансий соответствует формализованное представление группы требований, выполнение которых необходимо для получения права кандидату претендовать на вакансии. Также определяется множество всех возможных дополнительно учитываемых условий, выполнение которых может быть рассмотрено как преимущество (или наоборот недостаток) кандидата при прочих равных условиях. Автоматизация управления сегодня – это автоматизация различных областей учета, документооборота, анализа данных и т.д. с целью оперативной подготовки информации для принятия руководителями различных уровней обоснованных управленческих решений.

Общеизвестны преимущества автоматизации, а именно ускоряется информационный обмен, сокращаются рутинные операции, уменьшается количество ошибок, облегчается доступ к информации, появляются дополнительные средства быстрого анализа больших объемов данных и т.д.

Также в ходе автоматизации предприятия происходит упорядочивание и оптимизация бизнес-процессов, проводится реформирование структуры предприятия, часто меняется механизм принятия управленческих решений.

Перечислим задачи, на которые нацелена автоматизация управления вуза:

- а) создание или оптимизация единой системы планирования деятельности предприятия;
- б) поддержка принятия решений на всех уровнях управления на основе совершенствования процессов сбора и обработки различных видов информации и т. д.

Лицо, принимающее решение, в любой момент сможет получить консультацию нескольких высококлассных специалистов. Это позволит минимизировать вероятность допущения крупной управленческой ошибки.

II. МЕТОД КОНКУРСНОГО ОТБОРА

Чтобы ранжировать кандидатов, для каждой конкурсной позиции претендента, рассчитывается значение оценочной функции. Результатом конкурсного отбора должна стать матрица зачислений, каждый элемент которой равен единице, если в результате конкурсного отбора претендент был распределен на вакансию подмножества, и равен нулю в противном случае. Для конкурсного отбора в вузы представляет интерес множество величин, которые являются проходными баллами на вакансии множества. Если претендент, заявился на вакансию, где значение его оценочной функции для этой конкурсной позиции больше либо равно множеству величин и если приоритет всех остальных заявленных им вакансий ниже, то он должен быть распределен на эту вакансию. В ходе проведения конкурсного отбора возможно возникновение ситуации, при которой остается невостребованной часть мест на ряде специальностей, так как все заявившие их абитуриенты уже распределены. Необходимо заполнить множество оставшихся вакансий на основе личных пожеланий нераспределенных претендентов с наибольшими значениями оценочной функции, при этом инициативу выбора следует предоставить самим потенциальным кандидатам. Уже несколько лет ведутся работы по компьютеризации учебного процесса и созданию корпоративной сети управления учебным процессом на основе использования современных информационных технологий. Основная направленность работы ориентирована на разработку прикладных систем, способствующих дальнейшему совершенствованию управления образовательными процессами в вузе, а также формированию и упорядочиванию систем документооборота между подразделениями вуза.

Модель существующей ИС университета включает в себя следующие подсистемы (ПС): ПС «Отдел кадров сотрудников», ПС «Учебное Управление», ПС «Абитуриент», ПС «Отдел

кадров студентов», ПС «Бухгалтерия», ПС «Деканат», ПС «Тест». ИС обеспечивает поддержку принятия решения на всех этапах бизнес-процессов, в том числе по приему абитуриентов. Рассмотрим подробнее автоматизированную подсистему ПК. Автоматизированная подсистема ПК - обеспечивает организацию и сопровождения деятельности приёмной комиссии на всех этапах, начиная от анализа плана набора абитуриентов до формирования контингента студентов первого курса. Она взаимодействует с подсистемами учебного отдела, отдела кадров и бухгалтерией с целью исключения дублирования работ между отделами:

Основными задачами подсистемы ПК являются:

- формирование плана набора абитуриентов на новый учебный год;
- прием и оформление документов абитуриента по специальностям;
- формирование разных отчетов о ходе работы приемной комиссии;
- подготовка документов для рекомендации, подтверждения и зачисления абитуриентов по специальностям;

III. ВЫВОД

Предлагаемый нами алгоритм позволяет создавать специальное программное обеспечение для управления документооборотом потока абитуриентов с целью сокращения ошибок и конфликтных ситуаций при зачислении с минимально допустимым временем сопровождения одного абитуриента и с минимально допустимым числом персонала, задействованного в приемной кампании.

СПИСОК ЛИТЕРАТУРЫ

1. Зайцев А.С., Емельянов В.Ю., Воронин И.В. Автоматизация как основа оптимального планирования учебного процесса университета // Мехатроника, автоматизация, управление, 2007. № 5. С. 45-48
2. Крюков В.В., Шахгельдян К.И. Корпоративная информационная среда вуза: методология, модели, решения: Монография. Владивосток: Дальнаука, 2007. 301 с..
3. научная электронная библиотека «КИБЕРЛЕНИНКА» Режим доступа: <https://cyberleninka.ru/>. – Дата доступа: 10.10.2019.
4. Репозиторий БГУИР Режим доступа: <https://libeldoc.bsuir.by/>. – Дата доступа: 10.10.2019.

ИСПОЛЬЗОВАНИЕ ERP-СИСТЕМЫ MICROSOFT DYNAMICS 365 В ОБРАЗОВАТЕЛЬНЫХ УЧРЕЖДЕНИЯХ

Манин А. С., Нестеренко В. Н.

Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: mmts@bsuir.by

ERP-системы активно внедряются на производстве по всему миру. В последнее десятилетие преимущественно в развитых странах образовательные учреждения инвестируют значительные материальные и финансовые ресурсы во внедрение ERP-систем. В последние годы ERP-системы разных классов проникают и в белорусскую промышленность. Несмотря на множество плюсов, внедрение ERP-систем сопряжено с колоссальными затратами и рисками. Возможно, белорусские учреждения образования рано или поздно решатся на этот шаг. На примере Microsoft Dynamics 365 в данной работе рассматриваются основные этапы внедрения системы, влияние этого внедрения на процессы внутри университета и специфику модификаций стандартной ERP-системы для нужд университета. Данное исследование проводится в рамках работы по изучению процессов интеграции ERP-системы Microsoft Dynamics 365 с внешними системами.

ВВЕДЕНИЕ

Ключевой идеей внедрения ERP-систем на предприятиях является объединение разрозненных функциональных ветвей внутри него в единую информационную систему, и автоматизация рутинных процессов. Еще в девяностые годы спектр ERP-систем расширился от специализированных систем поддержки процессов производства до полноценных, детально смоделированных и стандартизированных моделей средних и крупных предприятий, включающих такие процессы, как заказы на закупку и продажу, финансы, управление активами и управление персоналом. В последнее время в ERP-системы включаются модули для работы в грантами, для автоматизации маркетинга, электронной коммерции, логистические системы. В последнее десятилетие по всему миру университеты и школы, схожие в своей структуре с современными промышленными предприятиями, отказываются от устаревающих систем управления предприятием и начинают использовать ERP-системы для автоматизации управления. Однако большинство современных ERP-систем по умолчанию не адаптированы под образовательные нужды и не покрывают ключевые процессы, протекающие в университетах. [1] Именно такие адаптации ERP-систем к требованиям образовательных учреждений и соответствующие расширения стандартной функциональности, на примере Microsoft Dynamics 365, являются объектом анализа этого исследования.

I. ОСОБЕННОСТИ ВНЕДРЕНИЯ ERP-СИСТЕМЫ В ОБРАЗОВАТЕЛЬНОМ УЧРЕЖДЕНИИ

Основные этапы внедрения любых ERP-систем на предприятии плотно пересекаются со стандартным процессом внедрения программных средств, включают в себя этапы планирова-

ния, проектирования, разработки, тестирования и последующей поддержки.

Особенностями внедрения ERP-системы являются необходимость миграции данных из предыдущих систем на этапе проектирования, либо интеграция новых решений со старыми. Из-за специфики интегрированной ERP-системы, этап тестирования предполагает переподготовку рабочих и персонала во всем учреждении образования. Внедрение нового программного средства предполагает параллельное тестирование новой системы вместе с использованием существующей системы. В случае ERP-систем необходимо спланировать синхронизацию этих процессов.

Microsoft Dynamics 365 по умолчанию содержит ряд модулей, необходимых для управления университетом. Так, рассматриваемая система имеет достаточно гибкий модуль финансового учета, модуль планирования, модуль управления активами и грантами, модуль контроля бюджета и затрат, систему подбора персонала и систему управления платежами. Как видно, Microsoft Dynamics 365 не содержит узкоспециализированных модулей для организации учебного процесса. Существуют различные подходы к проектированию и реализации этих модулей. [3] Ниже приводятся дополнительные модули, которые могут быть необходимы в ERP-системе учреждения образования:

- Система управления расписанием позволит создавать расписания как автоматически, так и вручную. Это гарантирует своевременное обновление информации о замене преподавателей, упрощает управление помещениями, ресурсами и преподавателями.
- Система управления приемом студентов помогает контролировать предварительный отбор и прием студентов, онлайн-регистрацию, регистрацию удостоверений

личности и отправку/получение электронных оповещений.

- Система управления оценками и экзаменами дает возможность проводить экзамены и создает условия для онлайн-экзаменов, оценивать успеваемость учащихся, составлять таблицы успеваемости, проводить массовые конкурсные/вступительные экзамены, а также предоставлять студентам и их родителям результаты обучения.
- Модуль управления общежитием упрощает организацию и процесс выдачи студентам и преподавателям мест в общежития, позволяет отслеживать, когда тот или иной студент находился в комнате или покинул общежитие. С помощью модуля может быть установлен контроль за предметами мебели в комнатах, межсеместровыми процессами выселения и заселения.

