

ИЗВЕСТИЯ

Волгоградского
государственного
технического
университета

ISSN 1990-5297



Серия
Актуальные проблемы управления,
вычислительной техники и информатики
в технических системах

16+

№ 8 (218)
2018

Главный редактор научного журнала
«Известия Волгоградского государственного технического университета»
академик РАН, профессор, доктор химических наук,
президент Волгоградского государственного технического университета (ВолгГТУ)
И. А. НОВАКОВ

Редакционная коллегия:

Байбурин В. Б., д-р физ.-мат. наук, проф., академик РАЕН, засл. деятель науки РФ, Саратовский государственный технический университет (г. Саратов, Россия)
Безъязычный В. Ф., д-р техн. наук, проф., Рыбинский государственный авиационный технический университет им. П. А. Соловьева (г. Рыбинск, Россия)
Бодров В. Н., д-р, проф., Университет прикладных наук (г. Берлин, Германия)
Бребельс А., PhD, доцент факультета инженерных наук университета г. Левена (г. Левен, Бельгия)
Буренин А. А., чл.-корр. РАН, Институт машиноведения и металлургии ДВО РАН (г. Комсомольск-на-Амуре, Россия)
Голованчиков А. Б., д-р. техн. наук, проф. ВолгГТУ
Гринберг Б. А., д-р физ.-мат. наук, Институт физики металлов УрО РАН (г. Екатеринбург, Россия)
Гуревич Л. М., д-р техн. наук, доц. ВолгГТУ
Добрушин Л. Д., д-р техн. наук, Институт электросварки им. Е. О. Патона НАН Украины (г. Киев, Украина)
Злотин С. Г., д-р хим. наук, проф., Институт органической химии им. Н. Д. Зелинского РАН (г. Москва, Россия)
Иванов А. М., д-р техн. наук, проф., Московский автомобильно-дорожный государственный технический университет (МАДИ, г. Москва, Россия)
Иващенко Н. А., д-р техн. наук, проф., засл. деятель науки РФ, МГТУ им. Н. Э. Баумана (г. Москва, Россия)
Королев А. В., д-р техн. наук, проф., Саратовский государственный технический университет (г. Саратов, Россия)
Кузьмин С. В., д-р техн. наук, проф. ВолгГТУ
Кураев А. А., д-р физ.-мат. наук, проф., Белорусский государственный университет информатики и радиоэлектроники (БГУИР, г. Минск, Республика Беларусь)
Лысак В. И., академик РАН, засл. деятель науки РФ, д-р техн. наук, проф. ВолгГТУ
Марков В. А., д-р техн. наук, проф., Московский государственный технический университет (МГТУ) им. Н. Э. Баумана (г. Москва, Россия)
Мертен Клеменс, д-р техн. наук, проф. Universität Stuttgart Institut für Chemische Verfahrenstechnik (г. Штутгарт, Германия)
Навроцкий А. В., д-р хим. наук, проф. ВолгГТУ
Нижегородцев Р. М., д-р экон. наук, Институт проблем управления им. В. А. Трапезникова РАН (г. Москва, Россия)
Пай В. В., д-р физ.-мат. наук, Институт гидродинамики им. М. А. Лаврентьева СО РАН (г. Новосибирск, Россия)
Поляничков Ю. Н., д-р техн. наук, проф. ВолгГТУ
Пустовойт В. Н., д-р техн. наук, проф., засл. деятель науки РФ, Донской государственный технический университет (г. Ростов-на-Дону, Россия)
Русинов В. Л., чл.-корр. РАН, Химико-технологический институт Уральского федерального университета (ХТИ УрФУ, г. Екатеринбург, Россия)
Рыбин В. В., чл.-корр. РАН, д-р физ.-мат. наук, проф., Санкт-Петербургский политехнический университет (г. Санкт-Петербург, Россия)
Трюэль Жан-Луи, д-р экон. наук, проф., Университет «Париж-12», Сорбонна, Франция, вице-президент Международного клуба экономистов «Круг Кондратьева»
Тхай Куанг Винь, д-р философии, Институт информационных технологий (г. Ханой, Вьетнам)
Федянов Е. А., д-р техн. наук, проф. ВолгГТУ
Фролов В. А., д-р техн. наук, проф., МАТИ – Российский государственный технологический университет им. К. Э. Циолковского (г. Москва, Россия)
Шарилов В. М., д-р техн. наук, проф., Московский государственный технический университет (МАМИ, г. Москва, Россия)
Шаховская Л. С., д-р экон. наук, проф. ВолгГТУ
Шейн А. Г., д-р физ.-мат. наук, проф. ВолгГТУ
Щербаков М. В., д-р техн. наук, проф. ВолгГТУ

Серия

«АКТУАЛЬНЫЕ
ПРОБЛЕМЫ УПРАВЛЕНИЯ,
ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ
И ИНФОРМАТИКИ
В ТЕХНИЧЕСКИХ СИСТЕМАХ»

ИЗВЕСТИЯ



**ВОЛГОГРАДСКОГО ГОСУДАРСТВЕННОГО
ТЕХНИЧЕСКОГО УНИВЕРСИТЕТА**

Журнал входит в перечень утвержденных ВАК РФ изданий для публикации трудов соискателей ученых степеней

**РЕДАКЦИОННЫЙ
СОВЕТ:**

Председатель

Щербаков М. В., д.т.н., проф.

Аверченков В. И., д.т.н., проф.
(БрГТУ, г. Брянск, Россия)

Бериадский А. М., д.т.н., проф.
(ПензГУ, г. Пенза, Россия)

Брискин Е. С., д.ф.-м.н., проф.

Горобцов А. С., д.т.н., проф.

Дворянкин А. М., д.т.н., проф.

Заболеева-Зотова А. В., д.т.н.,
проф.

Кравец А. Г., д.т.н., проф.

Курейчик В. М., д.т.н., проф.
(ТаГРТУ, г. Таганрог, Россия)

Петрова И. Ю., д.т.н., проф.
(АИСИ, г. Астрахань, Россия)

Проталинский О. М., д.т.н., проф.
(АстрГТУ, г. Астрахань, Россия)

Садовникова Н. П., д.т.н., проф.

Тхай Куанг Винь, д-р философии
(Институт информационных
технологий, г. Ханой, Вьетнам)

Фоменков С. А., д.т.н., проф.

Шилин А. Н., д.т.н., проф.

Юрген Баст, д.т.н., проф.
(Технический университет Горной
Академии, г. Фрайберг, Германия)

Международный индекс журнала
ISSN 1990-5297

Журнал распространяется
по подписке.

Индекс журнала по каталогу
Агентства «Роспечать»

для Российской Федерации –
80811(ОК+ЭК). По вопросам
подписки обращаться
в издательство.

Тел. издательства ВолГТУ:

(8442) 24-84-06

Факс (8442) 24-84-06

otr@vstu.ru

Научный журнал
Издается с января 2004 г.
Выходит двенадцать раз в год

№ 8 (218)

Август 2018

УЧРЕДИТЕЛЬ:

**ФГБОУ высшего образования
«Волгоградский государственный технический университет»**

Адрес редакции:

Волгоград, пр. В. И. Ленина, 28.

Телефон: гл. редактор – (8442) 24-80-00

E-mail: president@vstu.ru

Журнал зарегистрирован в Федеральной службе по надзору в сфере связи,
информационных технологий и массовых коммуникаций (Роскомнадзор)

Свидетельство о регистрации ПИ № ФС77–61113 от 19 марта 2015 г.

Печатается по решению редакционно-издательского совета
Волгоградского государственного технического университета

**Авторское право на журнал в целом принадлежит Учредителю,
на отдельные статьи – сохраняется за автором**

*Перепечатка из журнала «Известия Волгоградского государственного технического
университета» категорически запрещена без оформления договора
в соответствии с действующим законодательством РФ*

*При перепечатке материалов ссылка на журнал «Известия Волгоградского
государственного технического университета» обязательна*

Head Editor of the scientific journal "Izvestiya VSTU":

Novakov I. A. – Academician of RAS, Prof., Doctor of Chemistry, President of VSTU

Editorial board:

Baiburin V. B., D. Sc. (Physical and Mathematical Sciences), Prof., Academician of RAS, Honored Scientist of the Russian Federation, State Technical University of Saratov (Saratov, Russia)

Bezyazychny V. F., D. Sc. (Engineering), Prof., Rybinsk State Aviation Technical University named after P.A. Solov'yov (Rybinsk, Russia)

Bodrov V. N., D. Sc., Prof., University of Applied Sciences (Berlin, Germany)

Brebels A., PhD, Associate Prof., Faculty of Engineering Science of University of Leuven (Leuven, Belgium)

Burenin A. A., Corresponding Member of RAS, Institute of Machinery and Metallurgy of the FEB RAS (Komsomolsk-on-Amur, Russia)

Golovanchikov A. B., D. Sc. (Engineering), Prof. of VSTU

Grinberg B. A., D. Sc. (Physical and Mathematical Sciences), Institute of Metal Physics, Ural Division of RAS (Ekaterinburg, Russia)

Gurevich L. M., D. Sc. (Engineering), Associate Prof. of VSTU

Dobrushin L. D., D. Sc. (Engineering), E.O. Paton Electric Welding Institute of the National Academy of Sciences of Ukraine (Kiev, Ukraine)

Zlotin S. G., D. Sc. (Chemistry), Prof., N.D. Zelinsky Institute of Organic Chemistry Russian Academy of Sciences (Moscow, Russia)

Ivanov A. M., D. Sc. (Engineering), Prof., Moscow Automobile and Road Construction University (MADI) (Moscow, Russia)

Ivashchenko N. A., D. Sc. (Engineering), Prof., Honored Scientist of the Russian Federation, Bauman Moscow State Technical University (Moscow, Russia)

Korolev A. V., D. Sc., Prof., State Technical University of Saratov (Saratov, Russia)

Kuzmin S.V., D. Sc. (Engineering), Prof. of VSTU

Kurayev A. A., D. Sc. (Physical and Mathematical Sciences), Prof., Belarusian State University of Informatics and Radioelectronics (BSUIR) (Minsk, Republic of Belarus)

Lysak V. I., Academician of RAS, Honored Scientist of the Russian Federation, D. Sc. (Engineering), Prof. of VSTU

Markov V. A., D. Sc. (Engineering), Prof., Bauman Moscow State Technical University (Moscow, Russia)

Merten K., D. Sc. (Engineering), Prof., Institute of Chemical Process Engineering (Stuttgart, Germany)

Navrotsky A. V., D. Sc. (Chemistry), Prof. of VSTU

Nizhegorodtsev R. M., D. Sc. (Economy), Chief research worker, Institute of Control Sciences V. A. Trapeznikov Academy of Sciences (Moscow, Russia)

Pai V. V., D. Sc. (Physical and Mathematical Sciences), Lavrentyev Institute of Hydrodynamics of SB RAS (Novosibirsk, Russia)

Polyanchikov Y. N., D. Sc. (Engineering), Prof. of VSTU

Pustovoit V. N., D. Sc. (Engineering), Prof., Honored Scientist of the Russian Federation, Don State Technical University (Rostov-on-Don, Russia)

Rusinov V. L., Corresponding Member of RAS, Institute of Chemical Technology of Ural Federal University (Ekaterinburg, Russia)

Rybin V. V., Corresponding Member of RAS, D. Sc. (Physical and Mathematical Sciences), Prof., St. Petersburg Polytechnic University (St. Petersburg, Russia)

Thai Quang Vinh, D. Sc. (Philosophy), Institute of Information Technology (Hanoi, Vietnam)

Truel J.-L., D. Sc. (Economy), Prof., University Paris Est Créteil, Sorbonne, France, Vice-president of International economists club "Kondratiev Circle"

Fedyanov E. A., D. Sc. (Engineering), Prof. of VSTU

Frolov V. A., D. Sc. (Engineering), Prof., "MATI Russian State Technological University named after K.E. Tsiolkovsky" (Moscow, Russia)

Sharipov V. M., D. Sc. (Engineering), Prof., Moscow State Technical University "MAMI" (Moscow, Russia)

Shakhovskaya L. S., D. Sc. (Economy), Prof. of VSTU

Shcherbakov M. V., D. Sc. (Engineering), Prof. VSTU

Sheyin A. G., D. Sc. (Physical and Mathematical Sciences), Prof. of VSTU

SERIES "ACTUAL
PROBLEMS
OF MANAGEMENT,
COMPUTING HARDWARE
AND INFORMATICS
IN ENGINEERING
SYSTEMS"

IZVESTIA



VOLGOGRAD STATE TECHNICAL
UNIVERSITY

The Journal is included to the list of peer-reviewed scientific journals and publications of the Higher Attestation Commission (HAC) of the RF for publishing of results of candidates for scientific degrees

Editorial board:

Chairman

Shcherbakov M. V., D. Sc.
(Engineering), Prof.

Averchenkov V. I., D. Sc.
(Engineering), Prof. (BSTU, Bryansk,
Russia)

Berdshadskiy A. M., D. Sc.
(Engineering), Prof. (PSU, Penza,
Russia)

Briskin E. S., D. Sc. (Physical
and Mathematical Sciences), Prof.
Gorobtsov A. S., D. Sc. (Engineering),
Prof.

Dvoryankin A. M., D. Sc.
(Engineering), Prof.

Zaboleeva-Zotova A. V., D. Sc.
(Engineering), Prof.

Kravets A. G., D. Sc. (Engineering)

Kureytchik V. M., D. Sc.
(Engineering), Prof. (TRTU,
Taganrog, Russia)

Petrova I. Y., D. Sc. (Engineering),
Prof. (AICE, Astrakhan, Russia)

Protalinskiy O. M., D. Sc.
(Engineering), Prof. (ASTU,
Astrakhan, Russia)

Sadovnikova N. P., D. Sc. (Engineering)
Thai Quang Vinh, D. Sc. (Philosophy),
Institute of Information Technology
(Hanoi, Vietnam)

Fomenkov S. A., D. Sc. (Engineering),
Prof.

Shilin A. N., D. Sc. (Engineering), Prof.

Jurgen Bast D. Sc. (Engineering),
Prof. (TU Bergakademie Freiberg,
The University of Resources,
Freiberg, Germany)

International index of the journal
ISSN 1990-5297

The journal is distributed by subscrip-
tion.

Index of the journal in the catalogue of
the Agency "Rospechat" for the Rus-
sian Federation – 80811(OK+ЭК).
Concerning subscription turn to the
editorial office.

Tel. Of VSTU editorial office:

(+7 8442) 24-84-06

Fax: (+7 8442) 24-84-06

otr@vstu.ru

Scientific journal

Published since January 2004

12 times a year

№ 8 (218)
August
2018

FOUNDER:

FSBEI of Higher Education
"Volgograd State Technical University"

Editorial office address:

Volgograd, Lenin avenue, 28.

Tel.: Head Editor – (+7 8442) 24-80-00

E-mail: president@vstu.ru

The journal is registered at the Federal Service on Control in the Sphere of Communication,
IT and Mass Communications (Roscomnadzor)
Registration certificate ПИ № ФС77–61113 of March 19, 2015

Is printed according to the decision of editorial-review board of
Volgograd State Technical University

***The copyright for the journal in general belongs to the Founder,
for separate articles – to authors.***

*Reprint from the journal "Izvestia VSTU" is strongly forbidden without conclusion of an
agreement in accordance with the legislation of the RF
When reprinting the materials, the citation to the journal "Izvestia VSTU" is obligatory*

© Volgograd State Technical University,
"Izvestia VSTU", 2018

АВТОРСКИЙ УКАЗАТЕЛЬ

- | | |
|---------------------------|------------------------|
| Аль-Хадша Ф. А. Х. 83 | Малолетов А. В. 117 |
| Андреев А. Е. 30 | Машевская Ю. А. 73 |
| Бершадский А. М. 7 | Михнев И. П. 105 |
| Бождай А. С. 7 | Михолап Л. А. 97, 101 |
| Борозенец В. Н. 49 | Недоступов Д. О. 54 |
| Брискин Е. С. 117 | Орлова Ю. А. 54 |
| Воля Я. И. 63 | Панфилов А. Э. 42, 68 |
| Голованчиков А. Б. 14 | Пересыпкин И. П. 109 |
| Гудков А. А. 7 | Петрова Т. М. 73 |
| Гусынин О. С. 54 | Петрухин А. В. 49 |
| Дворянкин А. М. 19, 88 | Полежаев А. В. 30 |
| Доан Минь Кыонг 14 | Поцелуйко А. С. 88 |
| Евсеева Ю. И. 7 | Привалов О. О. 39 |
| Егунов В. А. 79 | Ражева А. В. 54 |
| Жерлицын С. А. 63 | Ребриков Г. И. 26 |
| Заболеева-Зотова А. В. 54 | Розалиев В. Л. 54 |
| Кислов О. В. 26, 97, 101 | Садовникова Н. П. 109 |
| Князев А. А. 79 | Сальникова Н. А. 105 |
| Козлов Д. В. 109 | Смыковская Т. К. 73 |
| Колесников С. Г. 83 | Старусев А. В. 97, 101 |
| Коробкин Д. М. 83 | Умницын М. Ю. 112 |
| Кравец А. Г. 105 | Филимонов М. И. 14 |
| Крушель Е. Г. 39, 42, 68 | Фоменков С. А. 83 |
| Кульцова М. Б. 88 | Харитонов И. М. 42, 68 |
| Лепетухин К. Ю. 117 | Частикова В. А. 63 |
| Лобейко В. И. 97, 101 | Шевченко К. А. 49 |
| Лютая Т. П. 39 | Щербаков М. В. 39 |

СОДЕРЖАНИЕ

Часть I. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ЧИСЛЕННЫЕ МЕТОДЫ

<i>Бершадский А. М., Бождай А. С., Евсеева Ю. И. Гудков А. А.</i> Математическая модель рефлексии самоадаптивных программных систем..	7
<i>Голованчиков А. Б., Филимонов М. И., Доан Минь Кыонг</i> Моделирование процесса обратного осмоса в фильтрующих сверхцентрифугах.....	14
<i>Дворянкин А. М.</i> Решение задачи размещения нагруженных отрезков на системе LINGO.....	19
<i>Кислов О. В., Ребриков Г. И.</i> Анализ и сравнение результатов работы методики выделения огигающих из сигналов, различных по структуре и частотным свойствам.....	26

Часть II. СИСТЕМНЫЙ АНАЛИЗ И ОБРАБОТКА ИНФОРМАЦИИ

<i>Андреев А. Е., Полежаев А. В.</i> Разработка веб-сервиса для сравнения текстовых документов.....	30
<i>Крушель Е. Г., Лютая Т. П., Привалов О. О., Щербаков М. В.</i> Распознавание пожароопасных ситуаций в лесных массивах и лесополосах.....	39
<i>Крушель Е. Г., Панфилов А. Э., Харитонов И. М.</i> Трендовая модель прогнозирования качества подготовки выпускников вуза на примере КТИ (филиала) ВолгГТУ.....	42
<i>Петрухин А. В., Борозенец В. Н., Шевченко К. А.</i> Автоматизация дивидендного мониторинга компаний-эмитентов.....	49
<i>Розалиев В. Л., Заболеева-Зотова А. В., Орлова Ю. А., Гусынин О. С., Ражева А. В., Недоступов Д. О.</i> Разработка методов определения эмоциональности человека по мимике, тексту, движениям рук и глаз.....	54
<i>Частикова В. А., Жерлицын С. А., Воля Я. И.</i> Нейросетевой подход к решению задачи построения фоторобота по словесному описанию.....	63

Часть III. УПРАВЛЕНИЕ В СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ СИСТЕМАХ

<i>Крушель Е. Г., Панфилов А. Э., Харитонов И. М.</i> Анализ эффективности изучения профилирующих учебных дисциплин в вузе на примере КТИ (филиала) ВолгГТУ.....	68
<i>Петрова Т. М., Смыковская Т. К., Машевская Ю. А.</i> Использование Google-документов при конструировании индивидуальных образовательных траекторий.....	73

Часть IV. АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ

<i>Егунов В. А., Князев А. А.</i> Проектирование оптимизированной архитектуры WEB-клиента, взаимодействующего с REST API.....	79
<i>Коробкин Д. М., Фоменков С. А., Колесников С. Г., Аль-Хадша Ф. А. Х.</i> Синтез и анализ физических принципов действия технических систем с использованием сетей Петри.....	83

<i>Поцелуйко А. С., Кульцова М. Б., Дворянkin А. М.</i> Редактор для создания адаптивных интерфейсов мобильных приложений для людей с ограниченными возможностями.....	88
<i>Старусев А. В., Михолап Л. А., Кислов О. В., Лобейко В. И.</i> Метод оценки затрат труда программиста при решении задачи испытаний.....	97
<i>Старусев А. В., Михолап Л. А., Кислов О. В., Лобейко В. И.</i> Метод оценки производительности ЭВМ при испытаниях сложных технических систем.....	101
 Часть V. ПОСТРОЕНИЕ СЕТЕЙ И ЗАЩИТА ИНФОРМАЦИИ	
<i>Михнев И. П., Сальникова Н. А., Кравец А. Г.</i> Защита информации от несанкционированного доступа при анализе радиационных характеристик помещений спектрометрическим методом.....	105
<i>Пересыпкин И. П., Садовникова Н. П., Козлов Д. В.</i> Анализ подходов и требований к системе безопасности POS-сети.....	109
<i>Умницын М. Ю.</i> Подход к полунатурному анализу защищенности информационной системы.....	112
 Часть VI. РОБОТОТЕХНИКА	
<i>Лепетухин К. Ю., Малолетов А. В., Брискин Е. С.</i> О критериях оптимизации движения секций дождевальной машины кругового действия при обработке полей сложной формы.....	117
К сведению авторов.....	123

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ЧИСЛЕННЫЕ МЕТОДЫ

УДК 004.94

А. М. Бершадский, А. С. Бождай, Ю. И. Евсеева, А. А. Гудков

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ РЕФЛЕКСИИ САМОАДАПТИВНЫХ ПРОГРАММНЫХ СИСТЕМ*

Пензенский государственный университет

bam@pnzgu.ru, bozhday@yandex.ru, shymoda@mail.ru, alexei.gudkov@gmail.com

В статье рассмотрена проблема создания универсального математического аппарата для реализации самоадаптивных свойств прикладных программных систем. Предлагаемая авторами математическая модель базируется на концепции рефлексивной адаптации, согласно которой программная система способна формировать и встраивать в свою структуру новые обратные связи путем интеллектуального анализа собственной поведенческой информации. Ядром предлагаемой модели является гиперграфовое представление диаграммы характеристик, с помощью которого программная система способна изменять свою логическую и компонентную структуру непосредственно в процессе исполнения.

Также в статье рассматриваются вопросы применения самоадаптивных программных систем, показывается их актуальность и практическая востребованность.

Ключевые слова: самоадаптивная программная система, самоадаптация, рефлексия, математическая модель изменчивости, теория гиперграфов, диаграмма характеристик приложения.

A. M. Bershadsky, A. S. Bozhday, Y. I. Evseeva, A. A. Gudkov

MATHEMATICAL MODEL OF REFLECTION FOR SELF-ADAPTIVE SOFTWARE SYSTEMS

Penza State University

The article considers the problem of creating a universal mathematical apparatus for realizing the self-adaptive properties of applied software systems. The mathematical model described in the article is based on the concept of reflection proposed by the authors, according to which the program system is capable of forming and integrating into its structure new feedbacks through the intellectual analysis of their own behavioral information. The core of the proposed model is a hypergraph representation of the characteristics diagram, by means of which the software system is able to change its logical and component structure directly in the execution process.

Also in the article questions of application of self-adaptive program systems are considered, their urgency and practical relevance is shown.

Keywords: self-adaptive software system, self-adaptation, reflection, mathematical model of variability, theory of hypergraphs, application feature diagram.

Введение

Непрерывное усложнение информационных процессов современного общества накладывает все более жесткие требования на поведенческие модели используемого программного обеспечения (ПО). Различные информационные системы используются в управлении крупными социально-экономическими структурами (банки, органы государственного регулирования и т. д.), программы-тренажеры успешно применяются в об-

разовательном процессе, различные мобильные утилиты (например, геонавигаторы и переводчики) способны существенно упростить как профессиональную, так и повседневную деятельность человека.

Возрастающая сложность программных систем, используемых в жизни общества, порождает и ряд проблем, которые могут стать критичными в определенных ситуациях. Это проблемы высокой стоимости создания и сопровож-

дения программной системы, поддержания требуемой длительности жизненного цикла ПО, отказоустойчивости, надежности, гибкости. Так, неспособность к самовосстановлению может привести к потерям больших объемов данных в аварийных ситуациях, а в условиях динамично изменяющегося современного мира многие информационные системы (например, системы, обеспечивающие работу крупных предприятий, ERP-системы), лишённые способности к саморазвитию, быстро устаревают. Решения перечисленных проблем можно достичь путем создания универсальных механизмов самоадаптации прикладных программных систем.

В статье предлагается новый подход к самоадаптации программных систем, схожий с механизмами, присущими наиболее высокоорганизованным биологическим существам. Речь идет о рефлексивных типах адаптивного поведения, которые связаны с сознательной деятельностью человека: самоанализом, способностью ориентироваться в сложных ситуациях и планировать собственные действия, обучением посредством наблюдения за собственными состояниями и объектами внешнего мира. Подобный выбор связан с тем, что современные программные системы прикладного назначения, обрабатывающие большие объемы сложноструктурированной и неструктурированной информации, имеющие крайне сложную архитектуру, должны иметь соответствующую поведенческую сложность и разнообразие без нарушения заданной функциональной компактности.

1. Постановка задачи и существующие решения

Основной задачей исследования является разработка универсальных теоретических основ самоадаптации прикладного ПО. Первый этап ее решения заключается в создании универсальной математической модели адаптивной программной системы. Модель должна учитывать определенный набор свойств и возможностей адаптивной программы:

- возможность системы, на этапе исполнения, формировать и встраивать в свою структуру новые обратные связи, позволяющие ей отслеживать и обрабатывать внешние и внутренние (межкомпонентные) взаимодействия, непредвиденные на этапе компиляции;

- возможность самоадаптации с учетом поведенческих особенностей разнородных объектов, входящих в структуру программной системы – информационных и графических моделей,

интерфейсных элементов, алгоритмов, программных функций, структур данных и т. д.;

- возможность структурной и поведенческой реорганизации приложения в ходе работы без перекомпиляции исходного кода;

- свойство инвариантности к предметной области.

Таким образом, необходимо выбрать математический аппарат, позволяющий реализовать все перечисленные свойства и обеспечить создание универсальной (обобщенной) модели адаптивного поведения программной системы. Модель должна настраиваться разработчиком системы под нужды конкретной предметной области (т. е. из общей, абстрактной модели преобразовываться в уточненную модель).

Большинство попыток разработать универсальный аппарат для реализации адаптивного поведения в программных системах нельзя считать удачными, поскольку они предпринимались в основном специалистами-практиками для создания частных архитектурных решений, а не фундаментальных принципов, пригодных для построения широкого класса самоадаптивного ПО. К работам, посвященным разработке адаптивной архитектуры, относится прежде всего ряд публикаций о создании адаптивных компьютерных игр и виртуальных тренажеров [1–5]. Малочисленные работы, посвященные формализации процессов адаптации в программных системах, к сожалению, в настоящий момент не предлагают универсальных подходов. Так, широко известная концепция автономных вычислений (autonomic computing), подробно изложенная в работе [6], применима в основном для задач конфигурирования в режиме реального времени крупных программно-аппаратных комплексов. Ряд других публикаций, например, работы P. Wang, K. Cai и Q. Yang, посвященные использованию нечеткой логики и конечных автоматов при проектировании адаптивных программ [7, 8, 9], или методы, основанные на агентных технологиях [10], также не являются достаточно гибкими и применимы далеко не во всех предметных областях.

В данной работе предлагается использовать технологию моделирования изменчивости (variability modeling), в частности, применить так называемые модели характеристик (feature models) [11].

2. Концепция рефлексии и рефлексивные компоненты

Одним из наиболее интересных явлений в сфере психической деятельности разумного

существа (человека) является рефлексия. Под рефлексией понимается способность мышления к критическому самоанализу. Благодаря ей, человек может не просто знать или чувствовать нечто, но и знать о своем знании и переживании. Это умение отслеживать то, что происходит на разных уровнях сознания, с возможностью дальнейшего переосмысления. Именно путем рефлексивного самоанализа человек способен выявлять те стратегии своего поведения, которые приводят к нежелательным результатам, а также формировать новые более эффективные.

Одной из проблем современного программного обеспечения является проблема недостаточности знаний во многих предметных областях. Их динамика и сложность зачастую столь сложны, что невозможно учесть все связанные с ними обстоятельства на этапе разработки информационных моделей и компиляции исходного кода. Это заставляет создателей таких систем проявлять повышенное внимание к процессу сопровождения, регулярно выпускать обновления, не содержащие кардинально новой функциональности, но отнимающие внимание, время и материальные средства конечного пользователя.

Решением данной проблемы может стать внесение в функционал программной системы интеллектуальных свойств самоадаптации, схожих с механизмами человеческой рефлексии. В таком случае, разработчику достаточно формализовать лишь базовые модели анализа и поиска закономерностей (тем самым делая предположение, что какие-то моменты на этапе разработки остались неучтенными), а выявлением недостающих знаний система будет заниматься самостоятельно на протяжении своего жизненного цикла. Такой подход не только позволит снизить затраты на сопровождение системы, но и выявить те сведения о ее предметной области, которые могут быть неизвестны даже экспертам.

Концепция рефлексивной адаптации базируется на следующих принципах:

- инвариантность к предметной области, типу и назначению программной системы;
- масштабируемость;
- интеллектуальный анализ ретроспективных данных.

Условимся называть данный тип адаптации рефлексивной самоадаптацией или рефлексией программной системы. Можно выделить три базовых уровня программной рефлексии (рис. 1):

уровень самонаблюдения, самооптимизации и самоадаптации.



Рис. 1. Уровни рефлексии программной системы

Уровень самонаблюдения является базовым и наиболее примитивно организованным уровнем. Он не описывается какой-либо сложной математической моделью. Основная задача механизмов адаптации, функционирующих на этом уровне – составление поведенческих протоколов программной системы.

Уровень самооптимизации затрагивает организацию приложения на уровне его программного кода и интерфейса. Механизмы адаптации, работающие на этом уровне, используют поведенческую информацию уровня самонаблюдения для оптимизации указанных составляющих программной системы. В данном случае программа адаптируется не под конкретного пользователя или устройство, а устраняет в себе свойства, которые являются объективно нежелательными. Объективно нежелательными свойствами являются те, которые признаны таковыми для любых пользователей и при любых условиях. К числу таких свойств относится, например, избыточность пользовательского интерфейса.

Уровень самоадаптации является наиболее сложноорганизованным уровнем. Именно на этом уровне для формализации адаптивного поведения используются дополнительные модели – модели изменчивости. Модель изменчивости представляет собой своеобразное морфологическое множество адаптивной программной системы, т. е. множество его потенциальных конфигураций. Если уровень самооптимизации

определяет то, насколько оптимально написан код или организован интерфейс, то уровень самоадаптации отвечает именно за поведенческую реорганизацию приложения в процессе выполнения. Именно на этом уровне формиру-

ются новые обратные связи системы.

В качестве моделей изменчивости предлагается использовать модели характеристик, в частности, их графическое представление – диаграммы характеристик (рис. 2).

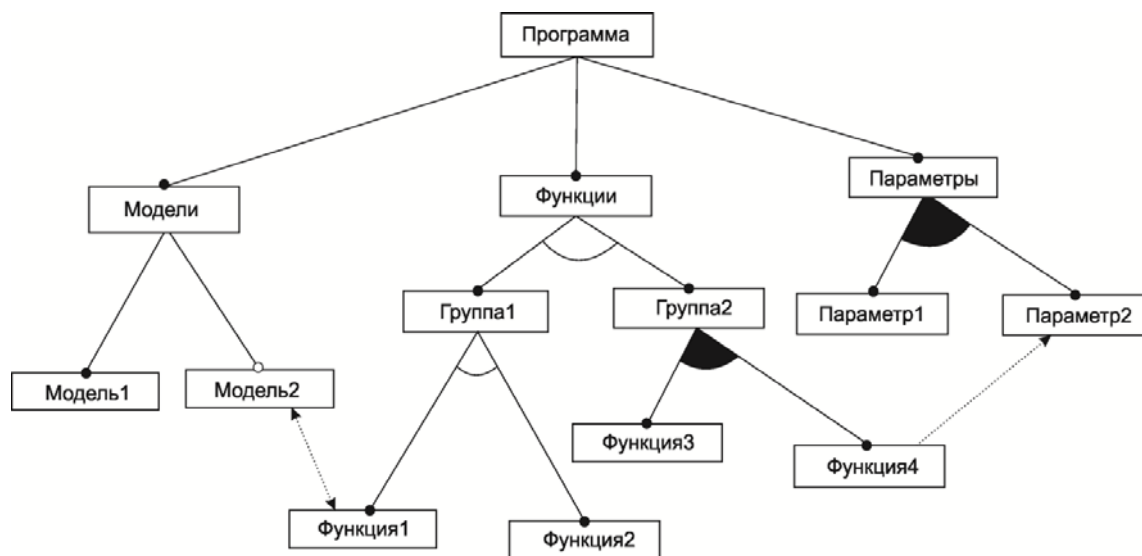


Рис. 2. Диаграмма характеристик

Диаграмма характеристик представляет собой усовершенствованное И/ИЛИ-дерево, вершинами которого являются компоненты программной системы, а связи между вершинами определяют совместимость компонентов в пределах одной конфигурации.

Рефлексивная программная система также должна иметь собственную специфическую архитектуру. Обобщая существующий опыт разработки систем с модульной архитектурой, можно сформулировать несколько важных принципов построения программных систем с рефлексивной самоадаптацией функционала:

- минимальными компонентами программной системы, подверженными адаптации, являются блоки изменчивости. Каждый блок изменчивости, входящий в состав системы, отвечает за возможную реализацию какого-либо атомарного компонента программы (функции, класса, модели объекта предметной области и т. п.);
- совокупность логически связанных блоков изменчивости, относящихся к одной предметной области и предназначенной для решения одной задачи, называется рефлексивным компонентом;
- рефлексивная программная система может состоять из одного или нескольких рефлексивных компонентов. Их количество будет опре-

делять адаптивное разнообразие и рефлексивную сложность программы;

- рефлексивный компонент включает в себя два базовых архитектурных уровня: уровень самооптимизации и уровень самоадаптации. Уровень самооптимизации имеет также четкое разделение на уровень представления (уровень интерфейса, не является обязательным) и уровень программного кода. В свою очередь, уровень самооптимизации построен на основе моделей изменчивости;

- рефлексивные компоненты объединены в единую систему посредством общего интерфейса и изначально могут не иметь данных друг о друге. Однако в процессе рефлексивного анализа будет происходить выявление степени и характера влияния одних компонентов на другие. Для этого необходимо предусмотреть каналы информационного обмена между компонентами.

Таким образом, общая математическая модель рефлексии адаптивной программной системы будет иметь два базовых уровня: уровень самооптимизации и уровень самоадаптации. Для построения математической модели уровня самоадаптации требуется использование моделей изменчивости. Оба уровня рефлексии также подразумевают использование алгоритмов интеллектуального анализа данных [12]: в слу-

чае с уровнем самооптимизации они будут использоваться для выявления объективно нежелательных свойств системы, в случае с уровнем самоадаптации – для формирования новых обратных связей.

3. Математическая модель уровня самооптимизации

Математическая модель рефлексии уровня самооптимизации будет иметь следующий вид:

$$S_A = A(S, L(S, V)), \quad (1)$$

где S_A – модифицированная структура системы; A – функция анализа поведенческой продукции системы и модификации ее первоначальной структуры; L – функция вычисления поведенческой продукции системы; S – исходная структура системы; V – совокупность внешних воздействий на адаптируемые объекты.

Применительно к S также справедливо

$$S = \{I, FC\},$$

где I – множество объектов уровня интерфейсов; FC – множество адаптируемых объектов уровня исходного кода.

Рефлексию объектов уровня интерфейсов можно определить следующим образом:

$$I_A = A(I, L_I(S, V)). \quad (2)$$

Пространство интерфейсов программной системы можно представить как кортеж из двух множеств:

$$I = (C, B).$$

Множество $B = \{B_1, B_2, B_3, \dots, B_n\}$, $i = 1, \dots, n$ представляет собой множество связей между элементами интерфейса (n – количество связей). Множество C описывает структуру пространства на уровне элементов:

$$C = \{C_1, C_2, C_3, \dots, C_m\}, C_i = (T_i, P_i, FB_i) \forall i = 1, \dots, m,$$

где m – число элементов интерфейса; $T_i = 0 \mid 1$ – тип i -го интерфейса (элемент-контейнер, используемый только для отображения или вызова дочерних интерфейсов, или функционирующий элемент, связанный с определенными элементами программного кода); P_i – множество параметров i -го интерфейса (например, ширина и высота окна); $FB_i \subseteq F$ – множество идентификаторов, связанных с i -м интерфейсом объектов исходного кода (функций и классов).

Механизм самооптимизации пользовательского интерфейса, разработанный авторами, представлен в работе [13].

Рефлексия объектов уровня программного кода будет иметь следующий вид:

$$FC_A = A(FC, L_{FC}(S, V)). \quad (3)$$

Пространство элементов исходного кода системы также можно представить как кортеж из двух множеств:

$$FC = (EC, RC).$$

Множество $EC = \{EC_1, EC_2, EC_3, \dots, EC_k\}$, $i = 1, \dots, k$ представляет собой множество сущностей исходного программного кода (функций, методов классов, объектов классов). Множество R является множеством связей между сущностями. Оно также имеет следующую форму записи:

$$RC = (AG, CL),$$

где $AG = \{AG_1, AG_2, AG_3, \dots, AG_l\}$, $i = 1, \dots, l$ – множество связей между сущностями исходного кода, обозначающими отношение агрегации (l – число таких связей); $CL = \{CL_1, CL_2, CL_3, \dots, CL_p\}$, $i = 1, \dots, p$ – множество связей, обозначающих отношение вызова (p – количество связей).

4. Математическая модель уровня самоадаптации

Формально модель характеристик представляет собой ориентированный прямой гиперграф (или F -граф)

$$F = (N, E, \Delta, \Psi), \quad (4)$$

где $N = \{N_1, N_2, \dots, N_n\}$ – конечное множество вершин, каждая из которых представляет собой характеристику диаграммы; $E = \{E_1, E_2, \dots, E_m\}$ – конечное множество ребер графа (отношений модели характеристик); $\Delta \in N$ – вершина, представляющая корневую характеристику диаграммы; $\Psi: E \rightarrow M, M \subset N \times N$ – функция маркировки, которая присваивает значение мощности $mv(E_i) = (min, max) \in M$, где M – конечное множество значений мощности модели характеристик; N – конечное множество вершин гиперграфа, так что $min, max \in Z \wedge min \geq 0, max > 0, min \leq max \wedge max \leq q_i$, где Z – множество целых чисел.

С помощью отношений между характеристиками диаграммы реализуются блоки изменчивости – структурные единицы программной системы, затрагиваемые адаптивными процессами.

Блок изменчивости может использоваться для определения множества состояний отдельного элемента (или компонента) программной

системы, а также правила выбора того или иного состояния в конкретной ситуации. Так, с его помощью можно описать ситуацию выбора наиболее оптимального алгоритма генерации изображения в зависимости от конфигурации аппаратной платформы. Конкретное состояние блока изменчивости будет определяться выбранным в текущий момент времени алгоритмом. В более сложных случаях с помощью блоков изменчивости реализуются настройки конфигурации целостного компонента либо совокупности компонентов (системы).

Блок изменчивости реализует принцип масштабируемости концепции рефлексивной адаптации. С помощью блока изменчивости может описываться адаптивное поведение как отдельной составляющей компонента, так и самого компонента, а также целой системы и совокупности систем. Отсюда следует, что блок изменчивости не атомарен, он может включать в себя другие блоки, иметь иерархическую структуру.

Блок изменчивости описывается диаграммой характеристик. Корневая вершина такой диаграммы называется точкой вариации.

Формально блоку изменчивости соответствует следующее определение:

$$V_B = (SubF, BlockParams, BlockStates, Rule), \quad (5)$$

где $SubF$ – гиперграфовое представление модели характеристик блока изменчивости, который может описывать адаптивное поведение отдельного компонента либо целостной системы; $BlockParams = \{BlockParam_1, BlockParam_2, \dots, BlockParam_k\}$ – множество параметров, влияющих на состояние блока; $BlockStates = \{BlockState_1, BlockState_2, \dots, BlockState_z\}$ – множество состояний блока, при этом $BlockState_i \subseteq SubF \forall i = 1, \dots, z$; $Rule: BlockParams' \rightarrow BlockStates$ – функция, ставящая в соответствие каждому элементу множества $BlockParams'$ некоторый элемент множества $BlockStates$. Множество $BlockParams'$ представляет собой множество подмножеств $BlockParams$.

Множество влияющих параметров $BlockParams$ можно также представить следующим образом:

$$BlockParams = ExtParams \cup InnerParams \cup TargetParams, \quad (6)$$

где $ExtParams = \{ExtParam_1, ExtParam_2, \dots, ExtParam_{k1}\}$ – множество внешних по отношению к рассматриваемому блоку параметров;

$InnerParams = \{InnerParam_1, InnerParam_2, \dots, InnerParam_{k2}\}$ – множество внутренних варьируемых параметров блока;

$TargetParams = \{TargetParam_1, TargetParam_2, \dots, TargetParam_{k3}\}$ – множество целевых параметров. Элементы множества $TargetParams$, как правило, связаны с некоторыми целевыми функциями и определяются (в отличие от других подмножеств множества $Params$) экспертным путем.