II. ЗАТРАТЫ И РИСКИ

Различные исследования в области ERP-систем за последние двадцать лет показывают, что их внедрение сопряжено с высокими рисками в силу различных причин (нечеткие цели, неучастие руководства в реализации проекта, недостаточно точная миграция данных из предыдущих систем, недостаточный уровень подготовки сотрудников, отсутствие натренированных навыков работы с новой системой). [4]

Внедрение ERP-системы также создают необходимость в значительных финансовых инвестициях для учреждения образования. По некоторым данным, средняя стоимость внедрения Microsoft Dynamics 365 составляет от \$250 тысяч до \$1 миллиона за вычетом последующей поддержки системы со стороны разработчиков, продолжающейся в среднем до двух лет. [5]

Кроме того, по статистике 80% внедрений Microsoft Dynamics 365 продолжаются в среднем на 4 месяца дольше запланированного срока. Первоочередной причиной для этого считается возникновение дополнительных требований в ходе этапа разработки. [5]

Основными причинами отказа от разработки и внедрения ERP-систем в образовательном учреждении тем не менее является в первую очередь неготовность руководства, работников и преподавателей к переходу, а не финансовые трудности, так как по средним оценкам финансовые результаты по внедрению приводят к значительным выгодам. [6]

III. ПЛЮСЫ ВНЕДРЕНИЯ ERP-СИСТЕМ В ОБРАЗОВАТЕЛЬНОМ УЧРЕЖДЕНИИ

Основной причиной внедрения ERP-систем в образовательных учреждениях называется повышение конкурентоспособности учреждения за счет предоставляемых сервисов как для работ-

ников университета, так и для студентов и их родителей. [7]

Кроме того, плюсами внедрения ERP-системы в университетах и школах считается возможность хранения и обработки больших массивов данных, возможность мониторинга всех образовательных процессов, решение вопросов разграничения доступа к информационным ресурсам, автоматизация процессов документооборота, что позволяет сократить количество используемых бумажных носителей, ускорить выполнение процессов и так далее. [8]

Безусловно, ERP-система не является обязательной для организации процессов внутри образовательного учреждения. Тем не менее для образования важно идти в ногу со временем, и замена устаревающих систем современной ERP-системой является важным шагом на этом пути.

СПИСОК ЛИТЕРАТУРЫ

1. Magic Quadrant for Cloud Core Financial Management Suites for Midsize, Large and Global Enterprises [Электронный ресурс] /Gartner, Inc. – 2019 – Режим доступа: <https://www.gartner.com/en/documents/3913508> – Дата доступа: 01.10.2019.
2. Goeun Seo, Challenges in Implementing Enterprise Resource Planning (ERP) system in Large Organizations: Similarities and Differences Between Corporate and University Environment / Seo Goeun // Working Paper CISL, Massachusetts Institute of Technology – 2013
3. Watson, E., Schneider, Helmut. Using ERP systems in education. Communications of The Ais / E. Watson, Helmut. Schneider // Communications of AIS – 1999 – Vol. 1, Ch. 9.
4. Implementing ERP systems in higher education institutions [Электронный ресурс] / Zornada, L., Velkavrh, Tamara. – Faculty of Management Koper, 2005. – Режим доступа: <https://www.academia.edu/6850431/Implementing-ERP-Systems-in-Higher-Education-Institutions> – Дата доступа: 24.09.2019.
5. Какие ERP-решения лучше всего показывают себя на практике [Электронный ресурс] / Про бизнес – 2015 – Режим доступа: <https://probusiness.io/statistics/976-kakie-erp-resheniya-luchshe-vsego-pokazyvayut-sebya-na-praktike.html> – Дата доступа: 22.09.2019.
6. 2019 ERP report [Электронный ресурс] / Panorama Consulting Group – 2019 – Режим доступа: <https://cdn2.hubspot.net/hubfs/4439340/2019-ERP-Report-3.pdf> – Дата доступа: 24.09.2019
7. Critical success factors for implementing an ERP system in a university environment [Электронный ресурс] / Nielsen, J. – Faculty of Engineering and Information Technology. Brisbane, Griffith University, 2002. – Режим доступа: <https://www.academia.edu/19685565/Critical-Success-Factors-for-Implementing-an-ERP-System-within-University-Context-Concepts-and-Literature-Review> – Дата доступа: 22.09.2019.
8. Использование ERP-систем как инструмента управления качеством в образовательных учреждениях [Электронный ресурс] / С. А. Батуркин, А. А. Скунцев – ЭКОНОМИНФО, 2007 – Режим доступа: <https://cyberleninka.ru/article/n/ispolzovanie-erp-sistem-kak-instrumenta-upravleniya-kachestvom-v-obrazovatelnyh-uchrezhdeniyah> – Дата доступа: 08.10.2019.

ИНТЕГРАЦИЯ В СРЕДСТВА ДОКУМЕНТООБОРОТА ERP-СИСТЕМЫ MICROSOFT DYNAMICS 365

Нестеренко В. Н., Манин А. С.

Факультет компьютерных систем и сетей, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: mmts@bsuir.by

Объектом исследования является анализ возможностей интеграции в процессы Microsoft Dynamics 365, связанные с документооборотом, для отслеживания возникновения документов и сбора метаданных. Данное исследование проводится в рамках научной работы, цель которой – построение набора программных средств, реализующих архивирование документов, порождаемых в бизнес-процессах Microsoft Dynamics 365, с использованием удалённого хранилища. В ходе работы были изучены и проанализированы документация, а также исходный код основных компонент, обеспечивающих функциональность средств документооборота в Microsoft Dynamics 365.

ВВЕДЕНИЕ

На сегодняшний день, для автоматизации управления предприятием, в качестве оптимального инструмента, используются ERP-системы. Системы этого класса содержат компоненты, которые поддерживают различные функции предприятия и постоянно взаимодействуют друг с другом. Таким образом, система управления предприятием осуществляет «воспроизведение» бизнес-процессов в программном обеспечении и сопровождает всякое действие каждого сотрудника. Это положительно сказывается на производительности предприятий в целом.

Однако, работе предприятий с ERP-системой сопутствуют большие объемы различного рода документов и файлов. В условиях, когда актуальность, гарантированность, своевременность документной информации имеют большое значение для предприятия, является разумным прибегнуть к организации и поддержанию удалённого хранилища документов и файлов, порождаемых в бизнес-процессах ERP-системы. В качестве хранилища можно использовать любой набор программных средств, при условии наличия способности к сетевому взаимодействию.

Одним из этапов, предшествующих транспортировке документов и файлов в удалённое хранилище, является этап получения копий документов и файлов, возникших в ERP-системе, а также сопутствующей информации, которая могла бы быть полезна для их последующей идентификации в структурах хранилища как при автоматизированном подходе, так и вручную. Вопросы, возникающие на данном этапе, являются объектом анализа этого исследования.

I. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДОКУМЕНТООБОРОТА

В ERP-системе Microsoft Dynamics 365 документы и файлы могут возникать по одной из двух причин.

Первой причиной является ассоциация заметок, документов или других файлов с сущностями ERP-системы посредством функциональности, реализованной в модуле Document Management ERP-системы. Под сущностью, в данном контексте, подразумеваются транзакции или заголовки транзакций. Таким образом, пользователь ERP-системы имеет возможность обеспечить, например, заказ на покупку (Sales Order) или заказ на продажу (Purchase Order) сопутствующими документами и файлами. При этом, в данном случае, мы имеем дело с документами и файлами, которые фактически существуют, т. е. они представлены индивидуальными файлами в файловой системе [1–2].

Вторая причина – добавление записей в журнал или обновление записей журнала. В Microsoft Dynamics 365, журнал – это последовательность зарегистрированных событий в рамках некоторой активности при этом каждое событие снабжается необходимой информацией для его описания. Примером может служить журнал счет-фактур (Invoice Journal). В таком журнале в качестве события может быть зарегистрирован факт создания счет-фактуры (Invoice). А информация, описывающая событие, чаще всего представлена идентификаторами товаров, их количеством и стоимостью; идентификаторами других сущностей ERP-системы, связанными с данным событием, такими как юридическое или физическое лицо (Customer) и др. [3].

Особенностью журналов является то, что зачастую на их основе происходит генерация документов. Для этого, как правило, используется технология SSRS (SQL Server Reporting Services). Она позволяет разработать верстку необходимых документов, называемых в Microsoft Dynamics 365 отчёты (Reports), а также предоставляет уникальный для данной ERP-системы способ формирования отображаемых данных, т. е. фактически поддерживает источник дан-

ных (Data Source) специфический для семейства ERP-систем Microsoft Dynamics [4–5].

II. ИНТЕГРАЦИЯ В DOCUMENT MANAGEMENT

В результате исследования документации и исходного кода модуля Document Management было выяснено, что он не предоставляет программный интерфейс (API), позволяющий легко обнаружить и использовать желаемые документы и файлы. Таким образом, для того, чтобы автоматизировать работу с данными модуля, придётся прибегнуть к интеграции собственного кода в исходный, а также к непосредственному обращению к управляемым им таблицам базы данных.

Структура базы данных модуля Document Management достаточно проста, а ключевыми в ней являются таблицы DocuType, DocuParameters, DocuValue и DocuRef.

Таблица DocuParameters содержит общие настройки модуля, в то время как таблица DocuType используется для конфигурирования типов файла. Тип файла, в данном случае, является абстрактным понятием, описывающим то, каким образом файлы, помеченные данным типом, будут обрабатываться ERP-системой. Одной из доступных конфигураций является место хранения ассоциируемых файлов. При этом данная настройка привязывается к типу файла, т. е. находится в каждой записи таблицы DocuType. Однако, пользователь имеет возможность указать системе использовать для хранения документов некоторого типа значение по умолчанию, значение по умолчанию располагается в таблице DocuParameters [1].

Таблица DocuValue содержит информацию о ассоциируемых файлах. Таковую, как название файла, оригинальное название файла, ссылку на тип файла, и, самое важное, путь к файлу. Путь к файлу определяется относительно каталога, указанного в настройке места хранения ассоциируемых файлов, которая расположена в таблице DocuType или DocuParameters.

С помощью таблицы DocuRef осуществляется ассоциация документов и файлов с сущностями ERP-системы. Говоря техническим языком, записи в таблице DocuRef могут ссылаться на записи, практически, в любых таблицах Microsoft Dynamics 365. Также, таблица DocuRef содержит идентификатор записи в таблице DocuValue.

Таким образом, комбинируя данные таблиц DocuType, DocuParameters, DocuValue и DocuRef, можно получить абсолютный путь к файлу и информацию, которая может быть использована в качестве сопутствующей к нему. Также, модифицировав должным образом методы вставки, обновления и удаления таблицы DocuRef можно отслеживать соответствующие события и реагировать на них.