Изменчивость всех архитектурных уровней рефлексивной программной системы будет описываться единой математической моделью:

$$V_E = (F, V_{BS}, States, GenStates, Trans), \quad (7)$$

где F – гиперграфовое представление модели изменчивости рассматриваемого архитектурного уровня; $V_{BS} = \{V_{B1}, V_{B2}, \dots, V_{Br}\}$ – множество блоков изменчивости архитектурного уровня; $States = \{State_1, State_2, \dots, State_f\}$ – множество состояний архитектурного уровня, при этом

$State_i \subseteq F \forall i = 1, \dots, f$; $GenStates: GenParams, F \rightarrow NewState$ – функция генерации нового состояния $NewState \subseteq F$ на основе множества параметров $GenParams = \bigcup_{i=1}^r ExtParam_i$, где r – число точек вариации рассматриваемого архитектурного элемента, и гиперграфа F ; $Trans: TransParams, MX \rightarrow States$ – функция перехода системы в состояние $State_i \subseteq F \forall i = 1, \dots, f$ на основе множества влияющих на смену состояния параметров $TransParams = GenParams$ и матрицы переходов MX .

Обобщенный алгоритм работы функции $GenStates$ выглядит следующим образом:

1. Получение значений множества влияющих параметров $GenParams$.
2. Создание нового подграфа $NewState$, изначально состоящего только из одной корневой вершины.
3. Начало обхода ориентированного гиперграфа F .
4. В случае если встреченная вершина представляет собой точку вариации (начало блока изменчивости), переход к пункту 5. В противном случае – добавление вершины в подграф и переход к пункту 7.

5. Если функция *Rule* для блока изменчивости определена, переход к пункту 6. В противном случае в подграф включается произвольная вершина отношения и осуществляется переход к пункту 7.

6. Получение фрагмента формируемого подграфа из множества *States* путем оценки элементов множества *GenParams* с помощью функции *Rule*.

7. Если обход гиперграфа еще не завершен, переход к пункту 4 алгоритма.

8. Завершение обхода ориентированного гиперграфа *F*.

9. Верификация сформированного состояния *NewState*.

10. Завершение работы алгоритма.

Функция *Trans* определяет правила перехода между состояниями *States*. Функция оперирует значениями матрицы переходов *MX* и множеством влияющих на переход в другое состояние параметров *TransParams*.

Матрица переходов *MX* имеет вид:

$$MX = \begin{pmatrix} (MX_{1,1,1}, MX_{1,1,2}, \dots, MX_{1,1,k}), \dots, (MX_{1,m,1}, MX_{1,m,2}, \dots, MX_{1,m,k}) \\ (MX_{2,1,1}, MX_{2,1,2}, \dots, MX_{2,1,k}), \dots, (MX_{2,m,1}, MX_{2,m,2}, \dots, MX_{2,m,k}) \\ \dots \\ (MX_{n,1,1}, MX_{n,1,2}, \dots, MX_{n,1,k}), \dots, (MX_{n,m,1}, MX_{n,m,2}, \dots, MX_{n,m,k}) \end{pmatrix}$$

Элементами матрицы *MX* являются векторы, содержащие конкретные значения элементов множества, влияющих на переход параметров *TransParams*. Переход из состояния *k* в состояние *n* осуществляется в том случае, если значения вектора влияющих параметров в ячейке MX_{kn} являются более близкими к текущим значениям элементов множества *TransParams*, чем значения в иных ячейках.

Матрица переходов *MX* и множество *TransParams* могут быть изначально полностью определены либо определены частично. Рефлексивная программная система способна достраивать матрицу переходов и дополнять множество *TransParams* в процессе функционирования. Множество состояний *States* также может расширяться на протяжении жизненного цикла программы.

Можно выделить следующие базовые архитектурные уровни рефлексивной системы:

1. Уровень компонентов.

2. Уровень внутрисистемный (межкомпонентный).

3. Уровень межсистемный.

В случае с межсистемным уровнем рассматривается взаимное влияние полностью самостоятельных программных систем друг на друга. В таком случае объектом рефлексии является некоторая метасистема, в которой рассматриваемые самостоятельные системы будут выполнять роль подсистем.

Первые два уровня будут описываться математической моделью в соответствии формулой (7). Таким образом, общая математическая

модель рефлексивной системы будет иметь следующий вид:

$$MS = \{S_1, S_2, S_3, \dots, S_i\}, i = 1, \dots, n, \quad (8)$$

где *MS* обозначает саму метасистему как совокупность отдельных систем *S_i*. При этом каждая система *S_i* описывается моделью изменчивости в соответствии с формулой (7), в которой *F* – гиперграфовое представление диаграммы характеристик, вершинами которой являются компоненты *i*-й системы. В свою очередь, каждый из компонентов *i*-й системы также описывается формулой (7), но в случае с компонентами *F* в качестве вершин будет включать в себя непосредственно элементы программного приложения (функции, объекты классов, модели, алгоритмы и т. д.).

Заключение

Предложенная в работе математическая модель рефлексии адаптивной программной системы обладает свойствами универсальности и масштабируемости и может использоваться для построения адаптивного программного обеспечения различного типа и назначения. Уровень самооптимизации модели позволит находить ошибки проектировщиков программы непосредственно в процессе ее функционирования, а уровень самоадаптации – повысить гибкость программного приложения и наделить его способностью к формированию новых обратных связей в течении жизненного цикла. Последнее свойство делает программную систему более долгоживущей и снижает стоимость ее сопровождения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Hogg, C. Game AI for domination games / C. Hogg [и др.] // Artificial intelligence for computer games / под ред. P. A. Gonzalez-Galero, M.A. Gomez-Martin. – New York: Springer, 2011. – С. 83–101.
2. Wang, A. L. Software architectures and the creative processes in game development / A.L. Wang, N. Nordmark // Entertainment computing / edited by K. Chorianopoulos [et al.]. – Cham: Springer International Publishing, 2015. – P. 272–285.
3. Riener, R. Virtual reality in medicine / R. Riener, M. Harders. – London: Springer London, 2012. – 294 p.
4. Harris, A. Simulation and training in minimal access surgery / A. Harris, F. Bello, R. Kneebone // Training in minimal access surgery / edited by N. Francis [et al.]. – London: Springer London, 2015. – P. 35–47.
5. Torbeyns, J. Describing and studying domain-specific serious games / J. Tornbeyns, E. Lehtinen, J. Elen. – Cham: Springer International Publishing, 2015. – 250 p.
6. Ahuja, K., Dangey, H. Autonomic Computing: An emerging perspective and issues // IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT 2014). – Washington: IEEE Computer Society, 2014. – P. 471–475.
7. Wang P., Cai, K. Y. Representing extended finite state machines for SDL by a novel control model of discrete event systems // Sixth IEEE International Conference on Quality Software (QSIC 2006). – Washington: Ieee Computer Society, 2006. – P. 159–166.
8. Wang P., Cai, K. Y. Supervisory control of a kind of extended finite state machines // 24th IEEE Chinese Control and Decision Conference (CCDC). – Washington: Ieee Computer Society, 2012. – P. 775–780.
9. Yang, Q., L., J., Xing, J., Tao, X., Hu, H., Zou, Y. Fuzzy control-based software self-adaptation: A case study in mission critical systems // IEEE 35th Annual Computer Software and Applications Conference Workshops (COMPSACW). – Washington: IEEE Computer Society, 2011. – P. 13–18.
10. Sim, K. M. Agent-based cloud computing // IEEE Transactions on Services Computing. – Washington: IEEE Computer Society, 2012. – P. 564–567.
11. Schobbens, P. E. Feature diagrams: a survey and formal semantics / P. E. Schobbens, P. Heymans, J.C. Trigaux // In 14th IEEE International Requirements Engineering Conference (RE'06). – Washington: IEEE Computer Society, 2011. – P. 139–148.
12. Барсегян, А., Куприянов, М., Холод, И., Тесс М., Елизаров С. Анализ данных и процессов. – СПб.: БХВ-Петербург, 2009. – 314 с.
13. Бождай, А.С. Применение методов интеллектуального анализа данных для реализации рефлексивной адаптации в системах E-Learning / А.С. Бождай, Ю.И. Евсеева, А.А. Гудков // Открытое образование. – 2017. – № 4. – С. 113–120.

УДК 66.081.63

А. Б. Голованчиков, М. И. Филимонов, Доан Минь Кьонг
МОДЕЛИРОВАНИЕ ПРОЦЕССА ОБРАТНОГО ОСМОСА
В ФИЛЬТРУЮЩИХ СВЕРХЦЕНТРИФУГАХ
Волгоградский государственный технический университет
 pahp@vstu.ru

Рассмотрена возможность проведения обратного осмоса в фильтрующих сверхцентрифугах, обеспечивающих центробежное давление в 50–60 атм. На базе дифференциальных уравнений материального баланса получены расчетные формулы для определения производительности по фильтрату, концентрированному раствору и концентрациям в них ионов растворенного вещества. Приведены примеры расчетов для концентрирования водного раствора хлорида кальция в центробежном поле фильтрующей сверхцентрифуги.

Ключевые слова: обратный осмос, фильтрующая сверхцентрифуга, центробежное давление, мембрана, концентрация, исходный и концентрированный раствор, истинная селективность, производительность.

A. B. Golovanchikov, M. I. Filimonov, Doan Minh Cuong
MODELING OF THE PROCESS OF REVERSE
OSMOSIS IN FILTERING SUPERCENTRIFUGES
Volgograd State Technical University

The possibility of reverse osmosis in filtering super centrifuges providing a centrifugal pressure of 50–60 atm is considered. Based on differential equations of material balance, calculation formulas are obtained for determining the productivity of the filtrate, the concentrated solution, and the concentrations of the ions of the dissolved substance in them. Examples of calculations for the concentration of an aqueous solution of calcium chloride in the centrifugal field of the filtering supercentrifuge are given.

Keywords: reverse osmosis, super centrifugal filtering, centrifugal pressure, membrane, concentration, initial and concentrated solution, true selectivity, productivity.

Процесс обратного осмоса требует для своего проведения давлений на мембране 40–60 атм. [1,2]. Такое давление можно реализовать на боковой поверхности ротора в сверхцентрифугах при высоком числе оборотов ротора. В настоящее время это число оборотов может достигать $n = 45000 \text{ об/мин}$ или угловой скорости вращения $\omega = 4,7 \cdot 10^3 \text{ с}^{-1}$ [3].

При плотности $\rho = 1000 \text{ кг/м}^3$, радиусе ротора $R_n = 0,05 \text{ м}$ и радиусе свободной поверхности жидкости $R_c = 0,04 \text{ м}$ давление на поверхности ротора, где может быть установлена полупроницаемая мембрана, определяется по формуле [1–3]:

$$p = \rho \cdot \omega^2 \cdot (R_n^2 - R_c^2) / 2 \quad (1)$$

и имеет численное значение $p = 10^7 \text{ Па} \approx 100 \text{ атм}$. Для рабочих давлений порядка 40–60 атм или $(4 \div 6) \cdot 10^6 \text{ Па}$ угловая скорость вращения ротора сверхцентрифуги должна быть в пределах:

$$\omega = \sqrt{\frac{2 \cdot p}{\rho \cdot (R_n^2 - R_c^2)}} = (2,9 \div 3,7) \cdot 10^3 \text{ (с}^{-1}\text{)}$$

или $n = (28 \div 35) \cdot 10^3 \text{ об/мин}$, то есть сверхцентрифуги могут обеспечить необходимое давление в 40–60 атм. на стенках ротора, где можно закрепить обратноосмотическую мембрану.

На рис. 1 представлена схема ротора фильтрующей сверхцентрифуги с полупроницаемой мембраной на поверхности ротора. Здесь же показаны материальные потоки с концентрациями в них ионов соли на входе и на выходе (обозначения приведены в таблице).

Для математического описания рассматриваемого процесса выделим на высоте z сечения боковой стенки ротора I-I и II-II с расстояниями между ними dz и составим элементарные материальные балансы по основным потокам концентрируемого раствора, фильтрата и концентрациями в них молекул соли на входе и на выходе этих сечений:

$$L = \left(L + \frac{dL}{dz} \cdot dz \right) + dL_p, \quad (2)$$

где $dL_p = G \cdot 2\pi \cdot R_n \cdot dz$;

$$L \cdot x = \left(L + \frac{dL}{dz} \cdot dz \right) \cdot \left(x + \frac{dx}{dz} \cdot dz \right) + x_p \cdot dL_p, \quad (3)$$

где

$$x_p = x \cdot (1 - \varphi). \quad (4)$$

Для расчета удельной производительности G :

$$G = g_w \cdot (1 - p_u / \Delta p) \quad (5)$$

необходимо знать зависимость осмотического давления p_u от концентрации x в растворе. При

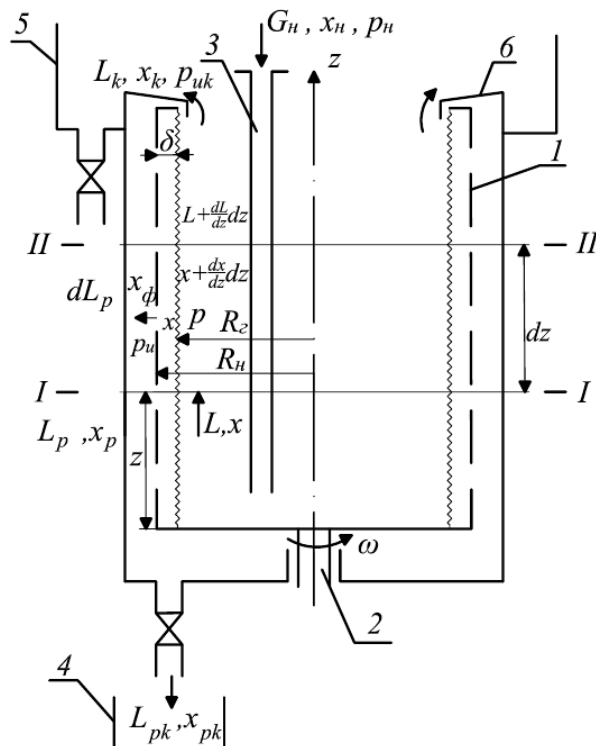


Рис. 1. Схема сверхцентрифуги с полупроницаемой мембраной на роторе для обратноосмотического процесса разделения:

1 – цилиндрическая стенка ротора с мембраной на его боковой поверхности; 2 – вал; 3 – труба для подвода исходного раствора в нижнюю часть ротора; 4 – емкость для сбора пермеата; 5 – емкость для сбора ретанга; 6 – кольцевой козырек над ротором для приема ретанта из ротора

небольших изменениях концентрации соли в начальном растворе x_n и ретанте x_k можно записать [1; 4–6]:

$$p_u = a + b \cdot x. \quad (6)$$

Так, для хлорида кальция эта зависимость с относительной ошибкой не более 0,9 % может быть МНК аппроксимирована при 0,8 % $< x < 3,25$ % в виде:

$$p_u = -0,047 + 0,62 \cdot x.$$

Дифференциальное уравнение (2) после интегрирования с граничными условиями $z = 0$, $L = G_n$ принимает вид:

$$L_p = G_n - L, \quad (7)$$

которое при $z = H$ приводится к интегральному уравнению материального баланса:

$$L_{pk} = G_n - L_k. \quad (8)$$

Дифференциальное уравнение (3) после алгебраических преобразований с учетом формулы (4) преобразуется к виду:

$$\frac{dx}{x \cdot dz} = -\varphi \cdot \frac{dL}{L \cdot dz},$$

и после интегрирования по параметру с учетом

граничного условия $z = 0$, $L = G_n$ приводит к зависимости:

$$L = G_n \cdot \left(\frac{x_n}{x} \right)^{1/\varphi}, \quad (9)$$

которую с учетом формулы (7) можно представить следующим образом:

$$L_p = G_n \cdot \left[1 - \left(\frac{x_n}{x} \right)^{1/\varphi} \right] \quad (10)$$

или в дифференциальном виде:

$$dL_p = \frac{1}{\varphi} \cdot G_n \cdot x_n^{1/\varphi} \cdot \frac{dx}{x^{1+1/\varphi}}.$$

С другой стороны, дифференциальный расход пермеата dL_p между сечениями I-I и II-II высотой dz можно записать в виде (рис. 1):

$$dL_p = 2\pi \cdot R_n \cdot g_w \cdot \left(1 - \frac{a+b \cdot x}{\Delta p} \right) \cdot dz.$$

Приравнявая правые части последних двух уравнений и разделяя переменные получаем:

$$\frac{dx}{x^{1+1/\varphi} \cdot (1 - B \cdot x)} = C \cdot dz.$$

Так как истинная селективность $\varphi \approx 1$, то интеграл $\frac{dx}{x^2 \cdot (1 - B \cdot x)}$ имеет аналитическое ре-

шение [7]. Тогда с учетом граничного условия $z = 0$, $x = x_n$ зависимость текущей концентрации x в ретанте от координаты высоты ротора z принимает вид:

$$z = A \cdot \left(\frac{1}{x_n} - \frac{1}{x} \right) + B \cdot \ln \left[\left(\frac{A}{x} + B \right) / \left(\frac{A}{x_n} + B \right) \right], \quad (11)$$

$$\text{где } A = 1 - \frac{a}{\Delta p}; \quad B = \frac{b}{\Delta p}; \quad C = \frac{2\pi \cdot R_n \cdot g_w \cdot \varphi}{G_n \cdot x_n^{1/\varphi}}.$$

При $z = H$, $x = x_k$.

В этом случае степень концентрирования $K = x_k/x_n$ и формула (10) примет вид при $z = H$:

$$L_{pk} = G_n \cdot \left(1 - K^{-1/\varphi} \right).$$

С учетом материального баланса по молекулам и ионам соли их концентрация в пермеате:

$$x_{pk} = (G_n \cdot x_n - L_k \cdot x_k) / G_{pk}.$$

Были проведены расчеты по исходным и справочным данным для обратноосмотического процесса, приведенным в справочнике [1], но при производительности $G_n = 0,001$ кг/с. Эти данные приведены в таблице, представленной ниже.

Исходные и справочные данные и расчетные параметры обратноосмотического процесса в сверхцентрифуге

№	Наименование параметра	Размерность	Обозначение	Величина параметра
1	2	3	4	5
Исходные данные				
1	Производительность по начальному раствору	кг/с	G_n	0,007
2	Начальная концентрация CaCl_2 в растворе	кг/кг, %	x_n	0,8
3	Плотность раствора	кг/м ³	ρ	1023,7
4	Теплота гидратации ионов кальция	кДж/моль	ΔH_1	1616
5	Теплота гидратации ионов хлора	кДж/моль	ΔH_2	352
6	Число ионов кальция в молекуле CaCl_2	—	n_1	1
7	Число ионов хлора в молекуле CaCl_2	—	n_2	2
8	Удельная производительность мембраны МГА-100	кг/(м ² ·с·МПа)	g_0	0,0014
9	Константы мембраны МГА -100 для расчета истинной селективности	—	a_m	6,7
		—	b_m	3,215
10	Коэффициенты уравнения (6), связывающего осмотическое давление и концентрацию CaCl_2 в растворе	МПа	a	-0,047
		кг/кг, %	b	0,62
11	Угловая скорость вращения ротора	1/с	ω	3200
12	Радиус ротора сверхцентрифуги	м	R_n	0,05
13	Радиус свободной границы раствора внутри ротора	м	R_g	0,04
14	Высота рабочей части ротора	м	H	1

Окончание таблицы

№	Наименование параметра	Размерность	Обозначение	Величина параметра
1	2	3	4	5
Расчетные параметры				
1	Давление на мембране	МПа	p	4,71
2	Перепад давления на мембране	МПа	Δp	4,61
3	Теплота гидратации молекулы CaCl_2	кДж/моль	ΔH	585,03
4	Истинная селективность мембраны МГА-100 для раствора CaCl_2	–	ϕ	0,994
5	Коэффициенты уравнения (11)	МПа	A	1,01
		МПа/кг/кг, %	B	1,34
		м	C	0,547
6	Конечная концентрация раствора на выходе из ротора	кг/кг, %	x_k	1,59
7	Степень концентрирования	–	K	1,988
8	Расход пермеата	кг/с	L_{pk}	$4,9909 \cdot 10^{-4}$
9	Расход ретанта	кг/с	L_k	$5,0091 \cdot 10^{-4}$
10	Концентрация молекул и ионов CaCl_2 в пермеате	кг/кг, %	x_{pk}	$6,913 \cdot 10^{-3}$
11	Средняя скорость фильтрации пермеата через мембрану	м/с	v_f	$1,55 \cdot 10^{-6}$
12	Среднее время пребывания раствора в роторе	с	τ	3855
13	Средняя скорость движения раствора вдоль поверхности мембраны	м/с	v_g	$2,59 \cdot 10^{-4}$
14	Поверхность мембраны	м ²	F_m	0,314

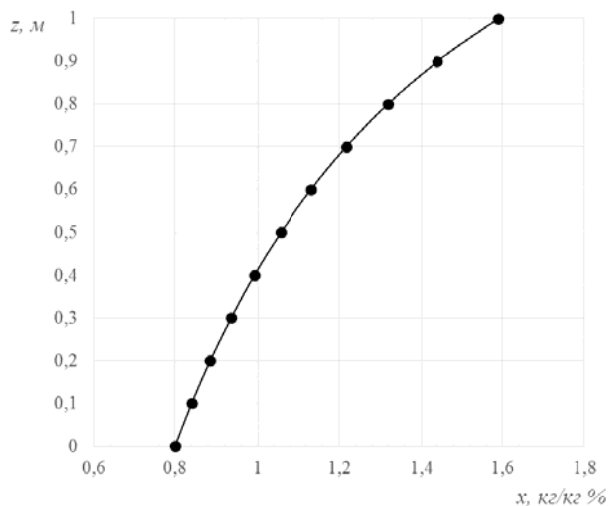


Рис. 2. Профиль концентраций молекул и ионов в растворе CaCl_2 по высоте ротора сверхцентрифуги в обратноосмотическом процессе

На рис. 2 представлен график зависимости концентрации CaCl_2 в растворе по мере его движения вдоль стенки ротора от входа к выходу.

Как видно из этого рисунка с увеличением высоты концентрация молекул и ионов в растворе нарастает нелинейно, хотя до трети высоты ротора носит почти линейный характер. На высоте ротора $H = 1$ и центробежном давлении

$p = 4,71$ МПа степень концентрирования практически удваивается, хотя площадь мембраны всего лишь $0,314 \text{ м}^2$.

Целесообразно проводить сравнение обратноосмотического процесса в сверхцентрифуге с обычным процессом обратного осмоса на мембране рулонного типа по параметру удельной производительности, то есть производительности, отнесенной к единице давления, степени концентрирования и поверхности. В нашем случае:

$$G_y = \frac{G_H}{\Delta p \cdot K \cdot F} = 3,47 \cdot 10^{-4} \frac{\text{кг}}{\text{с} \cdot \text{МПа} \cdot \text{м}^2},$$

что почти в 2 раза меньше, чем для обычного обратноосмотического процесса, приведенного в примере [1] для той же соли при рабочем давлении $\Delta p = 5$ МПа, степени концентрирования $K = 4,1$ и поверхности рулонной мембраны $F \approx 4000 \text{ м}^2$. Это объясняется большой толщиной слоя раствора $\delta = R_n - R_g = 10 \text{ мм}$ по сравнению с обратноосмотическим процессом на рулонной мембране. Уже при уменьшении толщины этого слоя δ с 10 мм до 4 мм $G_y = 6,87 \cdot 10^{-4} \text{ кг}/(\text{с} \cdot \text{МПа} \cdot \text{м}^2)$ с ростом степени концентрирования до $K = 2,425$. При $\delta = R_n - R_g = 2 \text{ мм}$ и рабочем давлении $\Delta p = 0,925 \text{ МПа}$

удельная производительность по раствору возрастает до $G_y = 6,66 \cdot 10^{-4} \text{ кг}/(\text{с} \cdot \text{МПа} \cdot \text{м}^2)$, а степень концентрирования до $K = 5,14$, то есть оба

параметра превышают аналогичные параметры типового обратноосмотического процесса на рулонной полупроницаемой мембране.

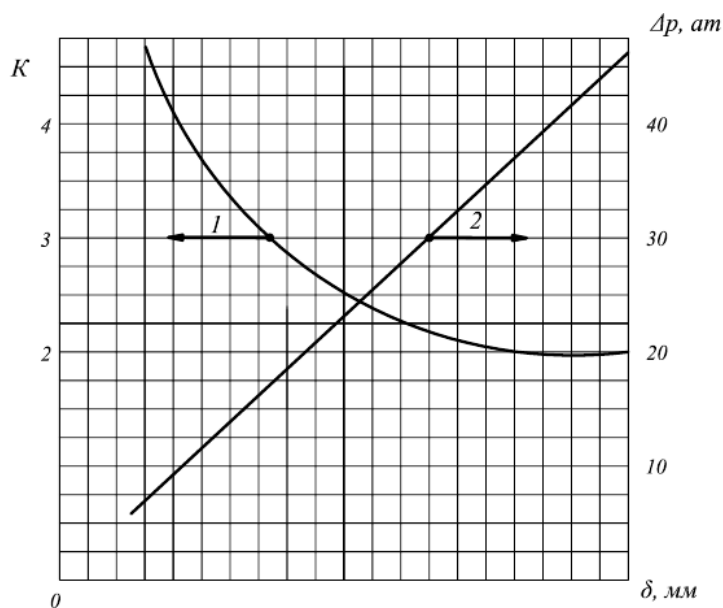


Рис. 3. Зависимости степени концентрирования раствора (1) и центробежного давления (2) на обратноосмотической мембране сверхцентрифуги от толщины слоя раствора на боковой поверхности ротора ($\omega = 3200 \text{ 1/с}$)

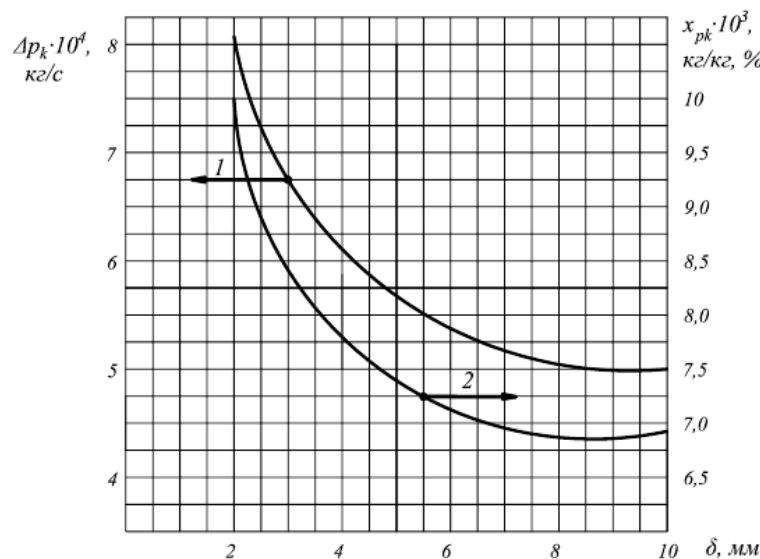


Рис. 4. Зависимости расхода пермеата и концентрации в нем молекул и ионов CaCl_2 от толщины очищаемого раствора на поверхности ротора сверхцентрифуги в обратноосмотическом процессе ($\omega = 3200 \text{ 1/с}$)

Из графиков рис. 3 и 4 видно, что толщина слоя раствора на стенке ротора сверхцентрифуги также сильно влияет на расход пермеата, концентрацию в нем молекул и ионов соли, степень концентрирования и перепад давления. В первых трех случаях эта зависимость носит гиперболический характер, а в последнем опи-

сывается практически прямой линией. К недостаткам рассмотренного процесса относится малая производительность из-за небольшой поверхности мембраны на поверхности ротора.

Таким образом, обратноосмотический процесс в центробежном поле сверхцентрифуги позволяет создавать необходимое рабочее дав-

ление, обеспечивающее высокую степень концентрирования раствора и удельную производительность, превышающие аналогичные технологические показатели типового процесса обратного осмоса на полупроницаемой мембране, но при толщине слоя раствора $\delta = 2 \div 3$ мм.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Основные процессы и аппараты химической технологии : Пособие по проектированию / Ю.И. Дытнерский, Г.С. Борисов, В.П. Брыков и др.; под ред. Ю.И. Дытнерского. – 2-е изд., перераб. и доп. – М.: Химия, 1991. – 496 с.
2. Машины и аппараты химических производств : Учеб. для вузов / А.С. Тимонин, Б.Г. Балдин, В.Я. Борщев, Ю.И. Гусев и др. ; под общ. ред. А.С. Тимониной. – Калуга : Ноосфера, 2014. – 856 с.

3. Касаткин, А.Г. Основные процессы и аппараты химической технологии : Учебник для вузов. – 10-е изд., стер., дораб. Перепечатано с изд. 1973 г. – М.: Альянс, 2004. – 753 с.

4. Развитие мембранных технологий и возможность их применения для очистки сточных вод предприятий химии и нефтехимии / А.Г. Баландина, Р.И. Хангильдин, И.Г. Ибрагимов, В.А. Мартяшева // Нефтегазовое дело. Электронный научный журнал. – 2015. – № 5. – С. 336–375.

5. Свитцов, А.А. Мембранные технологии в России // Химический журнал. – 2010. – № 10. – С. 22–26.

6. Опыт исследования мембранных технологий для очистки и опреснения воды / А.В. Десятов, А.Е. Баранов, Е.А. Баранов, Н.П. Какуркин, А.В. Асеев. – М.: Химия, 2008. – 240 с.

7. Смолянский, М.Л. Таблицы неопределенных интегралов. – 2-е изд., испр. – М.: Гос. изд-во физ.-мат. лит., 1963. – 112 с.

УДК 519.168

А. М. Дворянкин

РЕШЕНИЕ ЗАДАЧИ РАЗМЕЩЕНИЯ НАГРУЖЕННЫХ ОТРЕЗКОВ НА СИСТЕМЕ LINGO

Волгоградский государственный технический университет

dvam1@mail.ru

В работе строится модель и ставится задача размещения отрезков, нагруженных сосредоточенной силой, как задачи нелинейного целочисленного программирования. Критерий оптимального размещения – минимальное отклонение центра тяжести от заданного. Приводится текст программы решения задачи для системы моделирования и оптимизации LINGO.

Ключевые слова: размещение отрезков, сосредоточенная сила, центр тяжести, нелинейное целочисленное программирование, система моделирования и оптимизации LINGO.

A. M. Dvoryankin

THE SOLVING PROBLEM OF LOCATING SEGMENTS LOADED BY FORCE ON SYSTEM LINGO

Volgograd State Technical University

In the paper, a model is constructed and the problem of locating segments loaded with concentrated force as a problem of non-linear integer programming is posed. The criterion of optimal placement is the minimum deviation of the center of gravity from the given one. The text of the program for solving the problem for the LINGO modeling and optimization system is given.

Keywords: placement of segments, concentrated force, center of gravity, nonlinear integer programming, LINGO modeling and optimization system.

1. Введение

Рассматривается модель и задача размещения отрезков, нагруженных сосредоточенной силой, предложенные в работе [1]. Эта задача формулируется как задача нелинейного целочисленного программирования (НЛЦП). Критерий оптимального размещения – минимальное отклонение центра тяжести от заданного. Приводится текст программы решения задачи для системы моделирования и оптимизации

LINGO [2]. Приведены результаты решения задачи на примере.

2. Описание задачи на перестановках

Приведем обозначения и формулировку задачи, предложенные в [1]. Горизонтальный нагруженный отрезок обозначается: $A = (a, p, b)$, где a – расстояние от левого края отрезка до точки приложения силы p , b – расстояние, соответственно, до правого края отрезка. Сила p

интерпретируется как вес отрезка, а точка приложения силы как центр тяжести отрезка. Компоненты отрезка $A = (a, p, b)$ обозначаются также следующим образом: $a = a(A)$, $p = p(A)$, $b = b(A)$. Для обозначения различных отрезков используются нижние индексы. Соединение $A_{ij} = A_i A_j$ двух различных отрезков $A_i = (a_i, p_i, b_i)$ и $A_j = (a_j, p_j, b_j)$ определяется следующим образом:

$$a_{ij} = a(A_{ij}) = [p_i a_i + p_j (a_i + b_i + a_j)] / (p_i + p_j), \quad (1)$$

$$p_{ij} = p(A_{ij}) = p_i + p_j, \quad (2)$$

$$b_{ij} = b(A_{ij}) = [p_j b_j + p_i (a_j + b_j + b_i)] / (p_i + p_j). \quad (3)$$

В работе [1] сформулирована следующая задача размещения отрезков.

Дано: n отрезков $A_1, A_2, \dots, A_n$ и некоторое число c – заданное (целевое) положение центра тяжести для результата соединения этих отрезков. Пусть J_n – множество перестановок чисел $\{1, 2, \dots, n\}$. Любой перестановке $w = (i_1, i_2, \dots, i_n) \in J_n$ можно сопоставить соединение отрезков: $A_w = A_{i_1} A_{i_2} \dots A_{i_n}$. Это означает размещение отрезков в порядке перестановки. Первая компонента такого соединения отрезков есть функция $a(A_w) = a(A_{i_1} A_{i_2} \dots A_{i_n})$ – расстояние от левого края соединенного отрезка до его центра тяжести. Для заданного множества n отрезков и числа c определяется функция δ :

$$\delta(i_1, i_2, \dots, i_n, c) = |a(A_{i_1} A_{i_2} \dots A_{i_n}) - c|. \quad (4)$$

где $(i_1, i_2, \dots, i_n) \in J_n$

Найти: перестановку $(i_1^0, i_2^0, \dots, i_n^0) \in J_n$, на которой достигается минимум функции δ :

$$\delta(i_1^0, i_2^0, \dots, i_n^0, c) = \min_{(i_1, i_2, \dots, i_n) \in J_n} \delta(i_1, i_2, \dots, i_n, c). \quad (5)$$

Перестановка $(i_1^0, i_2^0, \dots, i_n^0)$ определяет оптимальное размещение отрезков с минимальным отклонением центра тяжести от заданного положения.

3. Постановка задачи нелинейного бинарного программирования

Сформулируем задачу размещения нагруженных отрезков в виде задачи нелинейного бинарного программирования. Введем бинарные (булевы) переменные – элементы матрицы перестановок $\|x_{ij}\|_{(i,j=\overline{1,n})}$, в каждой строке и в каждом столбце которой находится лишь одна единица.

Перестановка $(i_1, i_2, \dots, i_k, \dots, i_n) \in J_n$ задается такой матрицей, элементы которой для всех $k = \overline{1, n}$ определяются формулой:

$$x_{kj} = \begin{cases} 1, & j = i_k; \\ 0, & j \neq i_k. \end{cases} \quad (6)$$

Для вычисления центра тяжести отрезка: $A = A_1 A_2 \dots A_n$ воспользуемся формулами утверждения 4 работы [1]:

$$a(A) = \left[\sum_{k=1}^n p_k \left(a_k + \sum_{i=1}^{k-1} (a_i + b_i) \right) \right] / p(A), \quad (7)$$

$$p(A) = \sum_{i=1}^n p_i, \quad (8)$$

$$b(A) = \left[\sum_{k=1}^n p_k \left(b_k + \sum_{i=k+1}^n (a_i + b_i) \right) \right] / p(A). \quad (9)$$

Модифицируем эти формулы для вычисления $a(A_w) = a(A_{i_1} A_{i_2} \dots A_{i_n})$. Для всех $k = \overline{1, n}$ введем обозначения: $\overline{A}_k = (\overline{a}_k, \overline{p}_k, \overline{b}_k)$, где $\overline{a}_k = a(A_{i_k})$, $\overline{p}_k = p(A_{i_k})$, $\overline{b}_k = b(A_{i_k})$. Очевидно, что $\overline{A} = \overline{A}_1 \overline{A}_2 \dots \overline{A}_n = A_{i_1} A_{i_2} \dots A_{i_n} = A_w$.

Тогда для соединения $\overline{A}$ можно применить формулы (7)–(9) вычисления $a(\overline{A}), p(\overline{A}), b(\overline{A})$. Значения $\overline{a}_i, \overline{p}_i, \overline{b}_i$ вычисляются с помощью переменных x_{ij} матрицы перестановок по формулам:

$$\overline{a}_i = \sum_{j=1}^n a_j x_{ij}, \quad \overline{p}_i = \sum_{j=1}^n p_j x_{ij}, \quad \overline{b}_i = \sum_{j=1}^n b_j x_{ij}, \quad i = \overline{1, n}. \quad (10)$$

Подставляя выражения (10) в (7–9) и учитывая выражение (4), получаем постановку задачи размещения отрезков:

$$\left| \left[\sum_{k=1}^n \left(\sum_{j=1}^n p_j x_{kj} \right) \left(\sum_{j=1}^n a_j x_{kj} + \sum_{i=1}^{k-1} \left(\sum_{j=1}^n (a_j + b_j) x_{ij} \right) \right) \right] / \left(\sum_{i=1}^n p_i \right) - c \right| \rightarrow \min \quad (11)$$

при ограничениях:

1) x_{ij} – бинарные переменные:

$$x_{ij} = 0 \vee 1, i, j = \overline{1, n}; \quad (12)$$

2) сумма переменных x_{ij} в каждой i -й строке равна 1:

$$\sum_{j=1}^n x_{ij} = 1, i = \overline{1, n}; \quad (13)$$

3) сумма переменных x_{ij} в каждом j -м столбце равна 1:

$$\sum_{i=1}^n x_{ij} = 1, j = \overline{1, n}. \quad (14)$$

4. Решение задачи размещения нагруженных отрезков на LINGO

LINGO [2] – это инструмент для построения моделей и решения задач математического программирования. LINGO представляет собой интегрированный пакет, который включает: мощный язык для описания оптимизационных моделей, среду для создания и редактирования моделей, набор встроенных решателей оптимизационных задач.

4.1. Программа на LINGO

В табл. 1 представлен листинг программы *Upak 10.lg4* для решения задачи размещения нагруженных отрезков на LINGO.

Таблица 1

Листинг программы

```
MODEL:
    ! Задача размещения нагруженных отрезков;
CALC:
    ! Ввод количества отрезков (N_INDEX) и цели (C) – положения центра тяжести;
    N_INDEX, C = @READLN('Upak_10.lgt',1);
ENDCALC

SETS:
    ! Определение переменных X(I,J) – матрицы перестановок;
    INDEX/1..N_INDEX/;;
    BULPER(INDEX, INDEX): X,Y; ! Булевы переменные X(I,J);
    ! Определение отрезков на перестановке;
    OTR(INDEX): CA,CP,CB,SAB,R,RI;
    ! Определение входных данных об отрезках – матрица MATR(N_INDEX,3);
    INDM/1..3/;;
    MATR(INDEX, INDM): Z;
ENDSETS

CALC:
    ! Чтение матрицы входных данных отрезков;
    Z = @READRM();
    ! SP – общий вес всех отрезков;
    SP=@SUM(INDEX(I): Z(I,2));
    ! SD – сумма длин всех отрезков;
    SD=@SUM(INDEX(I): Z(I,1)+Z(I,3));
    ! Определение значений параметров отрезков на перестановке, определяемой матрицей X;
ENDCALC

SUBMODEL GRANMIN:
    MIN=KRITCEL;
ENDSUBMODEL

SUBMODEL GRANMAX:
    MAX=KRITCEL;
ENDSUBMODEL

SUBMODEL OPTCEL:
```

```

    MIN=DELTA;
ENDSUBMODEL
SUBMODEL BASE:
! Матрица переменных X(I,J) (I,J = 1..N_INDEX) – матрица перестановок;
! Переменные X(I,J) – булевы;
@FOR(INDEX(I): @FOR(INDEX(J):
    @BIN(X(I,J)));
! Сумма X(I,J) по каждой строке I = 1..N_INDEX равна 1;
@FOR(INDEX(I):
    @SUM(INDEX(J): X(I,J)) = 1);
! Сумма X(I,J) по каждому столбцу J = 1..N_INDEX равна 1;
@FOR(INDEX(J):
    @SUM(INDEX(I): X(I,J)) = 1);
! Определение значений параметров отрезков на перестановке, определяемой
матрицей X;
@FOR(INDEX(I):
    CA(I)=@SUM(INDEX(J): Z(J,1)*X(I,J));
@FOR(INDEX(I):
    CP(I)=@SUM(INDEX(J): Z(J,2)*X(I,J));
@FOR(INDEX(I):
    CB(I)=@SUM(INDEX(J): Z(J,3)*X(I,J));
! Вычисление критерия;
SAB(1)=0;
@FOR(INDEX(I) | I #GE# 2:
    SAB(I)=SAB(I-1)+CA(I-1)+CB(I-1));
@FOR(INDEX(I):
    R(I)=CP(I)*(CA(I)+SAB(I));
KRITCEL=@SUM(INDEX(I): R(I))/SP;
DELTA=@ABS(KRITCEL-C);
ENDSUBMODEL

PROCEDURE PRINT_REPORT:
@WRITE('Перестановка                                : ', @NEWLINE(1));
@WRITE(' Отрезки: №          A          P          B', @NEWLINE(1));
@FOR(INDEX(I):
    @FOR(INDEX(J):
        @IFC ( X(I,J) #EQ# 1: @WRITE(@FORMAT(J, '12.3g'),
            @FORMAT( Z(J,1), '12.4f'),
            @FORMAT( Z(J,2), '12.4f'),
            @FORMAT( Z(J,3),
'12.4f'),@NEWLINE(1))));
    ENDPROCEDURE

CALC:
@SET( 'TERSEO',2); ! Уровень выхода 2: только ошибки;
@WRITE('          ЗАДАЧА РАЗМЕЩЕНИЯ НАГРУЖЕННЫХ ОТРЕЗКОВ.', @NEWLINE(1));
@WRITE('Кол-во отрезков                                : ', @FORMAT(
N_INDEX, '7.3g'),@NEWLINE(1));
@WRITE('Суммарная длина всех отрезков                                : ', @FORMAT(SD,
'7.3g'),@NEWLINE(1));
@WRITE('Суммарный вес всех отрезков                                : ', @FORMAT(SP,
'7.3g'),@NEWLINE(1));
@WRITE('Желаемый центр тяжести – цель                                : ', @FORMAT( C,
'12.4f'),@NEWLINE(1));
@WRITE('Начальная нумерация отрезков                                : ', @NEWLINE(1));
@WRITE(' Отрезки: №          A          P          B', @NEWLINE(1));
@FOR(INDEX(J): @WRITE(@FORMAT(J, '12.3g'),
    @FORMAT( Z(J,1), '12.4f'),
    @FORMAT( Z(J,2), '12.4f'),

```

```

@FORMAT( Z(J,3), '12.4f'), @NEWLINE(1));

@SOLVE( GRANMIN, BASE);
@WRITE(' ', @NEWLINE(1));
@WRITE('Минимальное положение центра тяжести           : ',
@FORMAT(KRITCEL, '12.4f'), @NEWLINE(1));
PRINT_REPORT;

@SOLVE( GRANMAX, BASE);
@WRITE(' ', @NEWLINE(1));
@WRITE('Максимальное положение центра тяжести           : ',
@FORMAT(KRITCEL, '12.4f'), @NEWLINE(1));
PRINT_REPORT;

@SOLVE( OPTCEL, BASE);
@WRITE(' ', @NEWLINE(1));
@WRITE('Оптимальмальное положение центра тяжести       : ',
@FORMAT(KRITCEL, '12.4f'), @NEWLINE(1));
@WRITE('Желаемый центр тяжести - цель                   : ', @FORMAT( C,
'12.4f'), @NEWLINE(1));
@WRITE('Отклонение от цели                               : ', @FORMAT( DELTA,
'12.4f'), @NEWLINE(1));
PRINT_REPORT;

ENDCALC
END

```

Программа решает три подзадачи:

- поиск перестановки, на которой достигается $\min$ выражения (7) – SUBMODEL GRANMIN;
- поиск перестановки, на которой достигается $\max$ выражения (7) – SUBMODEL GRANMAX;
- поиск перестановки, на которой достигается $\min$ выражения (11) – SUBMODEL OPTCEL – основная задача.