III. ДОКУМЕНТЫ НА ОСНОВЕ ЖУРНАЛОВ

В Microsoft Dynamics 365 управление журналами реализуется с помощью технологии, называемой Formletter Framework. Formletter Framework представляет собой набор классов каждому из которых отведена определённая роль в организации процесса регистрации журналируемых событий. Эти классы предоставляют некоторый интерфейс для взаимодействия между собой, и реализуют базовую логику. Обычно для того, чтобы создать журнал и предоставить возможность им управлять разработчику необходимо унаследовать каждый из классов в реализации нового журнала, а также переопределить некоторые из методов для того, чтобы обеспечить доступ к данным и(или) внести специфику этого журнала.

Таким образом, интегрировав собственный код в исходный код базовой реализации Formletter Framework, можно отслеживать большое количество различных событий, происходящих в процессах журналирования, однако, стоит учитывать, что для некоторых журналов может возникнуть необходимость интегрироваться отдельно, ввиду возможной специфики конкретной реализации. Также для каждого журнала, как правило, определена уникальная структура данных, и при формировании сопутствующей информации к документу это придётся учитывать.

Как известно, документы на основе журналов не располагаются в файловой системе, а генерируются при необходимости. Эту проблему можно решить, обратившись к механизму генерации отчётов – SSRS для сохранения документа, например в формате pdf, локально. Такая функциональность предусмотрена в базовой поставке Microsoft Dynamics 365 [5].

СПИСОК ЛИТЕРАТУРЫ

1. Configure document management [Electronic resource] / Microsoft Docs. – 2019. – Mode of access: <https://docs.microsoft.com/en-us/dynamics365/fin-ops-core/fin-ops/organization-administration/configure-document-management>. – Date of access: 15.09.2019.
2. Document management [Electronic resource] / Microsoft Docs. – 2014. – Mode of access: <https://docs.microsoft.com/en-us/dynamicsax-2012/appuser-itpro/document-management>. – Date of access: 15.09.2019.
3. Working with General Journals [Electronic resource] / Microsoft Docs. – 2019. – Mode of access: <https://docs.microsoft.com/en-us/dynamics365/business-central/ui-work-general-journals>. – Date of access: 15.09.2019.
4. SQL Server Reporting Services (SSRS) reports that are available [Electronic resource] / Microsoft Docs. – 2018. – Mode of access: <https://docs.microsoft.com/en-us/dynamics365/fin-ops-core/dev-itpro/analytics/SSRS-report>. – Date of access: 15.09.2019.
5. Create reporting solutions [Electronic resource] / Microsoft Docs. – 2017. – Mode of access: <https://docs.microsoft.com/en-us/dynamics365/fin-ops-core/dev-itpro/analytics/create-nextgen-reporting-solutions>. – Date of access: 15.09.2019.

ОБ ОСНОВНЫХ АСПЕКТАХ РАЗРАБОТКИ СИСТЕМЫ НАКОПЛЕНИЯ И АНАЛИЗА ДАННЫХ ОБЪЕКТОВ ГОРОДСКОЙ СРЕДЫ

Неживинская А. Ю., Рудикова Л. В.

Кафедра современных технологий программирования, Гродненский государственный университет имени
Янки Купалы

Гродно, Республика Беларусь

E-mail: alina.nezhivinskaya@mail.ru, lada.rudikowa@gmail.com

В статье описываются общие подходы к разработке системы накопления и анализа данных объектов городской среды. Излагается концепция построения архитектуры системы. Выявлены основные требования архитектурного решения и определены инструменты, используемые для разработки.

ВВЕДЕНИЕ

В современном мире наблюдается отчётливая тенденция на продолжительное увеличение доли городского населения. По данным Всемирного Банка уже в 2007 году доля жителей городов составила большую половину населения планеты, и эта цифра постоянно растёт [1]. Учитывая факт, что город является местом сосредоточения огромного числа людей, стоит уделить особое внимание развитию современной городской среды. Поспособствовать решению данного вопроса могут современные технологии.

Всё большую популярность в сфере развития городов набирает концепция «умного города». Фундаментальной идеей данной концепции является внедрение информационных технологий во все сферы городского хозяйства, такие как промышленность, транспорт, системы здравоохранения, культуры и многие другие. Использование новейших технологий в данных областях позволяет поднять на более высокий уровень эффективность городских служб и, как следствие, повысить качество жизни городского населения.

В рамках теории «умного города» предполагается создание системы анализа и накопления данных объектов городской среды. Такая система позволит анализировать имеющиеся данные об объектах в окрестности города, а также накапливать новые данные. Пользователи данного приложения могут использовать данную информацию для улучшения собственного благосостояния и благосостояния города в целом.

I. ОБЩАЯ ХАРАКТЕРИСТИКА СИСТЕМЫ НАКОПЛЕНИЯ И АНАЛИЗА ОБЪЕКТОВ ГОРОДСКОЙ СРЕДЫ

Система накопления и анализа объектов городской среды предназначена для жителей города, желающих получить наглядную и систематизированную информацию в области городской среды.

Предлагаемое решение связано с созданием единой системы, которая позволит автоматически осуществлять сбор данных городской среды со сторонних источников, а затем анализировать

полученную информацию. Результаты анализа, произведённые системой, будут представлены в удобном для пользователя виде, а именно с помощью графиков, таблиц и схем.

Системой также будет предусмотрен просмотр объектов городской среды, поиск и фильтрация информации, проведение сравнительных характеристик городов и их районов, составление их рейтингов по странам и континентам. Главным функционалом разрабатываемой системы станет просмотр статистических данных уже хранящихся в системе, построение таблиц, наглядных графиков и схем.

Стоит отметить, что web-приложение для обработки объектов городской среды предполагает создание системы ролей.

II. ОСНОВНЫЕ ПОДХОДЫ РЕАЛИЗАЦИИ ПРИЛОЖЕНИЯ

Разрабатываемая система обработки объектов городской среды представляет собой веб-приложение, в основе которого лежит клиент-серверная архитектура. Система состоит из двух соответствующих компонентов: клиента и сервера, взаимодействие которых осуществляется посредством обмена файлами в формате JSON.

Клиентская часть приложения взаимодействует с пользователем и формирует запросы серверу посредством браузера. Серверная часть системы принимает запросы клиента и выполняет основную бизнес-логику приложения. Такая структура не зависит от конкретной операционной системы и позволяет сделать веб-приложение межплатформенным.

Система реализована в виде следующих уровней:

- уровень базы данных;
- уровень доступа к данным (dao);
- сервисный уровень;
- уровень, включающий бизнес-логику;
- уровень взаимодействия с пользователем.

Общая архитектура системы и связи между уровнями изображены на рисунке 1.

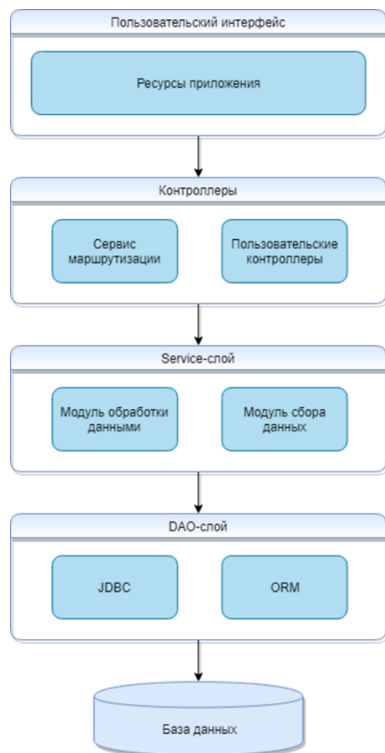


Рис. 1 – Многоуровневая архитектура системы

Каждый соответствующий слой отвечает за определённую часть логики приложения. База данных хранит все необходимые данные, связанные с городскими объектами. DAO-слой отвечает за предоставление выше лежащему слою контрактов для доступа к данным. Использование дополнительного уровня для доступа к данным делает систему расширяемой. Service-слой, или уровень бизнес-логики, содержит в себе всю логику приложения, связанную с манипуляцией над данными и реализацию алгоритмов. Уровень контроллеров принимает запросы клиентов и направляет к необходимому сервису для обработки и получения данных. Пользовательский интерфейс взаимодействует непосредственно с пользователем и отображает данные, полученные от сервера.

Реализации проекта может быть выполнена посредством следующих инструментов разработки:

- Язык программирования Java.
- Система управления базами данных MySQL и технология работы с базой данных Hibernate.
- Технология сборки проекта Maven.
- Фреймворк Spring.
- HTML, CSS.
- Фреймворк Angular.

Согласно рейтингу ТЮВЕ, на данный момент Java занимает первое место в рейтинге популярности языков программирования и за последние 15 лет всего несколько раз снижал свою позицию. Отличительными особенностями Java

являются быстрота, высокий уровень защиты и надёжность.

Angular представляет собой JavaScript-фреймворк с открытым исходным кодом от компании Google для создания клиентских приложений. Его цель — расширение браузерных приложений на основе MVC шаблона, а также упрощение тестирования и разработки.

Одной из основной функций системы является предоставление пользователю обработанных данных в наглядном виде. На рисунке 2 представлен пример интерфейса страницы визуализации результатов анализа данных в виде диаграммы.

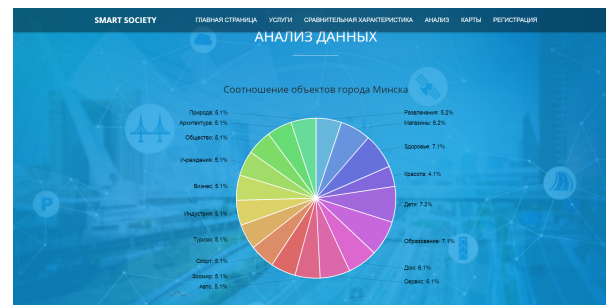


Рис. 2 – Интерфейс приложения. Страница результатов анализа данных

III. Выводы

С учетом всего вышесказанного, несомненно система накопления и анализа данных объектов городской среды будет востребована для жителей городов и управляющих органов, так как позволит им манипулировать имеющимися данными и использовать их для развития городской инфраструктуры. Обработанные данные могут быть полезными в решении вопросов озеленения территории города или размещения новых жилых комплексов. Информация, полученная системой, также может пригодиться туристам для выявления наиболее подходящих маршрутов для путешествия, абитуриентам при поиске учебных заведений и жителями города для выявления лучших районов для жизни и трудоустройства. Кроме того, данная система является расширяемой, доступной и простой в использовании.

СПИСОК ЛИТЕРАТУРЫ

1. The World Bank Data [Electronic resource] / United Nations Population Division. World Urbanization Prospects: 2018 Revision. – Mode of access: <https://data.worldbank.org>. – Дата доступа: 01.10.2019.
2. Неживинская, А. Ю. Общая концепция разработки Интернет-системы обработки объектов городской среды / А. Ю. Неживинская // Новые математические методы и компьютерные технологии в проектировании, производстве и научных исследованиях: материалы XXII Респ. науч. конф. студентов и аспирантов, Гомель, 25-27 марта 2019 г. / ГГУ им. Ф. Скорины – Гомель : ГГУ им. Ф. Скорины, 2019. – С.116.

ПРОГРАММА СБОРА ДАННЫХ О СТРУКТУРЕ ВЕБ-САЙТОВ

Потехин А. С., Александров А. А., Русецкий П. Н.

Кафедра информатики, Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: redrumofgod@gmail.com

В настоящее время всеобщие глобальные тенденции приближаются к тому, что все операции и торговые сделки будут проходить с использованием веб-ресурсов. Для того, чтобы успешно вести бизнес очень важно получать актуальные данные о движении рынка (динамика цен и товаров) и локальные новости, которые порой всецело влияют на формирование спроса, своевременно. Но необходимые данные не всегда легко доступны пользователю и чаще всего они не структурированы. Рассматривается приложение, которое будет обладать необходимым функционалом для сбора и структурирования данных с различных веб-ресурсов.

ВВЕДЕНИЕ

Анализ данных, представленных в Вебе, — распространенная на сегодняшний день исследовательская задача. Инструменты, позволяющие собирать данные для исследования из Веба, значительно развиты и позволяют не писать с нуля новые, но использовать и дорабатывать существующие [1, с.44]. Такие инструменты называются «веб-пауками» (web-spider), краулерами (web crawler) или скраперами (web scraper). В данной статье рассматриваются некоторые имеющиеся на рынке краулеры, обосновывается решение написания собственного проекта.

I. ОСНОВНАЯ ЗАДАЧА И ЕЕ ОГРАНИЧЕНИЯ

Целью исследования, требующего сбор данных из Сети Интернет, является сентимент-анализ данных с различных новостных сайтов. Данные должны содержать полную информацию о новости, включая заголовок, текст, дату и автора новости. Для того, чтобы обеспечить сбор указанной информации, необходимо реализовать инструмент — скрапер.

В широком понимании скрапер служит для сбора данных из различных интернет-ресурсов. Общий принцип его работы можно объяснить следующим образом: автоматизированный код выполняет запросы на целевой сайт и получая ответ, разбирает HTML-документ, ищет данные и преобразует их в заданный формат. Данные инструменты скрапинга позволяют вручную или автоматически извлекать новые или обновленные данные и сохранять их для последующего использования.

Для того чтобы выполнять эту задачу, инструмент должен поддерживать работу со следующими данными:

1. HTML, JavaScript, так как большинство сайтов построены с использованием этих технологий;
2. Plain text, PDF и другие форматы представления текстовых данных;

3. URLs, с возможностью построения на их основе графа веб-ресурсов.

Кроме того к краулеру предъявляется требование производительности. Согласно первичной оценке объема данных, которые предстоит собрать и обработать, нужно извлечь данные с 100 новостных сайтов, при условии того, что каждый из них в среднем выпускает по несколько статей в день, в худшем случае необходимо будет обработать около 108000 страниц. При средней скорости обработки 2 стр/сек, которую обеспечивают краулеры со средней производительностью [4], нам потребуется около 20 суток, чтобы собрать указанные данные. Такое время обработки повлечет увеличение длительности процесса сбора и анализа данных, поэтому для исследования требуется производительный и надежный краулер. Отдельным ограничением стоит отметить стоимость искомого инструмента, для исследовательских целей нами рассмотрены только решения с открытым исходным кодом, т.к. они распространяются бесплатно и их исходный код доступен для анализа и редактирования. Рассмотрим общий список требований, которым должен отвечать искомый инструмент.

II. ТРЕБОВАНИЯ К ИНСТРУМЕНТУ

Приступая к выбору инструмента, необходимо обозначить основные требования к краулеру [1],[2],[3], на основании которых можно было бы проводить их сравнительную оценку:

1. Надежность — Веб содержит ресурсы, которые могут вводить скрапер в бесконечный цикл или недоступные сервисы, ожидать выполнения которых, он не должен. Скрапер должен быть устойчивым к таким ловушкам;
2. Вежливость — интернет-ресурсы имеют явные и неявные политики, регулирующие частоту, с которой скрапер может посетить их. Они описаны в файле robots.txt и эти политики должны соблюдаться;

3. Распределенность – скрапер должен иметь возможность выполняться в распределенном режиме на нескольких машинах;
4. Масштабируемость – скрапер должен поддерживать возможность увеличения производительности за счет добавления дополнительных вычислительных узлов, на которых он исполняется;
5. Производительность и эффективность – скрапер должен обеспечивать эффективное использование системных ресурсов, включая процессор, память и полосу пропускания сети;
6. Качество – скрапер должен уметь отделять спам-страницы от полезных и извлекать последние;
7. Актуальность – скрапер должен поддерживать обновление собранных данных;
8. Расширяемость – скрапер должен быть модульным, т.е. позволять добавлять новую функциональность, для анализа новых форматов данных, протоколов и т.д.

Помимо описанных общих требований для скраперов можно выделить основные требования для нашей задачи исследования:

1. Скрапер должен быть кроссплатформенным, чтобы его можно было одинаково настраивать и конфигурировать на вычислительных узлах с разными операционными системами;
2. Скрапер должен обеспечивать производительность обработки порядка 100 стр/сек, чтобы время сбора описанного выше объема данных составляло часы, а не дни. В том случае если окажется, что данных для сбора и анализа больше предполагаемого, скрапер должен предоставлять возможность легко увеличить его производительность путем выделения ему для работы большего числа потоков или добавления дополнительных вычислительных узлов;
3. Скрапер должен быть интегрирован с базой данных для хранения собранной информации и полнотекстовым индексом, позволяющим быстро извлекать данные для последующего анализа, отвечающие указанным условиям;
4. Требуется скрапер для сбора данных в ширину и вертикального поиска, так как в указанной задаче необходимо извлечь информацию о конкретной предметной области, а не узкое множество фактов;

III. ОБОСНОВАНИЕ ВЫБОРА СКРАПЕРА

Исходя из нескольких источников информации, дающих представление относительно того или иного продукта с открытым исходным кодом, не было найдено ни одного решения, в полной мере подходящей для данной задачи исследования. Многие из скраперов перестали поддерживаться сообществом и развиваться, многие не являются кросс-платформенными, не обеспечивают необходимую скорость скачки или являются проблематичными для настройки. Также были рассмотрены облачные сервисы, но для решения данной задачи они не подходят, так как не рассчитаны на хранение и обработку большого количества данных, на ограниченных вычислительных ресурсах или не имеет бесплатной опции доступа.

В результате проведенного анализа было принято решение реализовать свой собственный скрапер, который будет являться эффективным инструментом для поиска в Вебе, ядро будет написано на C++ с которым взаимодействует Ruby-оболочка, будет поддерживаться граф связи узлов, различные парсеры, фильтры и нормализаторы URL. Он будет позволять использовать различные хранилища данных, такие как Cassandra, Hbase и др. Скрапер также будет являться масштабируемым (до 100 узлов в кластере) и легко настраивается и расширяться, в полной мере являться «вежливым».

ЗАКЛЮЧЕНИЕ

В данной работе рассмотрены требования и основные типы скраперов, которые на сегодняшний день выделяют в литературе. Исходя этого не были найдены скраперы, удовлетворяющие условиям реализации исследовательского проекта и в результате было принято решение реализовывать свой собственный скрапер.

СПИСОК ЛИТЕРАТУРЫ

1. PAPAVALASSIOLIOU V., PROKOPIDIS P., THURMAIR G. A modular open-source focused crawler for mining monolingual and bilingual corpora from the web // Proceedings of the 6th Workshop on Building and Using Comparable Corpora. – 2013.
2. ANUJA M.S., BAL J.S., VARNICA Web Crawler: Extracting the Web Data // International Journal of Computer Trends and Technology. – 2014.
3. YADAV M., GOYAL N. Comparison of Open Source Crawlers-A Review // International Journal of Scientific Engineering Research. – 2015.
4. Performance Benchmark [Электронный ресурс]. URL: <https://geekflare.com/web-scraping-frameworks/> (дата обращения: 03.06.2019)

АНАЛИЗ СУЩЕСТВУЮЩИХ ПОДХОДОВ РАСПОЗНАВАНИЯ ЛИЦ

Румысов В. С.

Факультет информационных технологий и управления, Белорусский государственный университет
информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: rvlad121096@gmail.com

Рассмотрен ряд различных подходов для решения задачи распознавания лиц на изображениях. Выявлены преимущества и недостатки рассмотренных методов. Распознавание образов; метод Виолы-Джонса; распознавание изображений при помощи нейронных сетей; распознавание; идентификация.

ВВЕДЕНИЕ

Задача распознавания лиц имеет множество применений в таких областях, как организация видеоконференций, системы машинного зрения, системы безопасности и контроля доступа и т.д. Основной трудностью данной задачи является зависимость качества результата распознавания человека по изображению лица от ракурса, положения, условий освещения, при наличии дефор-

мации, скрытых частей, поворотов относительно осей и прочих факторов, влияющих на качество распознавания образов. Далее будут рассмотрены и проанализированы современные методы распознавания лиц на изображениях.

Несмотря на большое разнообразие представленных методов, можно выделить общую структуру процесса распознавания лиц (см. рис. 1).



Рис. 1 – Общий процесс обработки изображения лица при распознавании

На первом этапе производится детектирование и локализация лица на изображении. На этапе распознавания производится выравнивание изображения лица (геометрическое и яркостное), вычисление признаков и непосредственно распознавание – сравнение вычисленных признаков с заложенными в базу данных эталонами. Основным отличием всех представленных алгоритмов будет вычисление признаков и сравнение их совокупностей между собой.

Существующие методы распознавания лиц:

1. метод гибкого сравнения на графах (Elastic graph matching);
2. нейронные сети;
3. скрытые Марковские модели (СММ, НММ);
4. метод главных компонент или principal component analysis (PCA);
5. и др.