При написании программы использовались

основные конструкции и операторы языка программирования LINGO: чтение данных из текстового файла, определение переменных и рабочих массивов, организация циклов, формирование и вывод отчета.

4.2. Примеры решения задач

Пример 1. Решение задачи из работы [1]. Исходные данные этой задачи приведены на рис. 1 (файл данных *Upak_10.lbt*). Отметим, что при полном переборе потребуется $5! = 120$ шагов вычисления значений критерия.

5	63		! Количество отрезков и целевое значение центра тяжести;
2	7	8	! Данные отрезков: A P B;
1	3	20	
10	21	30	
25	10	10	
30	12	1	

Рис. 1. Исходные данные задачи 1

На рис. 2 представлены параметры решения задачи – *Solver Status*.

Перечислим значения основных параметров:

- *State: Local Opt* – использовался поиск локального решения;

- *Objective: 0* – значение критерия, найдено глобальное решение;

- *Solver Type: B-and-B* – применялся метод ветвей и границ;

- *Steps: 6* – число шагов при поиске решения равно 6;

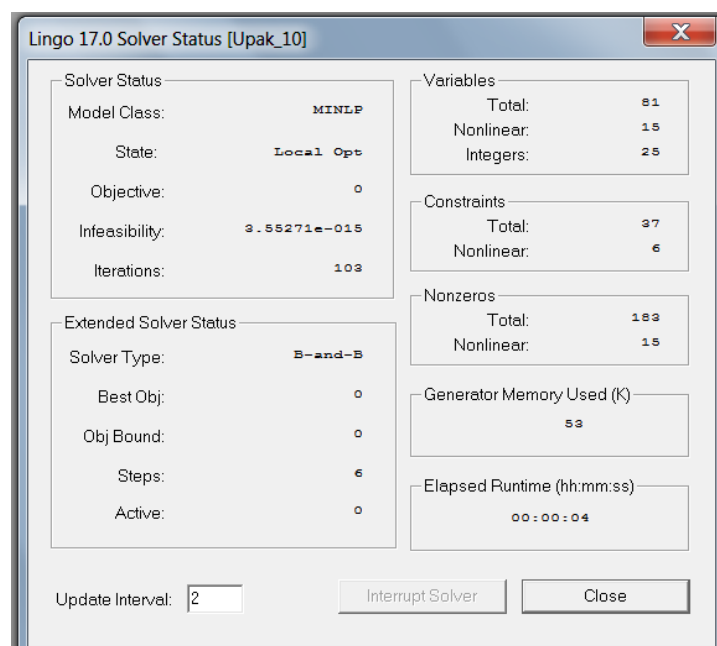


Рис. 2. Параметры решения задачи 1

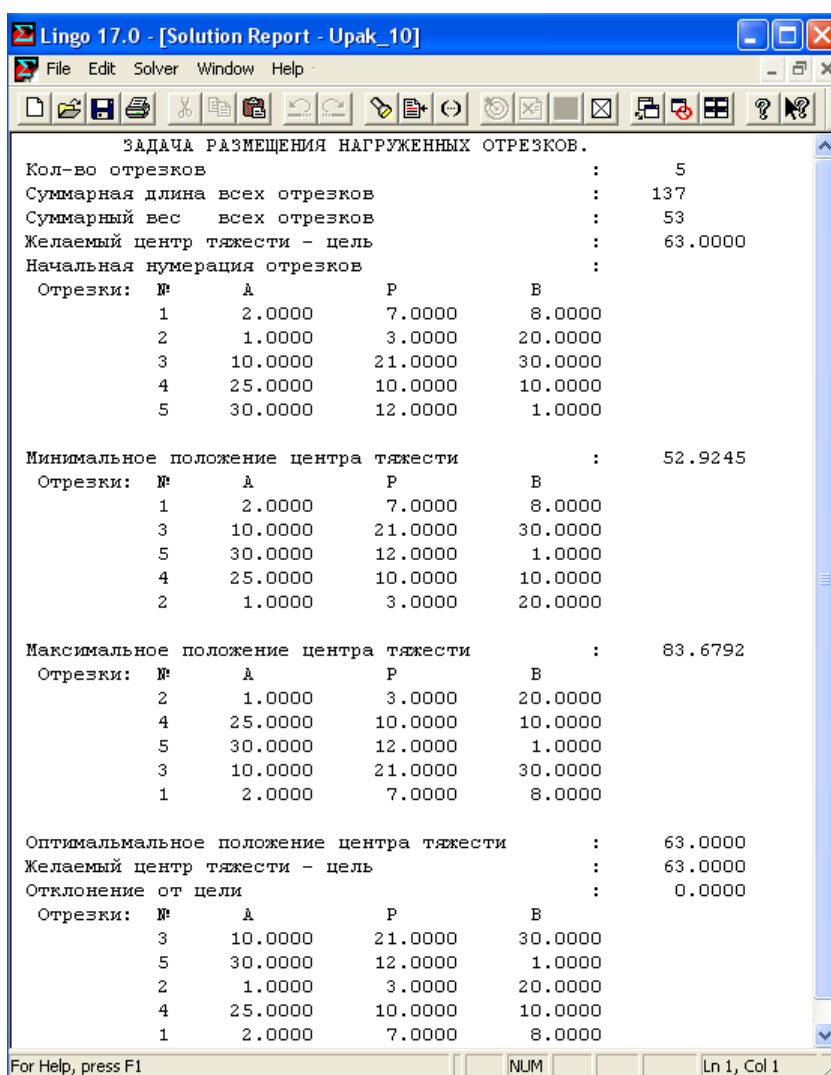


Рис. 3. Результаты решения задачи

Результат решения задачи представлен на рис. 3.

Здесь оптимальное значение критерия, равное 0, достигается на перестановке (3, 5, 2, 4, 1). Центр тяжести полученного размещения отрезков совпадает с целевым.

В табл. 2 приведены результаты решения задачи программой на LINGO и алгоритмом,

предложенным в работе [1]. Назовем реализацию этого алгоритма программой PR1. Этот алгоритм реализован в среде программирования Xbase++ [3]. Вычисления по программам LINGO и PR1 выполнялись на компьютере: производитель MSI, процессор AMD Z-01 1.00 GHz, индекс производительности Windows 7 – 2.8, ОЗУ – 2.00 ГБ, тип системы – 64-разрядная ОС.

Таблица 2

Результаты решения задачи примера 1

Показатель	Программа на LINGO	Программа PR1
min центра тяжести перестановка	52.9245 (1,3,5,4,2)	52.9245 (1,3,5,4,2)
max центра тяжести перестановка	83.6792 (2,4,5,3,1)	83.6792 (2,4,5,3,1)
орт центра тяжести перестановка	63 (3,5,2,4,1)	64.2453 (5,3,4,2,1)
Критерий	0	1.2453
Число шагов	6	11
Время решения задачи	00:00:04	00:00:04

На данном примере с помощью программы на LINGO получено лучшее решение – глобальное решение.

Пример 2. Необходимо разместить 10 одинаковых отрезков разного веса.

Исходные данные: $a_i = b_i = 1, p_i = i$, где $i = \overline{1, 10}$.

Целевое положение центра тяжести **10**. Отметим, что при полном переборе потребуется **10! = 3628800** шагов вычисления значений критерия. В табл. 3 приведены результаты решения этой задачи.

Таблица 3

Результаты решения задачи примера 2

Показатель	Программа на LINGO	Программа PR1
min центра тяжести перестановка	7.0000 (10,9,8,7,6,5,4,3,2,1)	7.0000 (10,9,8,7,6,5,4,3,2,1)
max центра тяжести перестановка	13.0000 (1,2,3,4,5,6,7,8,9,10)	13.0000 (1,2,3,4,5,6,7,8,9,10)
орт центра тяжести перестановка	10.0182 (6,5,8,9,1,4,2,3,10,7)	9.9818 (7,8,6,5,4,3,2,1,9,10)
Критерий	0.0182	0.0182
Число шагов	1062310	46
Время решения задачи	04:40:34	00:00:01

На данном примере существенное преимущество имеет программа PR1.

5. Заключение

Приведена постановка задачи размещения отрезков как задачи нелинейного бинарного программирования: сформулирован критерий задачи и ограничения на переменные. Разработана программа для среды математического моделирования и решения задач оптимизации LINGO. Приведены примеры решения задач

и сравнение результатов их решения с алгоритмом, предложенным в работе [1].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Дворянкин, А.М. Задача размещения нагруженных отрезков / А.М. Дворянкин, М.Б. Кульцова, И.Г. Жукова // Известия Волгоградского государственного технического университета. 2017. № 8 (203). С. 29–34.
2. Сайт LINGO: – URL: <http://www.lindo.com> (дата обращения: 14.05.2018).
3. Сайт Xbase++: – URL: www.alaska-software.com (дата обращения: 14.05.2018).

УДК 623

*О. В. Кислов, Г. И. Ребриков***АНАЛИЗ И СРАВНЕНИЕ РЕЗУЛЬТАТОВ РАБОТЫ
МЕТОДИКИ ВЫДЕЛЕНИЯ ОГИБАЮЩИХ ИЗ СИГНАЛОВ,
РАЗЛИЧНЫХ ПО СТРУКТУРЕ И ЧАСТОТНЫМ СВОЙСТВАМ****4-й Государственный центральный межвидовой полигон МО РФ**
rgeiv2159@yandex.ru

Предлагается методика выделения низкочастотных и низкочастотных параметров в зашумленных информативных сигналах, которая позволяет выделять верхние и нижние огибающие. Приводятся результаты обработки информативных сигналов с различными динамическими и частотными свойствами.

Ключевые слова: вейвлет-функция, выделение огибающей, метод кратномасштабного анализа, обработка сигналов, сглаживание, телеметрические измерения, фильтрация.

*O. V. Kislov, G. I. Rebrikov***ANALYSIS AND COMPARISON OF PERFORMANCE OF METHODS
OF EXTRACTION OF THE ENVELOPE OF THE SIGNALS,
DIFFERENT IN STRUCTURE AND FREQUENCY PROPERTIES****4 SCIP**

The technique of allocation of low-frequency and low-frequency parameters in noisy informative signals, which allows to allocate the upper and lower envelopes, is proposed. The results of processing informative signals with different dynamic and frequency properties are presented.

Keywords: Wavelet-function, definition of bendings, method of multiple scales analysis, signal processing, fitting, telemetry measurements, filtration.

В практике испытаний на межвидовом полигоне анализируемые телеметрические сигналы могут иметь разные особенности поведения, разные спектральные характеристики.

Ранее авторами [1, 2] приводились результаты обработки согласно разработанной методике выделения огибающей информативных сигналов с частотами модуляции, близкой и значительно меньшей, чем частота Найквиста.

Был сделан вывод, что разработанная методика применима ко всем возможным реализациям сигналов, но имеются различия в последовательности операций. Применение кратномасштабного анализа (КМА) [3] с помощью вейвлетов [3, 4] хорошо работает на первом этапе фильтрации для решения задачи выявления тонкой и низкочастотной структуры сигнала, у которого частота модуляций близка к частоте Найквиста. Он обладает возможностью проводить декомпозицию сигналов с различной степенью подробности. При декомпозиции все уровни разложения можно анализировать по отдельности, складывать в любых сочетаниях, выделять компоненты с разными частотами и временными свойствами. При обработке сигналов, у которых частота модуляций значительно меньше частоты Найквиста, метод КМА

хорошо работает на третьем этапе фильтрации после детектирования высокочастотной компоненты при вычислении окончательного вида огибающих [5].

Рассмотрим применение предлагаемой методики для обработки второго вида сигналов для разных процессов, у которых наблюдается различная динамика поведения.

Методика предназначена для выделения огибающих из информативного сигнала, содержащего модуляции и шум наряду с полезным сигналом, несущим информацию о параметре. Целью методики является выделение НЧ-компоненты и подготовка данных для анализа информативных компонент, содержащихся в огибающих сигнала. Информация может содержаться как в верхних, так и в нижних огибающих. Их выделение позволяет оценить характер их поведения, например, достижение установленных уровней, нахождение их в пределах заданных границ и т. п.

Выделение огибающих позволяет провести сравнительный анализ выделенных параметров с другими процессами при проведении ЛИ. Это дает возможность установить взаимные динамические связи анализируемых параметров.

Входными данными для методики являются:

- исходный информативный сигнал с привязкой к шкале времени с постоянным шагом дискретизации;

- минимальный и максимальный размер допустимых интервалов обработки сигнала;

- перечень интервалов, наиболее интересных с точки зрения событийного анализа.

Задание интервалов целесообразно потому, что обычный объем анализируемого телеметрического сигнала может достигать сотен тысяч точек. Обработка такого объема данных нецелесообразна по двум причинам: чрезмерная загрузка вычислительного процессора в ходе обработки и невозможность подбора математического аппарата, одинаково эффективного на разных характерных участках сигнала. Кроме того, всегда имеются моменты времени в ходе

эксперимента, в которых требуется более подробное изучение характера сигнала.

Методика определяет следующие характеристики зарегистрированного процесса:

- среднее значение, усредненное на выбранном интервале (НЧ-компонента);

- верхнюю огибающую процесса, усредненную на выбранном интервале;

- нижнюю огибающую процесса, усредненную на выбранном интервале;

- размах модуляции, как разность значений верхней и нижней огибающих.

Приведем результаты применения методики к сигналам, имеющим различную динамику изменения и сравним их.

Первым обрабатываем сигнал, изображенный на рис. 1.

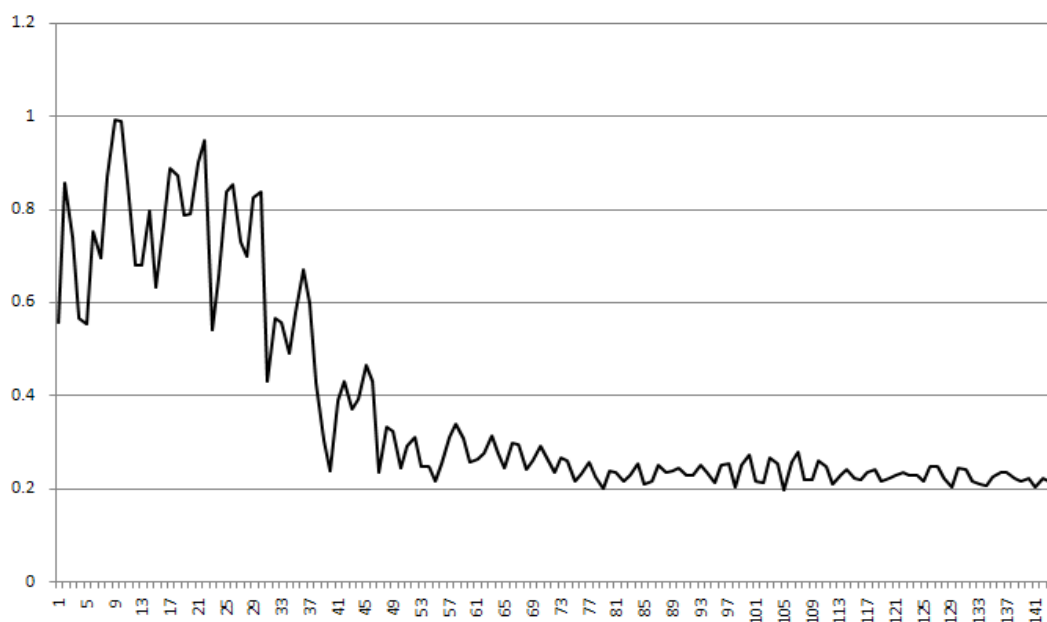


Рис. 1. Сигнал 1 одного из параметров динамического объекта

В методике использована следующая последовательность операций с применением методов цифровой обработки сигналов:

1. Вычисление скользящих средних посредством усредняющей функции с изменяемым весом или вейвлет-функции в зависимости от вида модуляций сигнала.

2. Вычисление периодической составляющей сигнала, как разности исходного сигнала и усредненной функции.

3. Детектирование знакопеременной периодической составляющей, разделение на верхнюю и нижнюю полуволны.

4. Фильтрация протектированных компонент методом свертки с экспоненциальной функцией (аналог RC-фильтрации в радиотехнике).

5. Фильтрация полученных данных во временной области с использованием вейвлет-функций или скользящих средних и получение огибающих.

На рис. 2 приведен результат применения методики к сигналу рис. 1. Видно, как выделена низкочастотная составляющая (пунктирная линия) и огибающие (штриховые линии).

Исследуем применение методики к другому рода сигналу, поведение которого имеет резкие изменения динамики передаваемого параметра.

Пусть задан входной сигнал 2 (рис. 3).

Результаты проведения этапов детектирования и сглаживания, соответствующие разработанной методике [1, 2], приведены на рис. 4.