Рассмотрим некоторые из перечисленных методов для распознавания лиц на изображениях, а также разберем их преимущества и недостатки.

I. НЕЙРОННАЯ СЕТЬ

В настоящее время существует около десятка разновидностей нейронных сетей (НС). Одним из самых широко используемых вариантов является сеть, построенная на многослойном перцептроне, которая позволяет классифицировать поданное на вход изображение/сигнал в соответствии с предварительной настройкой/обучением сети.

Обучаются нейронные сети на наборе обучающих примеров. Суть обучения сводится к настройке весов межнейронных связей в процессе решения оптимизационной задачи методом градиентного спуска. В процессе обучения нейронной сети происходит автоматическое извлечение ключевых признаков, определение их важности и построение взаимосвязей между ними. Предполагается, что обученная нейронная сеть сможет применить опыт, полученный в процессе обучения, на неизвестные образы за счет обобщающих способностей.

Наилучшие результаты в области распознавания лиц (по результатам анализа публикаций) показала Convolutional Neural Network или сверточная нейронная сеть, которая является логическим развитием идей таких архитектур нейронных сетей как когнитрона и неокогнитрона. Успех обусловлен возможностью учета двумерной топологии изображения, в отличие от многослойного перцептрона.

Отличительными особенностями сверточной нейронной сети являются локальные рецепторные поля (обеспечивают локальную двумерную связность нейронов), общие веса (обеспечивают детектирование некоторых черт в любом месте изображения) и иерархическая организация с пространственным сэмпингом (spatial subsampling). Благодаря этим нововведениям сверточная нейронная сеть обеспечивает частичную устойчивость при распознавании изображений к изменениям масштаба, смещениям, поворотам, смене ракурса и прочим искажениям.

Тестирование сверточной нейронной сети на базе данных ORL, содержащей изображения лиц с небольшими изменениями освещения, масштаба, пространственных поворотов, положения и различными эмоциями, показало 96% точность распознавания.

Основное преимущество данного подхода распознавания лиц – сверточная нейронная сеть, которая является логическим развитием идей таких архитектур нейронной сети как когнитрона и неокогнитрона позволяет получить 96% точность распознавания. Успех обусловлен возможностью учета двумерной топологии изображения [1].

К недостаткам нейронных сетей относятся:

- добавление нового эталонного лица в базу данных требует полного переобучения сети на всем имеющемся наборе, а это достаточно длительная процедура (в зависимости от размера выборки от 1 часа до нескольких дней);
- проблемы математического характера, связанные с обучением: попадание в локальный оптимум, выбор оптимального шага оптимизации, переобучение и т.д;
- трудно формализуемый этап выбора архитектуры сети (количество нейронов, слоев, характер связей).

Обобщая все вышераассмотренное, можно сделать вывод, что нейронная сеть – «черный ящик» с трудно интерпретируемыми результатами работы.

II. МЕТОД ВИОЛЫ-ДЖОНСА

Метод Виолы-Джонса был разработан и представлен в 2001 году Полом Виолой и Майклом Джонсом, и до сих пор является основополагающим для поиска объектов на изображении в реальном времени.

Основные принципы, на которых основан метод, таковы:

- используются изображения в интегральном представлении, что позволяет вычислять быстро необходимые объекты;
- используются признаки Хаара, с помощью которых происходит поиск нужного объекта (в данном контексте, лица и его черт);
- используется бустинг (от англ. boost – улучшение, усиление) для выбора наиболее подходящих признаков для искомого объекта на данной части изображения;
- все признаки поступают на вход классификатора, который даёт результат «верно» либо «ложь»;
- используются каскады признаков для быстрого отбрасывания окон, где не найдено лицо.

Обучение классификаторов идет очень медленно, но результаты поиска лица очень быстры, именно поэтому был выбран данный метод распознавания лиц на изображении. Виола-Джонс является одним из лучших по соотношению показателей эффективности распознавания/скорость работы. Также этот детектор обладает крайне низкой вероятностью ложного обнаружения лица. Алгоритм даже хорошо работает и распознает черты лица под небольшим углом, примерно до 30 градусов. При угле наклона больше 30 градусов процент обнаружений резко падает. И это не позволяет в стандартной реализации детектировать повернутое лицо человека под произвольным углом, что в значительной мере затрудняет или делает невозможным использование алгоритма в современных производственных системах с учетом их растущих потребностей.

Для большинства систем распознавания лиц основной задачей является сравнение полученного изображения лица с набором изображений лиц из базы данных. Характеристики систем распознавания лиц в этом случае оцениваются путем определения вероятностей ошибочного отказа в распознавании (для изображения лица, присутствующего в базе, принимается решение, как о неопознанном лице) и ошибочного распознавания.

Учитывая все вышеизложенное, для повышения качества работы, актуальным может являться создание гибридных методов, использующих преимущества и недостатки рассмотренных подходов [2].

СПИСОК ЛИТЕРАТУРЫ

1. Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, Lior Wolf. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. – 2014 – IEEE Conference on Computer Vision and Pattern Recognition – Columbus, OH, USA.
2. Анализ существующих подходов к распознаванию лиц [Электронный ресурс] / Хабр. – 2014. – Режим доступа: <https://habr.com/ru/post/133826/>. – Дата доступа: 10.08.2019.

МЕТОДЫ ЗАЩИТЫ АВТОРСКОГО ПРАВА НА ПРОГРАММНЫЕ ПРОДУКТЫ С ПОМОЩЬЮ ВОДЯНЫХ ЗНАКОВ И ОТПЕЧАТКОВ ПАЛЬЦЕВ

Шулицкий Д. С., Водейко А. Э.

Кафедра программного обеспечения информационных технологий, Белорусский государственный университет информатики и радиоэлектроники

Минск, Республика Беларусь

E-mail: dmitrii.shulitskii@gmail.com, avodeikoga@gmail.com

Защита цифровых работ от нелегального использования и распространения является сложной задачей в информационном обществе. Полное предотвращение незаконного использования работ недостижимо за разумную цену. Большинство схем защиты авторского права отпугивают людей от нелегального использования и распространения цифрового контента тем, что позволяют обнаружить незаконное использование. Для этого в оригинальную работу добавляют идентификационную информацию с помощью техники водяных знаков или отпечатков пальцев.

ВВЕДЕНИЕ

Цифровой водяной знак – это информация, внедрённая в цифровую работу, позволяющая автору доказать своё авторство. В самом простом случае, водяной знак представляет копирайт-строку. Цифровой отпечаток пальца – водяной знак, содержащий информацию не только о правообладателе, но и о субъекте, которому предоставлено право использования данного экземпляра объекта интеллектуальной собственности.

Технология цифровых водяных знаков давно используется для защиты прав интеллектуальной собственности на графические изображения, видеозаписи, фонограммы. В последние годы происходят попытки использования цифровых водяных знаков для защиты прав собственности на программное обеспечение.

I. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ

Рассмотрим математическую запись для формального описания наиболее важных понятий, касающихся внедрения водяных знаков в программное обеспечение. Пусть O – компьютерная программа, допускающая выполнение с ней различных манипуляций, S – текущее состояние вычислительной системы. $S[O, \dots]$ означает состояние вычислительной системы S , в котором существует только одна копия программы O . $S'[O, O, \dots]$ – состояние системы S' , в котором существует 2 идентичные копии O . Пусть ω – водяной знак, а E – функция, описывающая внедрение водяного знака в программу, тогда

$$E(S, \omega) = S_\omega,$$

где $S[O, \dots]$ – состояние вычислительной среды, содержащий объект O , в который внедряется водяной знак, ω – желаемый водяной знак. Обозначим как $S_\omega = [O_\omega, \dots]$ состояние вычислительной системы, содержащей объект с внедрённым водяным знаком. Соответствующая E функ-

ция извлечения водяного знака R имеет следующее свойство:

$$\forall S_\omega : R(S_\omega) = \omega.$$

Ложное распознавание водяного знака нежелательно, поэтому:

$$\forall S_\omega, \omega' \neq \omega : R(S_\omega) \neq \omega'.$$

Конкретный алгоритм водяного знака состоит из функции внедрения и функции распознавания водяного знака. [1]

II. ВИДЫ ВОДЯНЫХ ЗНАКОВ В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Если функция извлечения водяного знака общедоступна, водяной знак называют видимым. Если эта функция доступна только лицу, внедрившему водяной знак, его называют невидимым.

Водяные знаки в программном обеспечении можно разделить на водяные знаки данных и водяные знаки кода. Водяные знаки данных размещаются в структурах данных программы, которые не используются. Водяные знаки кода внедряются при помощи манипуляций с инструкциями микропроцессора.

Статические водяные знаки извлекаются непосредственно из файла программы. Для извлечения динамического водяного знака необходим запуск программы и получение результата её работы. [2]

III. ОТПЕЧАТКИ ПАЛЬЦЕВ В ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Цифровые отпечатки пальцев, в отличие от обычных водяных знаков, содержат не только информацию об авторе программы, но и информацию о покупателе программы. Это позволяют различать копии одной и той же программы. При продаже очередной копии программы, формируется содержимое отпечатка пальца. Далее

эта информация зашифровывается при помощи частного ключа разработчика. Это позволяет доказать, что информация была зашифрована конкретным лицом. Затем отпечаток пальца внедряется в программное обеспечение. Отпечатки пальца позволяют обнаруживать источники нелегального распространения программы, так как не составляет труда определить покупателя конкретной копии программы.

К цифровым отпечаткам пальцев предъявляется дополнительное требование по сравнению с водяными знаками: даже если атакующий имеет доступ к некоторому количеству копий программы, он не должен иметь возможность прочесть, повредить или удалить водяной знак на основе информации, полученной путём сравнения копий.

IV. ВОДЯНЫЕ ЗНАКИ НА ОСНОВЕ ИНВАРИАНТНЫХ СТАТИСТИЧЕСКИХ ХАРАКТЕРИСТИКАХ КОНТЕЙНЕРА

Защищаемый водяным знаком объект состоит из множества атомарных элементов. Для растровых изображений такими элементами являются пиксели, для программ – низкоуровневые инструкции. Множество элементов, составляющих объект, обладают некоторыми статистическими характеристиками. Некоторые характеристики инвариантны, то есть не зависят от содержания объекта. Такие характеристики можно использовать для внедрения особого вида водяного знака, называемого признаком авторства. Признак авторства позволяет ответить на вопрос «Является ли рассматриваемый субъект автором данного объекта?».

Рассмотрим величину $S = a_i - b_i$, равную разности яркости двух случайно выбранных пикселей изображения. Эксперименты показывают, что для достаточно большой выборки пар пикселей значение выражения:

$$S_n = \sum_{i=1}^n (a_i - b_i)$$

будет очень близким к нулю. Данная характеристика не зависит от содержимого изображения. Авторы «Pathwork» предлагают следующий алгоритм внедрения признака:

1. При помощи генератора псевдослучайных чисел выбираются две точки изображения со значениями яркости (a_i, b_i) ;
2. значение яркости a_i увеличивается на величину δ ;
3. значение яркости b_i уменьшается на величину δ ;
4. шаги 1-3 повторяются n раз.

После выполнения модификации, новое значение S'_n увеличится на величину $2\delta n$. Величина δ выбирается такой, чтобы модификации не были различимы человеческим глазом. Для того, чтобы воспроизвести псевдослучайную последовательность координат пикселей, необходимо знать начальное состояние генератора псевдослучайных чисел. Оно и является ключом внедрения признака авторства и держится в секрете. Субъект, претендующий на авторство объекта, предоставляет начальное состояние генератора псевдослучайных чисел, на основе которого генерируются пары пикселей. Авторство подтверждается, если значение характеристики S_n для данного набора пикселей значительно отличается от нуля.

V. СЛЕПАЯ ПРОВЕРКА ВОДЯНЫХ ЗНАКОВ

Концептуальной проблемой обычных водяных знаков является то, что демонстрация присутствия водяного знака как свидетельства раскрывает чувствительную информацию, которая может быть использована для удаления водяного знака. Желательно убедить верификатора в том, что водяной знак присутствует и не раскрыть ему информацию, которая может помочь удалить водяной знак.

Одним из подходов к решению данной проблемы является использование слепой проверки водяных знаков. Слепые протоколы позволяют убедить верификатора, что правообладатель знает секретное значение, и верификатор не узнаёт ничего нового о секретных данных правообладателя.

Например, правообладатель может создать настоящий водяной знак и спрятать его в большом списке поддельных водяных знаков. Затем он предлагает верификатору обнаружить все водяные знаки и доказывает, что, по крайней мере, один из водяных знаков является настоящим, не раскрывая, какой именно. Безопасность данного метода основана на том, что количество водяных знаков в списке должно быть на столько большим, что невозможно удалить их все без серьёзного ухудшения стего-данных. [3]

1. Криптография, стеганография и охрана авторского права / В.Н.Ярмолик, С.С.Портянко, С.В.Ярмолик / Издательский центр БГУ – Минск, 2007. – С. 195–211
2. Watermarking, Tamper-Proofing and Obfuscation – Tools for Software Protection / C. Collberg, C. Thomborson – Department of Computer Science University of Arizona, 2000. – P. 7-11.
3. Zero-Knowledge Watermark Detection and Proof of Ownership / A. Adelsbach, A. Sadeghi – LNCS, 2001. – P. 273-288.

РАЗРАБОТКА ВЕБ ПРИЛОЖЕНИЯ ДЛЯ УПРАВЛЕНИЯ БИЗНЕС ПРОЦЕССАМИ С ИСПОЛЬЗОВАНИЕМ SAP UI 5

Ясницкий С. В.

Кафедра интеллектуальных систем, Факультет радиопизики и компьютерных технологий,
Белорусский государственный университет
Минск, Республика Беларусь
E-mail: yasnickistas@gmail.com

Ключевые слова: веб-приложения, пользовательский интерфейс, бизнес процессы, SAP, SAP UI 5, FIORI, пользовательский опыт, SAP WEB IDE. Объект исследования – инструменты для разработки веб-приложений SAP WEB IDE, SCP, SAP Build Me, HTML5, JavaScript, основные алгоритмы разработки веб-приложений SAP UI 5/FIORI. Цель работы – разработать веб-приложения для управления бизнес процессами на базе SAP UI 5, с использованием инструментов SAP UI 5, SAP Fiori Guidelines, HTML5, JavaScript, Chrome DevTools.

ВВЕДЕНИЕ

В настоящее время веб-приложения самый быстрый и удобный способ предоставить требуемые функции пользователю. В их использовании есть большой ряд преимуществ, таких как: независимость от операционной системы пользователя, доступность везде где есть сеть интернет, адаптивность дизайна. Так же разработка таких приложений с использованием сервисов, предлагаемых компанией SAP позволяет не устанавливать дополнительное программное обеспечение, протестировать и запустить приложение в интернет браузере. Сегодня почти каждое программное обеспечение и веб-сайты имеют мобильную совместимую версию, и каждый может проверить что угодно на своем мобильном телефоне или планшете. GUI - это графический интерфейс пользователя, который был выпущен в базовой функциональности более 15 лет назад. SAP признала, что программы и потребности пользователей изменились с момента распространения смартфонов и приложений. Простое и интуитивно понятное управление обеспечивает лучшее взаимодействие с пользователем. Таким образом, для SAP было очень важно найти решение, и это был SAP Fiori - <<Один UX для всех продуктов SAP>> [1]. На протяжении многих лет SAP является ведущим мировым поставщиком клиент-серверных бизнес-решений. На данный момент актуальным является появление конфигурационных приложений в которые интегрирована бизнес-логика и бизнес-объекты SAP.

I. FIORI

Это новая технология пользовательского интерфейса SAP с современным дизайном для нового пользователя. Оригинальный пользовательский интерфейс SAP Fiori для веб-приложений, основанный на платформе SAPUI5. SAP Fiori Guidelines позволяет узнать, как создавать привлекательные и интуитивно понятные приложения, которые могут работать на любом

устройстве [3]. SAP Fiori - это язык проектирования, который обеспечивает удобство работы с корпоративными приложениями. На основе ролей пользователей и бизнес-процессов SAP Fiori упрощает ведение бизнеса. SAP Fiori - это смена парадигмы от монолитных ERP-решений к более легким приложениям, адаптированным к задачам пользователей. Чтобы ускорить трансформацию мировой цифровой экономики, SAP применяет этот язык проектирования для ведущих технологических платформ. Создан для пользователя, его потребностей и того, как он работает. Адаптируется к нескольким случаям использования и устройствам. Приложение включает в себя только то, что необходимо. Это веб-приложения, которые можно запускать не только на любом настольном компьютере, но и на всех современных планшетах и смартфонах и в любом браузере.[4] Ценности и принципы дизайна заключаются в следующем - SAP Fiori устанавливает стандарт для корпоративного пользовательского интерфейса, устраняя ненужные сложности. Разработчик дает пользователям возможность контролировать свои бизнес-задачи, предоставляя им только то, что им действительно нужно. Язык дизайна SAP Fiori помогает дизайнерам создавать согласованные и согласованные продукты. Он выражает ценности и приоритеты дизайна независимо от технологии или области. Элементы SAP Fiori предоставляют проекты для шаблонов пользовательского интерфейса и предопределенные шаблоны для общих случаев использования приложений. Разработчики приложений могут использовать элементы SAP Fiori для создания приложений SAP Fiori. Fiori основан на платформе, известной как SAPUI5, которая построена на основе HTML5 и совместима с любым устройством и экраном любого размера. Первым выпуском были транзакционные приложения Fiori для наиболее распространенных бизнес-функций, таких как задачи самообслуживания. Клиенты и поставщики услуг SAP также могут разрабатывать свои собственные приложения Fiori.

II. OPENUI5 и SAPUI5

OpenUI5 - это библиотека пользовательского интерфейса JavaScript с открытым исходным кодом, а SAPUI5 - это версия, которая может использоваться клиентом SAP с определенной лицензией. Использование SAPUI5 также бесплатно для клиентов, но они платят за другие продукты SAP. Наряду с SAPUI5 SAP предлагает современные и удобные инструменты для работы и создания веб-приложений. SAPUI5 (SAP User Interface HTML5) - это фреймворк для разработки пользовательских интерфейсов на базе кроссбраузерной JavaScript библиотеки. Ключевыми особенностями этой технологии являются: Документированный API, Расширяемые UI компоненты, Высокая производительность, Гибкие возможности по оформлению (дизайнер тем, прямая правка CSS), Возможность использования Ajax, Использование библиотеки jQuery. SAPUI5 позволяет создавать современные пользовательские интерфейсы как на базе стандартных компонентов, так и на базе собственных. WEB IDE – среда разработки и тестирования в браузере, облачные технологии для запуска клиентских приложений, создание структур на основе шаблонов, что значительно облегчает разработку.

III. РАЗРАБОТКА

Различают транзакционные приложения, аналитические приложения и информационные. На протяжении многих лет SAP становится ведущим мировым поставщиком клиент-серверных бизнес-решений. На данный момент актуальным является появление конфигурационных приложений в которые интегрирована бизнес-логика и бизнес-объекты SAP. Проанализирован алгоритм решения проблемы достижения заказчиком и разработчиком единого видения проекта, оценки требуемых работ и бюджета проекта, сформированы требования к приложению. Согласно алгоритму был разработан и запущен пример кор-

поративного приложения SAP UI 5 для управления бизнес процессами. Пройдены все этапы - формирование задания для решения проблемы реализации бизнес логики посредством веб приложения, выбраны необходимые инструменты для разработки, спроектирована структура приложения, выделен необходимый функционал приложения, протестированы прототипы, проведена непосредственно разработка с использованием сервисов SAP такие как SAP WEB IDE, SCP, SAP Build Me, изучены концепции и использованы инструменты SAP UI 5, SAP Fiori Guidelines, HTML5, JavaScript, разработанное приложение протестировано с помощью Unit Test, OPA Test, развернуто на облачной платформе для доступа потребителя.

ЗАКЛЮЧЕНИЕ

Таким образом разработка веб-приложений с использованием данных инструментов, является перспективной областью для предоставления клиенту современных и удобных решений связанных с бизнес логикой компании SAP. Использование предлагаемых инструментов упрощает процесс разработки за счет встроенных библиотек содержащих логику объектов SAP, использования современных концепций и языков программирования, что является быстроразвивающейся областью, а так же минимизирует время обучения и возможности ошибок разработчика, упрощается и запуск приложения с использованием облачных технологий для быстрого и независимого доступа к приложению через интернет.