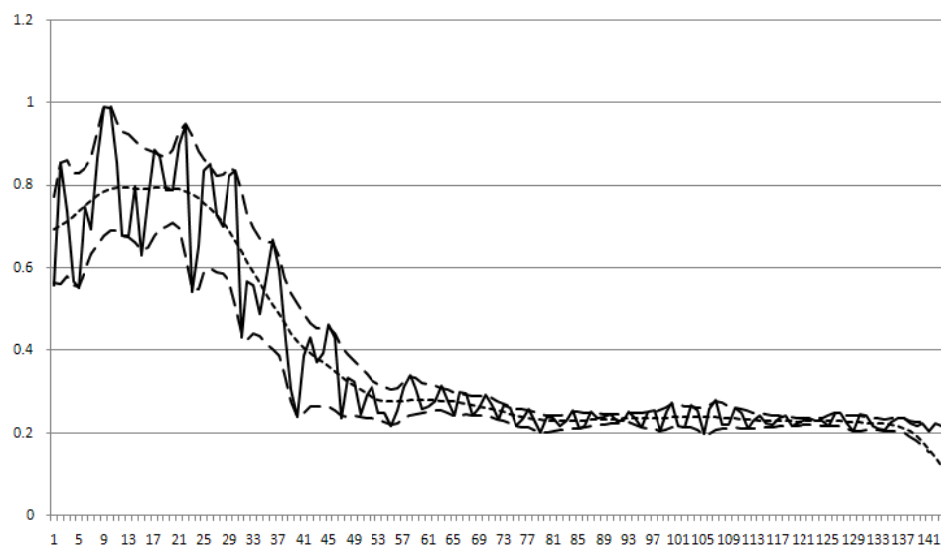


Рис. 2. Применение методики к сигналу 1

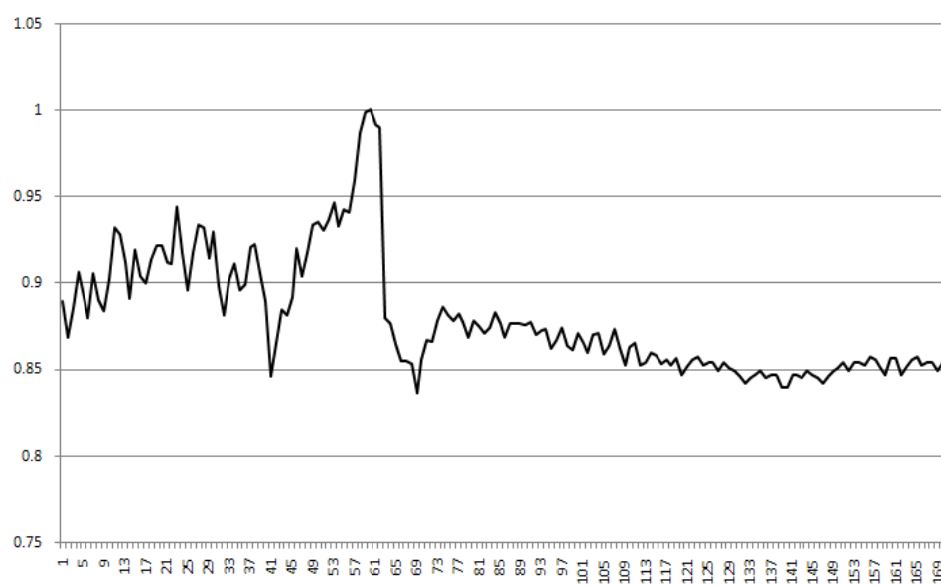


Рис. 3. Сигнал 2 одного из параметров динамического объекта

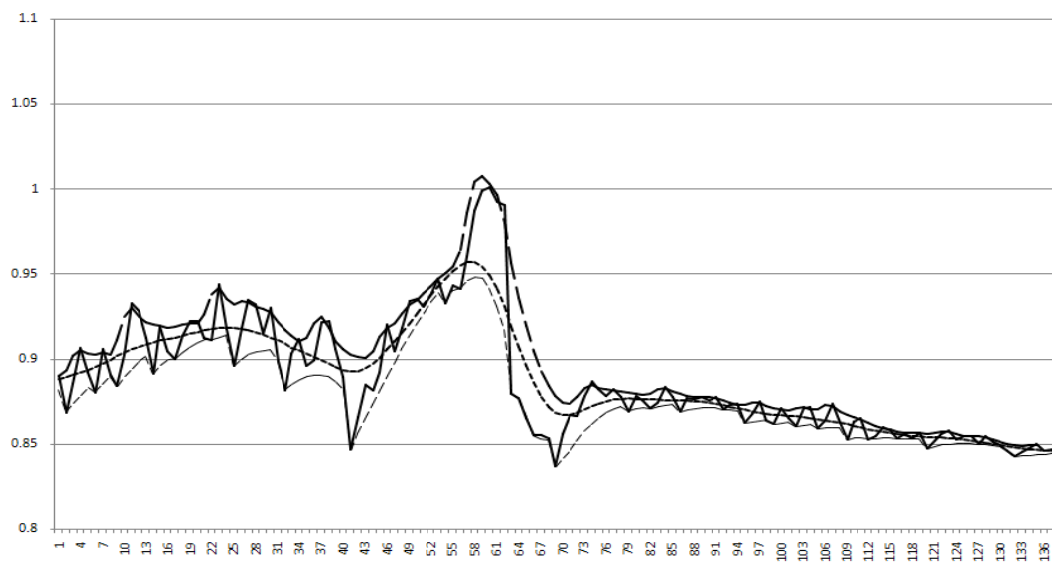


Рис. 4. Применение методики для выявления огибающих сигнала 2

На рис. 4 видно, как выделена низкочастотная составляющая (пунктирная линия) и огибающие (штриховые линии).

Как видно из рисунков, предложенный алгоритм работоспособен для сигналов разного вида с различными особенностями при наличии шумов. Параметры фильтров: Q – количество точек сглаживающего фильтра; q – параметр ширины фильтра Гаусса; K_n – коэффициент инерционности детектирующего RC-фильтра, примененных в данной методике, являются переменными и могут настраиваться анализатором в процессе обработки. Так, параметры Q и q задают гладкость получения усредненной формы основного сигнала. В приведенном примере для сигнала 1 они равны соответственно 21 и 5. Коэффициент RC-фильтра K_n определяет величину инерционности детектирующего фильтра и выбран равным 0,9. Для вейвлет-фильтра, определяющего сглаживание детектированного сигнала, применен вейвлет Добеши 5-го порядка.

Для сравнения, в обработке второго сигнала применены параметры $Q=21$; $q=4$; $K_n=0,8$. Вейвлет-фильтр применен на основе вейвлета Добеши 3 порядка.

В результате проведенных операций выполнен подбор параметров фильтрации и детектирования для сигналов с различной динамикой изменений, проведена декомпозиция медленноменяющихся и быстрых изменений в составе сигнала. Применение методики выполнено в диалоговом режиме для оптимального подбора параметров сглаживания сигнала и параметров усреднения для выявления информативных составляющих сигнала.

Выводы

1. Предлагаемая методика обеспечивает математически строгую обработку исходных сигналов, что позволяет достоверно вычислить и отобразить графически важные информационные составляющие анализируемого процес-

са: текущее среднее значение, верхнюю и нижнюю огибающие. Выделенные компоненты декомпозиции сигнала позволяют провести дальнейший анализ и оценку параметра, а также его искажений, выявить возможные динамические связи модуляций и шумов с движением динамического объекта, условиями измерений параметров, другими наборами данных об объекте исследований.

2. Применение фильтрации с помощью вейвлетов хорошо работает на последнем этапе фильтрации для решения задачи выявления огибающих. Особенностью этого вида фильтров является то, что они не срезают пики протектированного сигнала, а сохраняют их в огибающей, что важно для адекватного отображения параметров объекта.

3. На всех этапах обработки имеется возможность диалогового участия человека-анализатора, что обеспечивает гибкость методики, адаптивность ее для обработки разного вида сигналов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бориско, С. Н., Кислов, О. В., Кобзарь, А. А., Литвинов, С. П., Ребриков, Г. И. Анализ результатов обработки информационных сигналов с частотой модуляции, близкой к частоте Найквиста // Двойные технологии. – Передовые специальные технологии и материалы. – № 3. – 2018.
2. Бориско, С. Н., Кислов, О. В., Кобзарь, А. А., Литвинов, С. П., Ребриков, Г. И. Анализ результатов обработки информационных сигналов с частотой модуляции, значительно ниже частоты Найквиста // Двойные технологии. – Передовые специальные технологии и материалы. – № 3. – 2018.
3. Юдин, М.Н., Фарков, Ю.А., Филатов, Д.М. Введение в вейвлет-анализ : Учебно-практическое пособие для системы дистанционного образования. – Московская государственная геологоразведочная академия. – 2001. – 72 с.
4. Яковлев, А.Н. Введение в вейвлет-преобразования. – Новосибирский государственный технический университет. – Новосибирск, 2003. – 104 с.
5. Глинченко, А.С. Цифровая обработка сигналов. – В 2 ч. Ч. 1. – Красноярск : Изд-во КГТУ, 2001. – 199 с.

СИСТЕМНЫЙ АНАЛИЗ И ОБРАБОТКА ИНФОРМАЦИИ

УДК 004.4

А. Е. Андреев, А. В. Полежаев

РАЗРАБОТКА ВЕБ-СЕРВИСА ДЛЯ СРАВНЕНИЯ ТЕКСТОВЫХ ДОКУМЕНТОВ

Волгоградский государственный технический университет

andan2005@yandex.ru, stunning1505@mail.ru

Был спроектирован и реализован веб-сервис, позволяющий пользователю сравнивать два текста или один текст с базой документов и получать процентное содержание заимствований в тексте. Приложение состоит из двух частей: ASP.NET-приложения для работы с клиентом и WCF-сервиса, который реализует вычислительные методы и работу с базой данных. Приложения разрабатывались под операционную систему Linux для последующего запуска на вычислительном узле кластера ВолгГТУ. Методом сравнения пары тестов был выбран алгоритм шинглов. Была разработана многопоточная версия программы с применением технологий параллельных вычислений на CPU Xeon E5 и XeonPhi. Было произведено сравнение времени вычислений процента схожести текстов на CPU Xeon E5 и XeonPhi в различных ситуациях, после чего разработана оптимальная модель работы приложения. Преимуществами такой системы являются: быстродействие, благодаря вычислениям на высокопроизводительных устройствах и применению методов сокращения времени вычислений; гибкость и портативность, благодаря возможности удаленного доступа с любого устройства, имеющего доступ к сети.

Ключевые слова: веб-сервис, алгоритм шинглов, IntelXeonPhi, ASP.NET, OpenMP.

A. E. Andreev, A. V. Polezhaev

DEVELOPMENT WEB-SERVICE TO COMPARE TEXT DOCUMENTS

Volgograd State Technical University

A web service was designed and implemented that allows the user to compare two texts or one of a number of documents in the text. The application consists of two parts: ASP.NET applications to work with the client and WCF service, which implements computational methods and work with the database. The application was developed under the Linux operating system to run on a cluster of VSTU. As a method of comparison, the shingles algorithm. CPU Xeon E5 and Intel Xeon Phi. It was a comparison of the computation time of the degree of the similarity of texts in the Xeon E5 CPU and Xeon Phi in various situations, and then developed an optimal model of the application. The advantage of this system are: performance, thanks to high-performance computing devices and the application of techniques to reduce computation time, flexibility and portability, due to the possibility of remote access from any device with access to the network.

Keywords: web-service, shingles algorithm, Intel Xeon Phi, ASP.NET, OpenMP.

Введение

В последнее время наблюдается тенденция к усилению контроля качества студенческих работ и, как следствие, самого учебного процесса. С 1 января 2016 года вузы на законодательном уровне обязали проводить проверки выпускных квалификационных работ на наличие заимствований с последующим хранением документов в системе вуза. В связи с этим актуальна тема разработки собственной системы проверки на плагиат. Такая система имела бы

свои недостатки, такие как более низкая функциональность и надежность по сравнению с популярными системами, но при этом обладала бы рядом достоинств: гибкость, адаптивность, расширяемость, а главное, она не требовала бы существенных вложений. Также необходимо учитывать, что на кафедрах уже хранится большой объем документов, которых нет в интернет-базах, и которые можно использовать для проверки текстов в системе проверки на плагиат. Работа направлена на создание веб-сер-

виса, работающего в режиме сравнения двух текстовых документов и сравнения текста с базой документов, а также дополнительных возможностей работы с базой документов: удаление, скачивание файла с базы, добавление нового файла в базу, просмотр информации о документе из базы. В работе необходимо реализовать алгоритм сравнения текстовых документов с использованием многопоточных вычислений на нескольких сопроцессорах IntelXeonPhi 1 поколения (KNC). Также необходимо произвести тестирование алгоритма на центральных процессорах IntelXeon E5 и IntelXeonPhi 2 поколения (KNL) и оценить эффективность их использования.

1. Алгоритм шинглов для сравнения текстов

Шингл в переводе с английского – это ячейка, чешуйка, частичка, кирпичик. В сфере анализа текстов шингл означает те составные части, из которых строится текст. Это понятие впервые ввел в употребление по отношению к методам анализа текстов в 1997 году сотрудник компании «Yahoo!» Андрей Бродер [1].

Алгоритм шинглов – это алгоритм для поиска копий текста и выявления плагиата в тексте. На рис. 1 представлена диаграмма алгоритма шинглов, который включает в себя следующие этапы [1]:

- а) канонизация текста;
- б) разбиение текста на наборы шинглов;
- в) расчет контрольных сумм для шинглов;
- г) сравнение массивов контрольных сумм текстов;
- д) подсчет совпадений и определение результата.

Канонизация приводит текст в единую форму, все предлоги, союзы, знаки препинания и прочее удаляются, так как они не несут смысловой нагрузки. Также в ряде модификаций алгоритма шинглов могут удаляться прилагательные, а существительные приводиться к именительному падежу и единственному числу, либо вообще остаются лишь корни слов [1].

Следующий этап – это разбиение текста на наборы шинглов. Шингл в данном случае – это выделенные из текста последовательности слов определенной длины, которые формируются внахлест. К примеру, если имеется уже канонизированный текст «пришел домой сел компьютер делать уроки завтра» и длина шинглов равна трем, то для него будут следующие наборы шинглов: «пришел домой сел», «сел компьютер делать» и «делать уроки завтра».

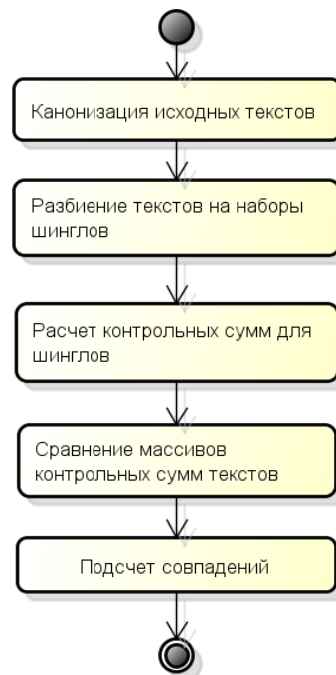


Рис. 1. Схема алгоритма шинглов

Увеличение количества шинглов приводит к экспоненциальному росту количества операций, что сильно отражается на производительности, поэтому в алгоритме сравниваются не сами шинглы в виде текста, а наборы контрольных сумм (хэшей), рассчитанные для каждого шингла при помощи хэш-функций, таких как SHA1, CRC32, MD5 и т. п.

На следующем этапе выполняется сравнение набора рассчитанных контрольных сумм шинглов одного текста с набором другого. Для увеличения скорости проверки могут сравниваться не каждый набор с каждым, а только выбранные по определенному критерию, например, только четные или нечетные значения хэшей, или минимальные значения строки. После подсчета количества совпадений становится известен результат сравнения.

2. Проектирование приложения для сравнения текстов с использованием многопоточных вычислений

Приложение хранит в базе данных не исходные тексты, а рассчитанные наборы контрольных сумм для шинглов текстов. Это сделано для повышения скорости сравнения текстов, так как нет необходимости при каждом новом запуске рассчитывать наборы шинглов для каждого текста.

Имеются следующие варианты для произведения вычислений разрабатываемого сервиса в рамках кластера ВолгГТУ:

- узел с CPUIntelXeonE5 (вычисления на CPU узла);
- узел с CPUIntelXeonE5 с подключенными сопроцессорами IntelXeonPhi1 поколения (KNC) (вычисления на платформах XeonPhi);
- узел с CPUIntelXeonPhi2 поколения (KNL) (вычисления на CPU узла).

При использовании для вычислений на CPU структура приложения будет такой, как показано на рис. 2.

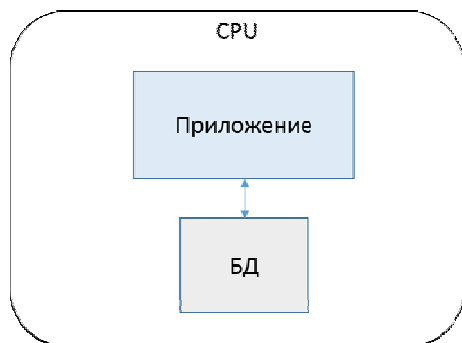


Рис. 2. Схема сравнения текста с базой на CPU узла кластера

Ускорители XeonPhi1 поколения (KNC) с архитектурой MIC поддерживают три режима эксплуатации:

- Offload (режим разгрузки для центрального процессора, в этом режиме происходит ав-

томатическая пересылка данных на ускоритель и обратно);

- Native (режим запуска всей программы на ускорителе из его файловой системы, так как ускоритель представляет собой отдельную вычислительную систему под управлением своей операционной системы; в этом режиме требуется явная пересылка и данных, и программы на ускоритель в виде файлов);

- Symmetric (поддержка одновременных вычислений на MIC и CPU, сложен в реализации).

Последний режим редко используется из-за своей сложности. Первый режим для ускорителей является преобладающим, но для следующего поколения устройств архитектуры MIC основным будет являться режим Native [2].

Для реализации задачи сравнения текста с базой документов наиболее подходящим является режим Native, так как при использовании режима Offload придется передавать большие массивы данных на каждой итерации цикла – наборы контрольных шинглов для каждого файла базы, набор контрольных сумм исходного текста, результат сравнения с каждой пары файлов. При использовании Native можно на каждом ускорителе XeonPhi хранить часть базы документов и вызывать функцию сравнения из сервиса на CPU узла.

Таким образом структура веб-приложения выглядит так, как показано на рис. 3.

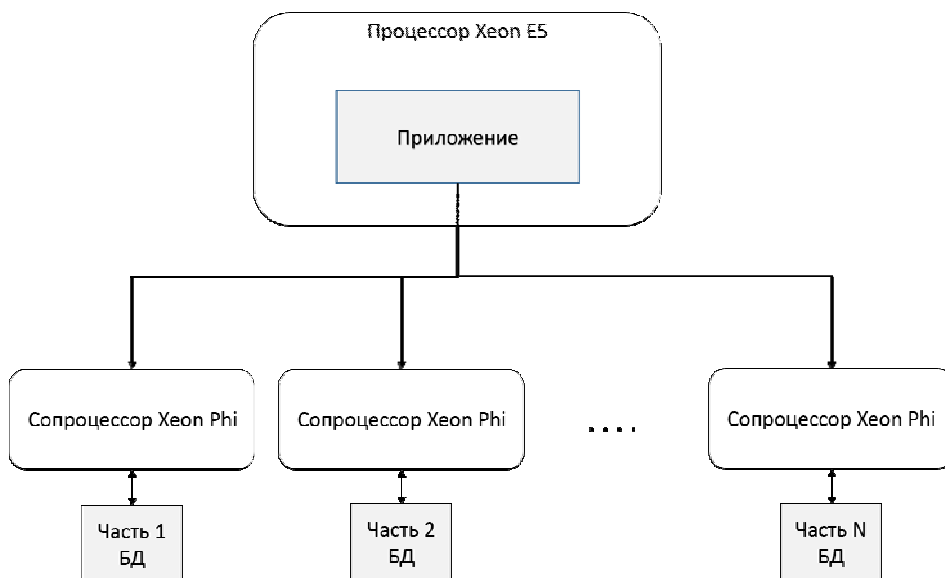


Рис. 3. Схема сравнения текста с базой в Native-режиме XeonPhi 1 поколения

Как уже было отмечено выше, на каждом из сопроцессоров хранится своя часть общей базы данных. Приложение вычисляет набор шинглов для поступившего на вход текста и сохраняет

его в файл, после чего запускает скрипт для копирования этого файла на сопроцессор. Затем вызывается скрипт, выполняющий последовательность команд для сравнения набора шинг-

лов с базой. После окончания сравнения приложение считывает итоговые данные, обрабатывает их и выдает итоговый результат.

3. Реализация приложения для сравнения текстов

Для варианта сравнения текстов на ЦП узла кластера база контрольных сумм хранится в па-

мяти этого же узла. Таким образом для добавления, скачивания и произведения других операций с документами базы не возникает сложностей, в отличие от варианта приложения, использующего вычисления на нескольких платформах XeonPhi 1 поколения. Рассмотрим этот вариант подробнее.

Вызов скрипта для сравнения текста с базой осуществляется при помощи функции:

```
public static string compareShingles(string path, string fileName)
{
    string output = "";
    Process compareProc = new Process ();
    try{
        compareProc.StartInfo.FileName = path;
        compareProc.StartInfo.RedirectStandardOutput = true;
        compareProc.StartInfo.UseShellExecute = false;
        compareProc.StartInfo.CreateNoWindow = true;
        compareProc.StartInfo.Arguments = fileName;
        compareProc.Start();
        StreamReader reader = compareProc.StandardOutput;
        output = reader.ReadToEnd();
    }
    catch(Exception ex) {
        Console.WriteLine (ex.Message);    }
    return output;
}
```

Скрипт запускается при помощи класса Process. Определяется путь к исполняемому скрипту (compareProc.StartInfo.FileName), аргументы (compareProc.StartInfo.Arguments), параметры запуска и др. Результат выполнения скрипта считывается в переменную output типа string.

Скрипт содержит последовательность команд, необходимых для выполнения заданной функции.

Пример скрипта, выполняющего копирование на сопроцессор файла с набором контрольных сумм шинглов для исходного файла:

```
#!/bin/bash
fileName=$1 # Получаем название файла
scp $fileName mic0:/home/user0/TextCompare/Base
```

Имя файла берется из входных аргументов скрипта и сохраняется в переменную fileName.

Ниже приведен пример скрипта, выпол-

няющего сравнение файла с базой документов на сопроцессоре.

```
#!/bin/bash
fileToCmp=$1 # Получаем название файла для сравнения
scp $fileName mic0:/home/user0/TextCompare
ssh mic0
cd /home/user0/TextCompare
LD_LIBRARY_PATH=. ./compare.mic $fileToCmp
```

Исполняемый mic-файл compare.mic выполняет сравнение с базой документа, имя которого хранится в переменной fileToCmp.