IV. СПИСОК ЛИТЕРАТУРЫ

1. Rohan Ahmed, Sapui5 and Fiori. Status and Future Perspective / Grin Verlag, 2018.
2. <https://sapui5.hana.ondemand.com/> «SAP UI 5 SDK» - Документация SAPUI5.
3. SAP Fiori Implementation and development / Anil Bavaraju, 2017, ISBN 978-1-4932-1542-3.
4. SAPUI5 and SAP Fiori: The Psychology of UX Design / Paul Modderman, 2017, ISBN 978-1-4932-1618-5.

Содержание

Пленарные доклады	7
Заливако С. С., Иванюк А. А. «Физически неклонированные функции»	8
Яцукевич В. В. «Индустрия игр Беларуси»	22
Секция «Системы управления»	25
Акинфина М. А., Бондаренко С. П. «Система учета работы банковского оборудования» ...	26
Алешко Е. Ю., Хаджинов М. К. «Алгоритмическое решение обратной задачи динамики для подавления колебаний»	28
Amirzhanova Z. V., Mashtaeva A. A., Satybaldina D. K. «Development of a robust aircraft control system in conditions of disturbances»	30
Ахметова С. О. «Робастная устойчивость системы с одним входом и одним выходом в классе катастроф «ласточкин хвост»	32
Бысов С. А., Ивановская Я. М. «Аппаратно-программный комплекс учёта показателей воздушной среды»	34
Волков А. В. и др. «Анализ и разработка модулей учебного курса по программированию» ...	36
Деменковец Д. В., Дубицкая К. Ю., Борисевич А. Н. «Программное средство автоматизации процесса динамического взвешивания железнодорожных вагонов с распознаванием номеров на основе искусственной нейронной сети»	38
Железничев А. И., Доманов А. Т. «Анализ производительности технологического участка вакуумной установки»	40
Imad A. Elzein , Yury Petrenko «Analysis of Model Predictive Control through a power converter in a renewable energy system»	42
Кожевников М. М., Илюшин И. Э., Юркина А. А., Демиденко Е. Ю. «Планирование траекторий промышленного робота-манипулятора Fanuc710iC с использованием нейронных сетей»	44
Kalmaganbetova Zh. A., Uashov E. K., Satybaldina D. K. «Synthesis of robust tracking control system under perturbation conditions»	46
Кива В. С., Брановицкий А. А. «Распределенная трассировка»	48
Конилов А. Д., Самаль Д. И. «Система точного позиционирования низкоскоростных минироботов»	50
Конончик О. Н. «Web-приложение для автоматизации процесса обработки заказов»	52
Котковец А. А., Поттосина С. А. «Об агрегированном подходе выбора команд IT-проектов в системах принятия решений»	54
Козловская Е. А., Рудикова Л. В. «Об основных аспектах разработки интернет-приложения «Online gallery»	56
Кузнецов В. В., Марко А. Ф. «Управление мехатронными системами перемещений с использованием технологии EtherCAT»	58
Куль В. А., Курулёв А. П., Островский В. А. «Повышение надежности автомобилей на этапе проектирования»	60
Лапко А. В. «Особенности классификации дефектов полупроводниковой пластины при автоматизированном контроле»	62

Лукьянец С. В., Снисаренко С. В. «Имитационное моделирование гибкого штамповочного производства»	64
Малашенков А. А. «Датчик-индикатор состояния электрической сети и потребителя электрической энергии для автоматизированной системы управления»	66
Марков А. Н., Савицкая Д. Г., Парамонов А. И. «Анализ распределения нагрузки учебных лабораторий УО БГУИР»	68
Нестеренков С. Н., Кукареко А. В. «Применения цифровых двойников и бессерверной архитектуры для управления IoT устройствами»	70
Новик И. П. «Оркестрация Docker контейнеров: концепция Zero Downtime Deployment»	72
Пархоменко И. Н., Господ А. В. «Методика программной идентификации параметров наблюдаемого объекта»	76
Парамонов А. И., Труханович И. А. «Подходы к решению задачи автоматизированного составления экскурсионных туров»	76
Пашук А. В., Гуринович А. Б., Кузнецов А. П., Смирнов В. Л. «Модификация метода разрешения лексической многозначности в области биомедицины»	78
Рахимов Р. В. «Повышение надежности электроснабжения потребителей Республики Казахстан путём внедрения цифровых технологий»	80
Рубель П. А. «Автоматизация процесса запуска автотестов с помощью грид системы»	82
Савик К. В. «Анализ метода сканирования ЛВС»	84
Сманцер М. С., Андрухович А. К. «Автоматизация учёта акцизной продукции для ЕГАИС»	86
Струц Д. А. «Обзор и анализ алгоритмов диспетчеризации группы лифтов»	88
Тарасюк Е. В. «Отображения тепловых источников на карте местности, основанное на методе выделения объектов с тепловым излучением на статических RGB изображениях»	90
Тимофеев А. М. «Влияние надежности легитимного приемного оборудования на вероятность ошибочной регистрации данных в квантово-криптографических каналах связи»	92
Ульянко В. Г., Сечко Г. В. «Создание программных средств для проектирования и управления разработкой программных продуктов»	94
Хаджинов М. К., Доманов А. Т., Павлова А. В. «Сервопривод с комбинированным управлением»	96
Черемисинова Л. Д., Черемисинов Д. И. «Построение сокращенного графа достижимых состояний параллельного алгоритма»	98
Секция «Интеллектуальные информационные технологии» .	101
Брановицкий А. А., Борисов Д. В. «Система автоматического распознавания эмоций по речевому сигналу»	102
Давлатов Ш. Р. «Разработка универсального веб-скрапера на базе фреймворка NestJS и браузера Chromium Headless»	104
Давыденко И. Т., Шункевич Д. В., Синельников П. М., Трунц В. В. «Подходы к интеллектуализации средств автоматизации процесса разработки компьютерных систем»	106
Галиевская Е. П. «Разработка модели системноонтологического реестра мульти-агентской корпоративной информационной системы»	108
Ивашенко В. П. «Распределение памяти в неограниченном линейном адресном пространстве»	110

Искра Н. А., Межень А. Л., Шункевич Д. В. «Онтология предметной области пространственных сущностей для системы семантического анализа изображений»	112
Криштапович А. С., Адуцкевич И. А. «Анализ вакансий для исследования регионального рынка труда»	114
Ли Вэньцзу, Цянь Лунвэй «Семантические модели и средства контроля знаний пользователей обучающих систем»	116
Ляпко А. М., Курочкин А. В. «Библиотека для создания экспертных систем на базе алгоритмов нечеткого вывода на языке TypeScript»	118
Цянь Лунвэй, Ли Вэньцзу «Онтологический подход к обработке текстов китайского языка»	120
Романов В. И. «Программная поддержка базы знаний в системе логического проектирования FLC-2»	122
Сафин К. В. «Автоматическое разрешение конфликтов при синхронизации реплик в распределённой системе»	124
Сиротко С. И., Фисько Д. В. «Использование дистилляции между слоями нейронной сети для улучшения качества сегментации дорожной разметки»	126
Хурсов П. С., Искра Н. А. «Алгоритмы детекции объектов для анализа изображений»	128
Секция «Проектирование встраиваемых систем»	131
Бибило П. Н. «Экспериментальное исследование алгоритма выделения подсистем «связанных» булевых функций»	132
Боровский М. А., Грабко А. Н. «Формат POSIT как формат представления числовых значений в нейронных сетях на устройствах программируемой логики»	134
Видничук В. Н., Иванюк А. А. «Лексическая обфускация как способ внедрения водяных знаков в исходные коды программ и проектных описаний»	136
Леванцевич В. А. «Псевдоисчерпывающее тестирование ОЗУ встраиваемых систем»	138
Кардаш С. Н. «Алгоритм дизъюнктивного разложения системы полностью определенных булевых функций»	140
Кириенко Н. А. «Устранение иерархии транзисторных описаний логических схем»	142
Pottosin Yu. V. «On bi-decomposition of incompletely specified Boolean functions»	144
Станкевич А. В., Петровский Н. А., Качинский М. В. «Процессор хэш-функции Salsa20/8»	146
Черемисинов Д. И., Черемисинова Л. Д. «Извлечение логических уравнений из описания КМОП схемы на SPICE»	148
Шамына А. Ю., Иванюк А. А. «Анализ симметричных путей физически неклонированной функции на FPGA»	150
Секция «Математическое моделирование и компьютерная графика»	153
Амельченко Н. П., Шаронова Е. И., Шаронова О. А. «Концепция компьютерного моделирования паводочного стока»	154
Ващилко А. П. «Применение дескрипторов при написании транспайлера на примере транспайлера из C# в Dart»	156
Видничук В. Н., Альшевский А. В., Бысов С. А. «Программно-аппаратная система захвата движений рук»	158