Код функции сравнения:

```
int main(int argc, char* argv[])
{
    const char* path = "/home/user0/TextCompare/";
    // Путь к поступившему файлу и каталогу базы
    char* pathFile = new char[100];
    sprintf(pathFile, "%s%s", path, argv[1]);
    char* pathBase = new char[100];
    sprintf(pathBase, "%s%s", path, (char *) "Base/");

    int res = 0;    // Результат
    // Открытие файла
    int* shinglesFile = new int[100000];
    int sizeShFile = openFile(shinglesFile, pathFile);

    vector<const char*> files = getFilesDir(pathBase); // Файлы базы
    int count = files.size();    // Кол-во файлов в каталоге

    const char* resStr = "";
    const char* tmpStr = "";

    for (int i = 0; i < count; i++)
    {
        // Открываем файл базы
        printf("Открытие файла: %s \n", 50, files[i]);
        int* shinglesFileFromBase = new int[10000];
        int sizeShFileFromBase = openFile(shinglesFileFromBase, files[i]);
        printf("Прочли файл, кол-во шинглов = %d\n", sizeShFileFromBase);

        // Сравнение
        res = CmpTwoShingles(shinglesFile, shinglesFileFromBase,
            sizeShFile, sizeShFileFromBase);
        printf("%s : %d\n", files[i], res);
    }
}
```

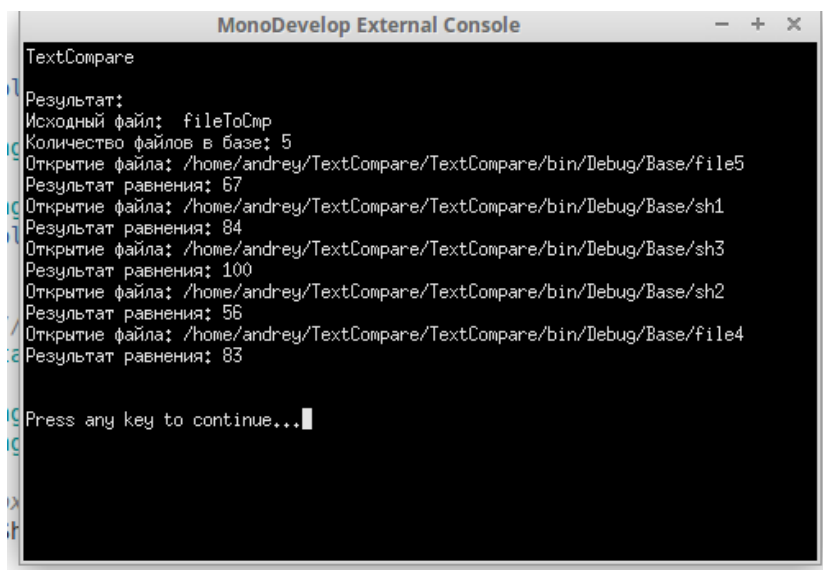


Рис. 4. Пример работы приложения

На рис. 4 представлен скриншот работы приложения.

Имя файла берется из аргумента командной строки. При помощи функции `openFile` происходит считывание наборов хэшей. Функция `getFilesDir` возвращает список имен всех файлов базы, которые в цикле перед сравнением открываются также методом `openFile`. Сравнение двух наборов контрольных сумм производится в функции `CmpTwoShingles`.

Вычисления на CPU Xeon E5 или Xeon Phi 2 поколения будут производиться аналогично вычислениям на каждом отдельном сопроцессоре Xeon Phi 1 поколения.

4. Распараллеливание алгоритма

Для сравнения двух текстов на этапах разбиения канонизированного теста на наборы шинглов, расчета контрольных сумм шинглов и сравнения этих контрольных сумм основные действия производятся в циклах. Поэтому здесь можно внедрить параллелизм данных, используя прагму OpenMP – `#pragma omp parallel for`. Директива `#pragma omp for` сообщает, что при выполнении цикла `for` в параллельном регионе итерации цикла должны быть распределены между потоками группы [4].

Тогда цикл разбиения текста на наборы шинглов и получения контрольных сумм будет выглядеть следующим образом:

```
#pragma omp parallel for
for (inti = 0; i<= (Words.size() - count_shingle); i++)
{
    stringShingleText = "";
    for (int j = 0; j <count_shingle; j++)
        ShingleText += (Words[i + j] + " ");
    int res = crc.FullCRC((const unsigned char*)ShingleText.c_str(),
(size_t)ShingleText.length());
    buf[i] = res;
}
```

Циклы этапа сравнения контрольных сумм:

```
#pragma omp parallel for
for (inti = 0; i<sizeA; i++)
    arrMatches[i] = 0;

#pragma omp parallel for
for (inti=0; i<sizeA ; i++) // string s inShingleA)
{
    for (int j = 0; j <sizeB; j++)
        if ( bufB[j] == bufA[i] ) {
            arrMatches[i]++;
            break;}
}
```

Также в одном из циклов метода сравнения контрольных сумм целесообразно использование прагмы `#pragma omp parallel for reduction`,

которая производит выполнение операций редукции, таких как вычисление глобальной суммы [25]:

```
matches = 0;
#pragma omp parallel for reduction(+:matches)
for (inti = 0; i<sizeA; i++)
    matches += arrMatches[i];
```

Также для повышения скорости сравнения двух наборов хэш-сумм целесообразно использовать векторизацию. Пример функции сравне-

ния двух массивов хэшей с использованием функций-интринсиков для регистров AVX:

```
intCompareShingles(int* bufA, int* bufB, intsizeA, intsizeB)
{
    int *arrMatches;
    arrMatches = (int*) _mm_malloc(sizeA * sizeof(int), 64);

    int matches = 0;

    // #pragma omp parallel for reduction (+: matches)
    #pragma omp parallel for
    for (inti = 0; i < sizeA; i++)
        arrMatches[i] = 0;

    #pragma omp parallel for
    for (inti = 0; i < sizeA; i++) {

        int A[16] __attribute__((aligned(64)));
        for (int j = 0; j < 16; j++) { A[j] = bufA[i]; }
        __m512i first = _mm512_load_epi32(A);

        for(int j = 0; j < sizeB; j += 16) {
            __m512i second = _mm512_load_epi32(&bufB[j]);
            __mmask16 found = _mm512_cmpeq_epi32_mask(first, second);
            if( found ) { arrMatches[i]++; break; }
        }

        matches = 0;
        #pragma omp parallel for reduction(+:matches)
        for (inti = 0; i < sizeA; i++)
            matches += arrMatches[i];

    }

    return 2*100*matches/(sizeA + sizeB);
}
```

Наряду с параллелизмом данных также можно использовать параллелизм задач – параллельное вычисление контрольных сумм для обоих сравниваемых текстов с помощью праг-

мы #pragma parallelsection [4]. В таком случае каждая секция выполняется в отдельном потоке, что позволяет производить декомпозицию по коду:

```
#pragma omp parallel sections
{
    #pragma omp section
    {
        sizeA = GetShingles(Canonize(TextA), ShingleA);
    }
    #pragma omp section
    {
        sizeB = GetShingles(Canonize(TextB), ShingleB);
    }
}
```

Как отмечалось выше, сравнение документа с базой текстов происходит в цикле, поэтому этот процесс также можно распараллелить при помощи прагмы `#pragma omp parallel for`. Таким образом, сравнение каждой пары файлов: исходного и файла из базы будет осуществляться отдельным процессом.

5. Тестирование реализованного приложения

Было произведено тестирование алгоритма на процессорах: IntelXeonE5, IntelXeonPhi 1 поколения (KNC), IntelXeonPhi 2 поколения (KNL) при однопоточном выполнении и многопоточном выполнении с использованием векторизации.

В табл. 1 представлены результаты сравне-

ния двух наборов хэшейшинглов для разных размерностей.

В табл. 2 представлены результаты оценки схожести текста с базой документов разной размерности на IntelXeonPhi 2 поколения. База состоит из наборов контрольных сумм, количество которых приблизительно равно около 5000 в каждом файле. Сравнялся с базой документ, также состоящий из 5000 элементов.

Было получено ускорение более чем в 55 раз для базы из одной тысячи файлов.

В табл. 3 представлены результаты оценки схожести текста с базой документов разной размерности на IntelXeonE5.

Было получено ускорение более чем в 9 раз для базы из одной тысячи файлов.

Таблица 1

Время оценки схожести двух документов

Количество контрольных сумм	IntelXeon E5 (20 потоков)	IntelXeonPhi 1 поколения (228 потоков)	IntelXeonPhi 2 поколения (256 потоков)
1000	0.0002	0.0002	0.0002
2000	0.0002	0.0003	0.0002
5000	0.0003	0.0008	0.0004
10000	0.0053	0.0113	0.0046

Таблица 2

Время оценки схожести документа с базой на IntelXeonPhi2 поколения (KNL)

Количество файлов в базе	Однопоточные вычисления	Многопоточные вычисления (256 потоков)	Ускорение
100	3.138	0.075	41.8
200	5.762	0.121	47.6
500	13.734	0.316	43.5
1000	25.879	0.463	55.9

Таблица 3

Время оценки схожести документа с базой на IntelXeonE5

Количество файлов в базе	Однопоточные вычисления	Многопоточные вычисления (20 потоков)	Ускорение
100	0.253	0.035	7.2
200	0.317	0.083	3.8
500	0.738	0.112	6.6
1000	1.484	0.162	9.2

Таблица 4

Время оценки схожести документа с базой на IntelXeonPhi1 поколения (KNC)

Количество файлов в базе	Однопоточные вычисления	Многopotочные вычисления (228 потоков)	Ускорение
100	3.919	0.064	61.2
200	7.826	0.086	91.0
500	19.570	0.281	69.6
1000	39.114	0.652	59.9

В табл. 4 представлены результаты оценки схожести текста с базой документов разной размерности на IntelXeonPhi 2 поколения.

Было получено ускорение 60 раз для базы

данных, состоящей из 1000 файлов.

В табл. 5 представлено сравнение времени сравнения текста с базой текстов на различных процессорах.

Таблица 5

Время оценки схожести документа с базой на разных процессорах

Количество файлов в базе	Хеон E5 (20 потоков)	IntelXeonPhi1 поколения (KNC) (228 потоков)	IntelXeonPhi2 поколения (KNL) (256 потоков)
100	0.035	0.064	0.055
200	0.083	0.086	0.101
500	0.112	0.281	0.216
1000	0.162	0.652	0.463

Заклучение

Алгоритм шинглов – это алгоритм для поиска копий текста и выявления плагиата в тексте. Он включает в себя следующие этапы: канонизация текста; разбиение текста на наборы шинглов; расчет контрольных сумм для шинглов; сравнение массивов контрольных сумм текстов; подсчет совпадений и определение результата.

Разработанное приложение имеет следующие функциональные возможности: сравнение двух текстов между собой, сравнение текста с базой документов, загрузка, скачивание и удаление документов базы, получение списка всех документов, содержащихся в базе на данный момент, получение информации о каком-либо конкретном файле, которая указывается при загрузке документа на сервер (название файла, тема работы, Ф.И.О. автора, дисциплина и дата выполнения).

Многopotочная версия алгоритма создавалась при помощи технологии OpenMP. Вычисления на центральном процессоре IntelXeon E5 выполняется быстрее приблизительно в 3 раза по сравнению с XeonPhi 2 поколения (KNL) и XeonPhi 1 поколения (KNC). Если сравнивать

KNL и KNC, то расчеты на первом выполняются в 1.5 раза быстрее, однако, на вычислительном кластере в перспективе возможно использование 6 платформ с сопроцессорами 1 поколения параллельно, с разделением БД по ним. Это может увеличить производительность пропорционально количеству ускорителей, то есть в 6 раз, а это приблизительно в 3 раза быстрее, чем на CPU E5.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Алгоритм шинглов – поиск нечетких дубликатов текста : CopyWatcher [Электронный ресурс]. – Режим доступа : http://copywatcher.ru/copyright_statyi/dlya-programmistov-i-vebmasterov/python-algorithm-shinglov-poisk-nechetkih-dublikatov-teksta
2. Сокращение времени оценки схожести текстовых документов на неоднородной многопроцессорной вычислительной системе / С. С. Серов, А. В. Андреев, П. Д. Кравченко, Р. И. Гушин, П. П. Чеботарев // Инженерный вестник Дона. – 2015. – № 2, ч. 2. – С. 98–107.
3. Intel® Xeon Phi™ Coprocessor System Software Developers Guide [Электронный ресурс]. – Режим доступа : <http://software.intel.com/en-us/articles/intel-Xeon-Phi-coprocessor-system-software-developers-guide>
4. Знакомство с уровнями распараллеливания : Блог компании Intel [Электронный ресурс]. – Режим доступа : <https://habrahabr.ru/company/intel/blog/80342/>

УДК: 004.932.72'1

*Е. Г. Крушель, Т. П. Лютая, О. О. Привалов, М. В. Щербаков***РАСПОЗНАВАНИЕ ПОЖАРООПАСНЫХ СИТУАЦИЙ
В ЛЕСНЫХ МАССИВАХ И ЛЕСОПОЛОСАХ****Волгоградский государственный технический университет***elena-krushel@yandex.ru, lyutaya.t.p@gmail.com, galva@mail.ru,
maxim.shcherbakov@gmail.com*

Рассматривается задача повышения эффективности ранней диагностики возникновения пожаров в естественных лесах и зонах искусственного лесоразведения. Рассмотрены и проанализированы существующие способы осуществления мониторинга пожаров в лесных массивах и лесополосах, указаны основания для их совершенствования. Актуальность задачи иллюстрируется кратким анализом состояния лесных массивов Волгоградской области за последние годы.

Подход к решению основан на организации дистанционных наблюдений за состоянием леса и автоматического распознавания очагов возгорания. Для осуществления дистанционного мониторинга предлагается использовать квадрокоптер, оснащенный камерой.

Рассмотрены и проверены экспериментально методы обработки результатов дистанционных наблюдений. Экспериментально подтверждена более высокая эффективность второго метода.

Ключевые слова: лесные пожары, Волгоградская область, нейронная сеть, обнаружение пожаров, распознавание образов, кластерный анализ, мониторинг пожаров, квадрокоптер.

*E. G. Krushel, T. P. Lyutaya, O. O. Privalov, M. V. Shcherbakov***FIRE-DANGEROUS SITUATIONS RECOGNITION
IN THE FOREST AREAS AND FOREST SHELTER BELTS****Volgograd State Technical University**

The problem of the fire occurrence early diagnostics in the natural forests and in the zones of the artificial foresting is discussed. The forest areas and forest shelter belts fire situations monitoring existing methods are overviewed and analyzed, and their improvement purposes are underlined. The brief overview of the recent years Volgograd region forestry condition tendency to worse is presented to show the problem's urgency.

The essence of the proposed approach consists in the forest state distant observations implementation as well as the computer-aided processing of the received images for the purposes of the fireplaces locations recognition. The camera equipped quadcopters facilities are recommended as the perspective mainstream for the distant monitoring system's technical support.

The methods of the distant observations treatment were examined. The numerous computer experiments confirmed the advantages of the second method.

Keywords: forest fires, Volgograd region, neural network, detection of fires, pattern recognition, cluster analysis, monitoring of fires, quadcopter.

Волгоградская область относится к малолесным регионам Российской Федерации. По данным Комитета лесного хозяйства Волгоградской области за 2016 год площадь лесов на территории области составила 473,1 тыс. га, что составляет 4,3 % площади области [1].

В Волгоградской области накоплен опыт (во многом – уникальный) по созданию защитного лесоразведения. По территории области проходят четыре государственных лесополосы и местные полезащитные лесные полосы общей площадью 186 тыс. га. Однако этой площади недостаточно для остановки и дальнейшего предотвращения деградации площади сельхозугодий. По данным Общественной палаты Волгоградской области на 2017 г. площадь деградированных сельхозугодий составляет

8,8 млн. га. Для их восстановления и нормального функционирования требуется 774 тыс. га защитных лесополос (на 36 % больше имеющейся площади) [2].

Природоохранные функции лесов и лесополос очень важны: они улучшают микроклимат, сохраняют влагу, препятствуют размыву поверхности, способствуют повышению урожайности сельхозкультур. В [3] указывается, что «защитные лесные насаждения являются экологическим каркасом территории и одним из основных регуляторов устойчивости геосистем».

Поэтому в Волгоградской области, бедной лесами, особенно важно охранять и приумножать естественную древесно-кустарниковую растительность и искусственные лесопосадки.

В 2017 г. в Волгограде общая площадь лесов составила 13,8 тыс. га. Для того, чтобы поддерживать их в надлежащем состоянии требуется ежегодное финансирование около 60 млн. руб. При этом требуется дополнительно засаживать 130 га деревьями и кустарниками каждый год. Однако фактически на мероприятия по озеленению выделяется 5 млн. руб. в год. За период с 2014 года было обновлено только 400 га.

Помимо естественных причин деградации деревьев (старение) развитию лесов препятствуют неблагоприятные природно-климатические условия для ведения лесного хозяйства на территории области (засухи, суховеи) и связанная с этим высокая пожарная опасность. Так, в 2017 году ГУ МЧС России по Волгоградской области зарегистрировано 8973 случаев возникновения природных пожаров, что на 80 % больше по сравнению с 2016 годом [4].

Борьбу с пожарами осложняет удаленность большей части лесов и лесополос от мест базирования организаций пожаротушения, в связи с чем диагностика начальной фазы возгорания затруднена, и меры по ликвидации пожара часто принимаются после того, как пожар уже распространился на большой площади.

Ниже рассматривается частная, но важная задача повышения эффективности ранней диагностики возникновения пожаров. Подход к решению базируется на организации дистанционных наблюдений за состоянием леса и автоматического распознавания очагов возгорания.

Выбор направления исследования

Известны следующие способы обнаружения лесных пожаров [5]:

1. Визуальный осмотр. Осуществляется человеком, как правило, с использованием наблюдательных вышек. Площадь, которую можно контролировать с одной вышки, равна 5–7 км. Количество наблюдательных вышек, а также трудовых ресурсов в настоящее время недостаточно для охвата всей лесной территории. К тому же, немалую роль при данном способе мониторинга играет человеческий фактор. Некоторые вышки оборудованы видеокамерами, что позволяет увеличить радиус обзора до 30 км, но слежение за территорией по мониторам по-прежнему ведется человеком. В 2016 году на территории Волгоградской области функционировало 39 наблюдательных вышек. Данного количества недостаточно для обеспечения полного обзора площади лесов, равной 473,1 тыс. га.

2. Спутниковый мониторинг. При данном способе изображения со спутников, снятые в высоком или среднем разрешении, загружаются на сервер, где анализируются человеком. Спутники также позволяют обнаружить температурные изменения с помощью датчиков инфракрасного излучения. С помощью спутников проводится непрерывный мониторинг поверхности Земли, но только в рамках орбиты движения конкретного спутника. В настоящее время имеющихся спутников недостаточно, чтобы охватить всю площадь лесных массивов одновременно. К недостаткам спутникового мониторинга также можно отнести высокую стоимость привлечения частных спутников при нехватке государственных; периодичность обновления снимков зависит от скорости движения спутника по орбите; помимо этого, погодные условия, например, сильная облачность, существенно затрудняют наблюдение.

3. Использование самолетов и вертолетов. Данные собираются в реальном времени. Если самолет или вертолет оснащены оборудованием для тушения пожаров, возможна нейтрализация возгорания на ранней стадии. Однако данный способ требует больших материальных затрат.

4. Наблюдение с помощью видеокамер. Данные со всех камер наблюдения сосредотачиваются в наблюдательном пункте. Камеры позволяют обнаружить возгорания в режиме реального времени. Однако площадь лесных массивов слишком велика и материальные затраты на установку необходимого количества видеокамер огромны.

Также в последнее время стало возможным проводить осмотр с беспилотных летающих аппаратов, оснащенных камерой, например, квадрокоптеры. Однако анализ получаемых изображений осуществляется человеком визуально через экран, на который аппарат транслирует данные (например, на экран смартфона или встроенного управляющего элемента – джойстика).

Можно заключить, что общим недостатком существующих способов обнаружения лесных пожаров (в том числе и ориентированных на использование дистанционных средств наблюдения) является недостаточная объективность и оперативность обработки информации и принятия решений. В связи с этим предлагается способ обнаружения участков возгорания на фотографиях лесных массивов в задачах дис-

танционного мониторинга лесных пожаров, позволяющий повысить точность диагностики возгораний за счет оперативной компьютерной обработки входных данных в режиме реального времени.

Существо предлагаемого метода

В качестве устройства дистанционного мониторинга выбран квадрокоптер, оснащенный камерой.

При выборе метода обработки данных были испытаны следующие гипотезы:

1. Выявление участков возгораний на основе кластеризации [6] изображения по цветам.

2. Обнаружение участков с огнем на фотографиях лесных массивов с помощью нейронной сети [7], при обучении которой используются указания не только по цветам, но и по текстуре изображения.

Результаты проверки первой гипотезы представлены на рис. 1–4.

На рис. 1 представлено исходное изображение лесного пожара.