Головатая Е. А., Садов В. С. «Оценка качества результатов алгоритмов трехмерной реконструкции сцен»	160
Грабко А. Н., Боровский М. А. «О некоторых методах классификации (псевдо)риманновых однородных пространств с помощью систем компьютерной математики»	162
Можей Н. П. «Однородные пространства со связностями ненулевой кривизны»	164
Нестеренков С. Н., Наливко В. Н. «Применение генетического алгоритма для решения задач многомерной оптимизации»	166
Панкевич Д. С., Кривоносова Т. М. «Неклассическая игра Змейка на OpenGL»	168
Россолова К. С., Бычков П. А., Шилин Л. Ю. «Техническая реализация и дидактическая ценность разработанной логопедической игры»	170
Рудикова Л. В., Колосов А. А. «Библиотека для параллельного обучения нейронных сетей в мультипроцессорных системах с помощью метода роя частиц»	172
Сманцер Д. С. «Моделирование переходных процессов в электрических цепях методом переменных состояния посредством Python»	174
Таранчук В. Б. «Примеры интеллектуального анализа и визуализации геоданных»	176
Турец Е. А., Кукин Д. П. «Алгоритмы моделирования динамики твердого тела»	178
Чистова К. О., Гуринович А. Б. «Прогнозирование погоды с помощью цепей Маркова» ...	180
Секция «Системы обработки информации»	183
Аврамов В. В., Рыбенков Е. В., Петровский Н. А. «Снижение уровня спекл-шума в цифровых изображениях на основе 2-D неразделимого Q-ПУБФ с использованием пороговой нейронной сети»	183
Адерейко А. Д., Жук Е. Е. «Выбор типа интервенции нестационарного временного ряда по заданным шаблонам»	180
Андрухович А. К., Герман Ю. О., Сманцер М. С. «Проблема промежуточной синхронизации данных в многоступенчатом процессе разработки современного радиоэлектронного изделия»	182
Богдан А. А. «Использование бессерверных вычислений для решения задач обработки информации»	184
Богдан А. А., Лапицкая Н. В. «Классификация и обработка электронных сообщений»	186
Борисов Д. В., Кива В. С. «Модифицированный алгоритм муравьиной колонии для поиска пути в нагруженной сети»	188
Бухаров Д. С., Гусев Р. А. «Программное обеспечение для мониторинга дистанционной защиты: подход к фиксации утяжеления режима»	190
Бухаров Д. С., Гусев Р. А. «Программное обеспечение для мониторинга дистанционной защиты: алгоритм выявления схожего режима»	192
German O. V., German J. O., Nasr S. «Decision making based on relief algorithm»	194
Гончарик И. А., Гуринович А. Б. «Система совместной работы с результатами автоматизированного тестирования»	196
Данильчик В. В., Прытков В. А. «Принципы идемпотентности и конвергентности в области миграции данных»	198
Денисов В. А., Федоров П. А. «Классификация дефектов подшипника с использованием нейронных сетей»	200
Дрозд П. С. «Локально-чувствительное хеширование для сегментации рынка»	202

Заерко Д. В., Липницкий В. А. «Алгоритм коррекции проекции однострочного номера» ..	204
Иванин Н. С., Романов А. А. «Сравнение алгоритмов кластеризации в задаче обработки GPS данных»	206
Иванюк А. И., Клапатов И. А., Чибисов И. В. «Применение языка программирования Scala в решении задач обработки данных»	208
Кадан А. М., Сазановец И. А. «Детектирование наличия скрытой информации в цифровых изображениях с использованием методов машинного обучения»	210
Калоша А. Л., Хоронеко М. П., Медунецкий М. М. «Система для прогнозирования популярности публикаций»	212
Кармаз А. М., Гуринович А. Б. «Технологии доступа к цифровым устройствам для людей с ограниченными возможностями»	214
Кисель А. Н., Головатая Е. А. «Графическая визуализация статистических данных в виртуальной реальности»	216
Кива В. С., Борисов Д. В. «Шаблоны распределения ответственности по запросам команд и источник событий»	218
Козарь Р. В., Навроцкий А. А. «Эвристика как средство для оптимизации транспортных потоков предприятий»	220
Куручкин А. В., Садов В. С. «Анализ процесса принятия решения в модели нейросетевой классификации методом локальных линейных аппроксимаций»	222
Лапицкая Н. В., Игнатюк Т. А. «Таблицы смертности на пожаре как основа для трансформации методики оценки стоимости человеческой жизни»	224
Лаппо А. И., Боброва Т. С., Кузнецова О. В. «Моделирование процесса лазерной прошивки отверстий в кремнии при формировании 3d структур»	226
Ляхор Т. В., Проволоцкий В. Н., Трофимович А. Ф., Захарьев В. А. «Современная высоконагруженная и масштабируемая система»	228
Лобач С. В., Меркулов Р. И., Акулич В. Н. «О применении статистических методов в стеганоанализе графических изображений»	230
Lobach V. I., Merkulov R. I., Lobach S. V. «Statistical Forecasting of panel data based on state space models»	232
Ma Jun, Ren Xunhuan, Tsviatkou, V. Yu., Konopelko, V. K. «Parallel skeleton algorithm with one-subiteration»	234
Моженкова Е. В., Парамонов А. И. «Метод агрегации потоков данных в задачах локализации проблем»	236
Mukha V. S., Kako N. F. «Dual control of multidimensional-matrix stochastic objects»	238
Нефёдов Д. С., Курилович В. В., Панасюк М. Н. «Обоснование классификационных признаков для распознавания маловысотных летательных аппаратов в электростатической системе пассивной локации»	240
Нестеренков С. Н., Федоров П. А., Денисов В. А. «Использование сверточных нейронных сетей для классификации и анализа тональности текстов»	242
Новицкий И. О., Шилин Л. Ю. «TON: Telegram Open Network»	244
Олейников А. А., Сорокин А. А. «Прогнозирование значений эксплуатационных параметров телекоммуникационного оборудования с использованием нейронных сетей»	246
Петров А. В., Фролов И. И. «Использование макросов для автоматизации процессов инженерных САЕ-систем»	248

Пинтусов Д. Н., Латушко Д. И. «Информационные системы идентификации рукописной подписи»	250
Полудворянин С. М., Нестеренков С. Н. «Методы борьбы со спамом в VoIP телефонии»	252
Потараев В. В. «Алгоритм построения семантической сети с обработкой местоимений текста, основанный на диалоге с пользователем»	254
Романов А. А., Иванин Н. С. «Применение методов вероятностного тематического моделирования для анализа дефектов в программном обеспечении»	256
Сабадаш М. А., Петров А. В., Ярошевич Д. И. «Автоматизация маркетинговых кампаний при помощи универсальной системы автоматизации задач»	258
Саскевич А. В., Стержанов М. В., Климович М. А. «Алгоритмы машинного обучения в приложениях для изучения иностранных языков»	260
Савёнок В. А., Медведев С. А., Селедец В. Н. «Построение классификатора текстов на естественном языке с использованием параметризованных шаблонов»	262
Свинарский М. В., Ярмолик С. Н., Храменков А. С., Зайко Е. В. «Адаптация устройства последовательного радиолокационного распознавания к пространственной ориентации цели» ..	264
Сосинович Е. С., Жук Е. Е. «Статистическое построение датчиков базовой случайной величины на основе флуктуаций физических систем»	266
Царегородцев Д. В. «Классификация вредоносного программного обеспечения по способу внедрения»	268
Урбанович А. И., Кадан А. М. «Расширение возможностей языка разметки AIML для создания интеллектуальных чат-ботов»	270
Урбанович П. П., Бирюк И. А., Плонковски М. Д. «Анализ синхронизации нейронных сетей в прикладной криптографии»	272
Фролов И. И. «Построение скоррелированных винеровских процессов»	274
Федосеев Д. С., Шилин Л. Ю. «Объективная оценка заболеваний гортани на основе акустического анализа голоса»	276
Chepeleva M., Yatskou M., Nazarov P. «The statistical stability of consensus independent component analysis for RNA-Seq data in cancer research»	278
Черняк А. И., Поттосина С. А. «Сравнительная характеристика методов выявления и идентификации рисков»	280
Ren Xunhuan, Ma Jun, Konopelko V. K. «An improved forming Algorithm of Error-Patterns for Two-Dimensional Codes»	282
Яцков Н. Н., Климух И. В., Скакун В. В., Гринев В. В. «Классификация молекул РНК»	284
Яцков Н. Н., Скакун В. В., Гринев В. В. «Автоматическое определение открытых рамок считывания в молекулах РНК человека с использованием алгоритмов векторизации и классификации»	286
Секция «Первый шаг в науку»	289
Бачило Е. Г. «Алгоритмы распределения заданий сотрудников отдела»	290
Бибик О. Н., Белый А. И., Гуринович А. Б. «Алгоритмы анализа автоматизации бизнес-процессов»	292
Гобрик О.Д., Гуринович А.Б. «Новые подходы оптимизации запросов в базах данных»	294
Геврасёва И. П. «Модель многомерного представления данных в хранилищах данных»	296

Евтушенко Е. Ю. «Обзор подходов к решению задачи переноса стиля текста с помощью нейронных сетей»	298
Заблоцкий В. В., Кита М. А., Рудикова Л. В. «О разработке веб-сервиса для семантического поиска фильмов»	300
Иванов И. П. «Анализ методов и алгоритмов поиска плагиата в исходном коде»	302
Комличенко В. Н., Салапура М. Н., Богданова Е. А. «Профессиональные компетенции специалистов информационных технологий»	304
Кулыба В. А., Гуринович А. Б. «3d-модель человеческого тела и измерение его показателей при помощи камеры»	306
Кукин Д. П., Свито И. Л., Свито А. И., Гриневич Я. Г. «Постановка задачи разработки CRM-системы»	308
Лукьянов В. Н. «Эффективная работа с памятью ЭВМ»	310
Макаров А. «Автоматизация процессов принятия решений приёмной комиссии»	312
Манин А. С., Нестеренко В. Н. «Использование ERP-системы Microsoft Dynamics 365 в образовательных учреждениях»	314
Нестеренко В. Н., Манин А. С. «Интеграция в средства документооборота ERP-системы Microsoft Dynamics 365»	316
Неживинская А. Ю., Рудикова Л. В. «Об основных аспектах разработки системы накопления и анализа данных объектов городской среды»	318
Потехин А. С., Александров А. А., Русецкий П. Н. «Программа сбора данных о структуре веб-сайтов»	320
Румысов В. С. «Анализ существующих подходов распознавания лиц»	322
Шулицкий Д. С., Водейко А. Э. «Методы защиты авторского права на программные продукты с помощью водяных знаков и отпечатков пальцев»	324
Ясницкий С. В. «Разработка веб приложения для управления бизнес процессами с использованием SAP UI 5»	326

Научное издание

**Информационные технологии и системы 2019
(ИТС 2019)**

Материалы международной научной конференции
(БГУИР, Минск, Беларусь, 30 октября 2019)

**Information Technologies and Systems 2019
(ITS 2019)**

Proceeding of The International Conference
(BSUIR, Minsk, Belarus, 30th October 2019)

В авторской редакции

Ответственный за выпуск *Л.Ю. Шилин, А.Б. Гуринович*
Компьютерная верстка *Я. Г. Гриневич, А.Ф. Трофимович*
Дизайн обложки *А.А. Навроцкий*

Подписано в печать 22.10.2019. Формат 60x84 1/8. Бумага офсетная. Гарнитура «BookAntiqua».
Отпечатано на ризографе. Усл. печ. л. 39,29. Уч.-изд. л. 45,3. Тираж 80 экз. Заказ 337.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6.