Рис. 1. Исходное изображение для кластерного анализа

После применения алгоритмов кластерного анализа получен следующий результат (рис. 2):

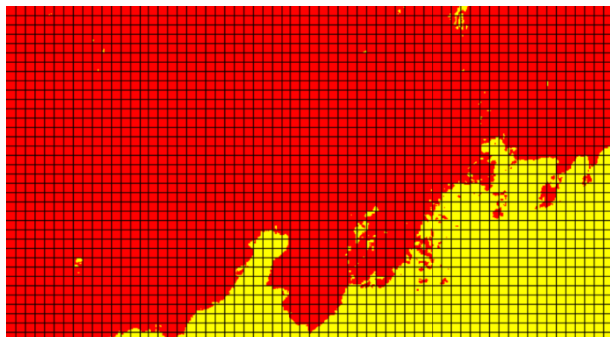


Рис. 2. Результат выполнения алгоритма кластерного анализа

Как видно из рис. 2, область, выделенная после выполнения алгоритма кластерного ана-

лиза, соответствует участку огня на исходном изображении.

Однако при загрузке в качестве исходного изображения фотографии пожара в осеннем лесу (рис. 3) полученные результаты оказались неудовлетворительными (рис. 4). В результате алгоритм распознал не только участки возгорания, но и массивы деревьев, листва которых окрашена в яркие осенние цвета.



Рис. 3. Исходное изображение пожара в осеннем лесу

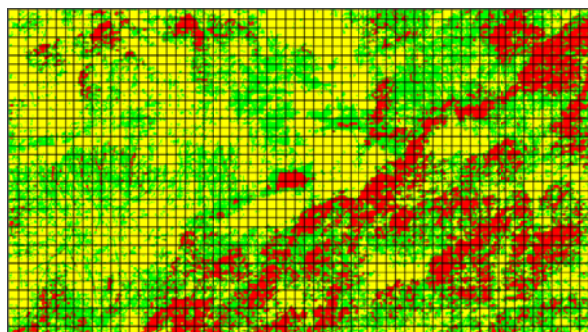


Рис. 4. Результат выполнения алгоритма кластерного анализа фотографии пожара в осеннем лесу

Таким образом, гипотеза 1 может быть использована только в первом полугодии, когда участки возгорания резко отличаются по цвету от зеленых лесных массивов.

Согласно гипотезе 2 предполагается, что успешное распознавание участков возгорания может быть достигнуто после обучения нейронной сети на основе выборки, в которой представлены данные не только о цвете, но и о текстуре изображения, что позволит отделить участки возгорания от листвы осенних деревьев.

Для проверки гипотезы 2 была создана нейронная сеть. В качестве обучающей выборки использовано 270 фотографий лесных массивов как зеленых, так и снятых осенью (с пожаром и без).

После обучения нейронной сети была произведена проверка на 130 фотографиях лесных массивов. Пример работы нейронной сети

представлен на рис. 5 (исходное изображение представлено на рис. 3):

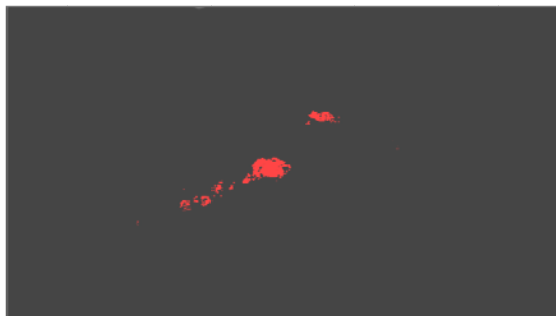


Рис. 5. Пример работы нейронной сети

Анализ точности нахождения участков с огнем произведен путем сравнения количества пикселей в маске изображения, полученной программно и заранее заданной пользователем. Расчет произведен по формуле:

$$Otkl = \frac{\sum_n^1 \frac{(F-P)}{F}}{n}, \quad (1)$$

где $Otkl$ – относительное отклонение; F – количество пикселей красного цвета в полученной маске; P – количество пикселей красного цвета в маске, прорисованной заранее вручную для тех же изображений; n – количество изображений тестовой выборки ($n = 130$); $Otkl = 0.93$.

Таким образом, обученная нейронная сеть успешно распознает участки с огнем на фотографиях.

Для использования результатов в системах экологического мониторинга состояния лесных массивов необходимо решить ряд технических задач, в частности: рассчитать необходимое количество летательных аппаратов для осуществления мониторинга пожаров в лесах Волго-

градской области; определить маршруты их движения, а также подсчитать финансовые затраты и сравнить их с затратами на другие способы мониторинга лесных массивов.

В результате исследования экспериментально подтверждена работоспособность распознавания участков возгорания лесных массивов на основе обработки результатов дистанционных наблюдений обученной нейронной сетью.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Отчет Комитета лесного хозяйства Волгоградской области об итогах деятельности за 2016 год [Электронный ресурс]. Комитет лесного хозяйства Волгоградской области. – 2017. – Режим доступа : <http://klh.volgograd.ru/current-activity/reports/>.
2. Общественники призвали возродить защитные леса Волгоградской области [Электронный ресурс]. Общественная палата Волгоградской области. – 2017. – Режим доступа : <http://opvo.volgnet.ru/news/161484/>.
3. Кулик, А. В. Влияние защитных лесных насаждений на эколого-хозяйственное состояние территории Волгоградской области : автореф. дис. канд. сельскохозяйств. наук. 06.03.04 Всероссийский научно-исследовательский институт агролесомелиорации / А. В. Кулик. – Волгоград, 2008. – 15 с.
4. С начала года в Волгоградской области произошло почти 9000 ландшафтных пожаров [Электронный ресурс] – Волгоград : Волга-Каспий информационное агентство. – 2017. – Режим доступа : <https://volga-kaspiy.ru/known/s-nachala-goda-v-volgogradskoy-oblasti-proizoshlo-pochti-9000-landshaftnyh-pozharov.html>.
5. Смирнов, А. П. Лесная пирология : учеб. пособие / А.П. Смирнов, Е.С. Мельников, А.А. Смирнов. – СПб.: СПбГЛТУ, 2010. – 96 с.
6. Бериков, В.Б. Современные тенденции в кластерном анализе / В.Б. Бериков, Г.С. Лбов // Всероссийский конкурсный отбор обзорно-аналитических статей по приоритетному направлению «Информационно-телекоммуникационные системы». – 2008. – 26 с.
7. Бессмертный, И.А. Системы искусственного интеллекта : Учебное пособие для академического бакалавриата / И.А. Бессмертный. – 2-е изд., испр. и доп. – М.: Юрайт, 2018. – 130 с.

УДК 004.42
X 20

Е. Г. Крушель, А. Э. Панфилов, И. М. Харитонов

ТРЕНДОВАЯ МОДЕЛЬ ПРОГНОЗИРОВАНИЯ КАЧЕСТВА ПОДГОТОВКИ ВЫПУСКНИКОВ ВУЗА НА ПРИМЕРЕ КТИ (ФИЛИАЛА) ВолгГТУ

Камышинский технологический институт (филиал)

Волгоградского государственного технического университета

elena-krushel@yandex.ru, pansanja@yandex.ru, wisdom_monk@mail.ru

Исследуется модель прогнозирования качества подготовки выпускников вуза. Методика прогнозирования базируется на использовании трендовой модели. На основе имеющихся данных о рейтинговых оценках выпускных работ бакалавров и дипломных работ инженеров направления 09.03.01 – «Информатика и вычислительная техника» Камышинского технологического института (филиала) Волгоградского государственного технического университета строятся разные виды трендовых моделей и рассчитываются их основные характеристики. Выбирается наиболее подходящая модель, на основе которой делается прогноз будущей успешной работы выпускников. Формулируется проблема повышения успеваемости и предлагаются пути ее решения.

Ключевые слова: трендовая модель, прогнозирование, эффективность подготовки выпускников.

E. G. Krushel, A. E. Panfilov, I. M. Kharitonov

THE TREND FORECASTING MODEL OF THE HIGH SCHOOL GRADUATES' PERFORMANCE QUALITY BY THE EXAMPLE OF KTI (BRANCH OF VSTU) DATA SOURCES HANDLING

**Kamyshin Technological Institute (branch)
Volgograd State Technical University**

The forecasting model of the high school graduates' education quality is presented. The forecasting method is based on the trend model application. The available rating estimations of the bachelors' graduate works and engineers' graduation projects of the Kamyshin technologic higher education institute (branch of the Volgograd state technical university) students were handled for the different trend models identification as well as for their main characteristics estimation. The suitable model was chosen and applied for the future graduates' successful activity prediction. The problem of the total academic performance improvement is discussed, and the certain ways of its implementation are proposed.

Keywords: trend model, forecasting, graduates' education efficiency.

Введение

Вопросы повышения эффективности процесса ввода выпускников вузов в сферу профессиональной деятельности приобрели особую актуальность в условиях рыночной экономики в связи с неполным соответствием предшествующих моделей подготовки специалистов новой экономической модели России [1]. В сфере высшего образования наметились и стали быстро развиваться кризисные явления:

1. Значительно увеличился спрос населения на получение высшего профессионального образования. Повышенный спрос оказалось возможным удовлетворить за счет коммерциализации вузов. Это привело к росту как их числа, так и контингента студентов и, как следствие, — к увеличению количества молодых людей с высшим образованием и связанной с этим жесткой конкуренцией за рабочие места.

2. Руководители предприятий стремятся принимать на работу специалистов, уже подготовленных к сфере их деятельности. Но выпускник при прежней технологии получения образования чаще всего не обладал нужными умениями и навыками на момент приема на работу. Высокие требования со стороны руководителей предприятий к умениям выпускников еще более усложняют их положение на рынке труда.

3. В частности, для технических специальностей и направлений, непрерывно и быстро растущий уровень сложности техники и технологий ведет к тому, что ознакомление студентов с основами работы инженера в различных сферах (вынужденно поверхностное, из-за лимита учебного времени) становится неэффективным. Большая часть приобретенных знаний на конкретном месте работы останется невостребованной, а нужные для данной работы знания окажутся недостаточными. Данные факто-

ры можно трактовать как признаки недостаточной эффективности использования образовательных ресурсов вуза.

Преодоление кризисных явлений осуществляется в рамках реформы образования. В настоящее время в вузах происходит переход к технологии обучения, соответствующей требованиям Федеральных государственных стандартов нового поколения, что вызывает необходимость проведения исследований и изменения методик планирования учебного процесса, с целью повышения его эффективности [2].

Важной задачей для формирования грамотных учебных планов является исследование качества подготовки выпускников вуза. Наиболее эффективными данными в таком исследовании являются рейтинговые оценки будущих выпускников вуза.

Для прогнозирования рейтинговых оценок авторами работы предлагается использовать трендовую модель с предварительной обработкой данных и выявлением аномальных значений.

Понятие трендовой модели

Обычно для представления динамических процессов, протекающих в технических, экономических и других системах, используют последовательность (ряд) значений некоторого показателя, упорядоченного по времени. Подобный ряд значений показателя отражает ход развития изучаемого процесса. Последовательность наблюдений одного показателя (признака), упорядоченная в зависимости от последовательно возрастающих или убывающих значений другого показателя, называется динамическим рядом, или рядом динамики. Если в качестве признака, в зависимости от которого происходит упорядочивание, берется время, то такой динамический ряд называется временным рядом.

Тренд – это долговременная компонента ряда динамики. Она характеризует основную тенденцию его развития. Тренд характеризует общее направление развития рассматриваемого процесса. Если математическая модель развития изучаемой системы описывается с учетом тренда ее основных показателей, то такая модель называется трендовой моделью [3].

Предварительный анализ данных. Перед построением трендовой модели необходимо произвести предварительный анализ данных. В ходе предварительного анализа определяют:

- соответствие данных требованиям, предъявляемым к ним математическими методами;
- строят график динамики и рассчитывают основные динамические характеристики (приросты, темпы роста, темпы прироста, коэффициенты автокорреляции).

К процедурам предварительного анализа относятся:

- выявление аномальных наблюдений;
- проверка наличия тренда;
- сглаживание временных рядов;
- расчет показателей динамики экономических процессов [4].

Этапы построения прогноза по временным рядам. Экстраполяционное прогнозирование процессов (прогнозирование «на будущее»), представленных одномерными временными рядами, сводится к выполнению следующих основных этапов:

- предварительный анализ данных;
- построение моделей: формирование набора аппроксимирующих функций (кривых роста) и численное оценивание параметров моделей;
- проверка адекватности моделей и оценка их точности;
- выбор лучшей модели.

Выявление аномальных наблюдений. Аномальными уровнями считаются такие значения временного ряда, которые «сильно выбиваются» из общей тенденции. Их наличие существенно искажает основные статистические характеристики ряда, в том числе и соответствующую трендовую модель. Причинами аномальных значений часто являются ошибки технического характера (сбои при фиксации и передаче данных, ошибки при вычислениях). Так как наличие аномальных наблюдений приводит к искажению результатов анализа данных, то необходимо убедиться в отсутствии аномалий. Для диагностики аномальных наблюдений разработаны различные критерии, например метод Ирвина.

Согласно критерию Ирвина, аномальной считается точка Y_t отстоящая от предыдущей точки Y_{t-1} на величину, большую среднеквадратичного отклонения. Для этого рассчитывается критерий Ирвина:

$$\lambda_t = \frac{|Y_t - Y_{t-1}|}{\sigma} \quad (1)$$

$$\sigma = \sqrt{\frac{\sum_{t=1}^n (Y_t - \bar{Y})^2}{n-1}}, \quad (2)$$

где σ – среднеквадратичное отклонение, рассчитываемое по формуле

$$\bar{Y} = \frac{1}{n} \sum_{t=1}^n Y_t, \quad (3)$$

где $\bar{Y}$ – среднее значение временного ряда.

Точка считается аномальной, если $\lambda_t > \lambda_{\text{таб}}$. Табличные значения $\lambda_{\text{таб}}$ уменьшаются с ростом длины ряда.

Проверка наличия тренда. Тенденция (тренд) в развитии исследуемого показателя прослеживается не только в увеличении или уменьшении среднего текущего значения временного ряда. Она присуща и другим его характеристикам: дисперсии, автокорреляции, корреляции с другими показателями. Тенденцию среднего визуально можно определить из графика исходных данных. Проверка наличия или отсутствия неслучайной (и зависящей от времени t) составляющей сводится к проверке гипотезы о неизменности среднего значения временного ряда. Процедура проверки может быть осуществлена с помощью различных методов, например метода проверки разностей средних уровней или метода Фостера – Стиюарта.

Сглаживание временного ряда. Сглаживание временного ряда, т. е. замена фактических уровней расчетными значениями, имеющими меньшую колебательность, чем исходные данные, является простым методом выявления тенденции развития. Соответствующее преобразование называется фильтрованием.

Существующие методы сглаживания делят на две группы:

- аналитические методы. Для сглаживания используется кривая, проведенная относительно фактических значений ряда так, чтобы она отображала тенденцию, присущую ряду, и одновременно освобождала его от мелких незначительных колебаний. Такие кривые называют еще кривыми роста, применяются они главным образом для прогнозирования экономических показателей;

– методы механического сглаживания. Сглаживается каждый отдельный уровень ряда с использованием фактических значений соседних с ним уровней. Для сглаживания временных рядов часто используются методы простой и взвешенной скользящей средней, экспоненциального сглаживания.

Виды трендовых моделей. Тренд – основная тенденция изменения временного ряда. Тренды могут быть описаны различными уравнениями – линейными, экспоненциальными, логарифмическими, полиномиальными, степенными.

При выборе уравнения тренда необходимо руководствоваться принципом простоты, который заключается в выборе из нескольких типов трендов более близкого к эмпирическим данным, наиболее точно отражающего динамику исходного временного ряда, при этом следует выбирать более простую функциональную зависимость. Обоснованно это еще и тем, что чем сложнее уравнение линии тренда и чем большее число параметров оно содержит, тем при равной степени приближения труднее дать надежную оценку этих параметров [5].

При подборе подходящей функциональной зависимости, иначе спецификации тренда, весьма полезным является графическое представление временного ряда.

Тренд, отражая действие долговременных факторов, является определяющим при построении долговременных прогнозов.

Исследование качества подготовки выпускников вуза

За последнее время в вузе наблюдается падение уровня подготовки выпускников, что подтверждается результатами отчетов государственной аттестационной комиссии о результатах защит выпускных работ бакалавров и дипломных проектов инженеров [6].

На рис. 1 представлены данные из отчетов государственной аттестационной комиссии, на примере направления 09.03.01 «Информатика и вычислительная техника» для Камышинского технологического института (филиала) Волгоградского государственного технического университета.

Данные взяты за 9 лет с 2006 по 2014 гг. На первом графике, который находится сверху справа от таблицы значений результатов подготовки бакалавров, представлены данные по бакалаврам в процентном отношении оценок ко всей группе в определенные годы защиты выпускной работы.

Аналогично представлен второй график по подготовке инженеров, который находится справа от таблицы значений результатов подготовки инженеров. Там также представлены года и процентные соотношения отличных оценок, хороших оценок удовлетворительных оценок.

На графиках черным жирным цветом выделены удовлетворительные оценки, пунктиром – хорошие и штрихами – отличные.

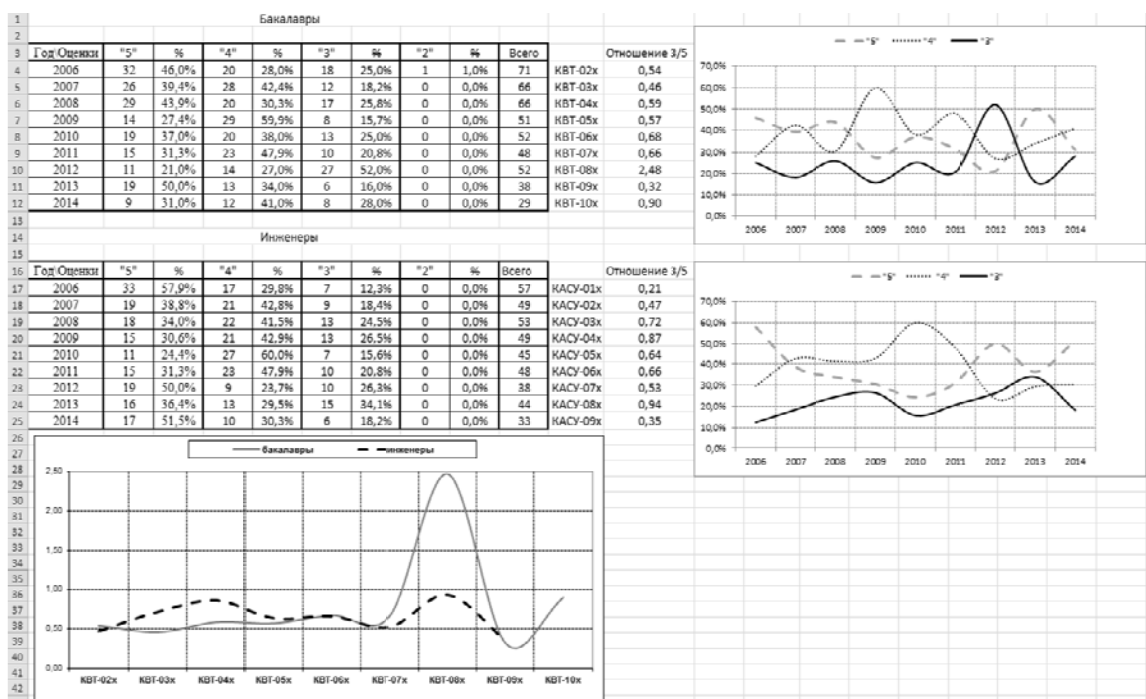


Рис. 1. Уровень дипломных работ

На самом нижнем графике показано отношение удовлетворительных оценок к отличным. Из графика видно, что за последние годы коэффициент увеличивается, а следовательно, процент удовлетворительных оценок растет. Все это говорит о необходимости частичной корректировки имеющихся методов формирования учебных планов, и рассмотреть вопрос о формировании системы индивидуальных учебных планов.

Все это говорит о необходимости частично пересмотреть имеющиеся «общие» учебные планы, ориентированные на всю группу в целом, и рассмотреть вопрос о формировании индивидуальных учебных планов для групп студентов с разной успеваемостью.

«Отличникам» составить учебные планы с максимальной наполненностью трудоемкими дисциплинами, чтобы на выходе получить из них отличных специалистов высшего звена, способных продвигать отечественную науку, экономику и промышленность. Наиболее подходящими для этой группы студентов будут являться учебные планы, построенные на основе академического бакалавриата, позволяющего сильным студентам получать хорошую теоретическую базу, а также продолжать обучение в магистратуре и аспирантуре.

«Хорошистам» предложить несколько упрощенные учебные планы с меньшим количеством трудных дисциплин, позволяющим сформировать грамотных будущих работников среднего звена, но с возможностью добавления каких-ли-

бо отдельных групп дисциплин в виде факультативных занятий, позволяющих освоить один или несколько наиболее приглянувшихся специальностей будущей трудовой деятельности.

«Троечникам» же составить учебные планы с минимальной наполненностью наукоемкими дисциплинами, уделив большую часть времени практическим дисциплинам для выработки определенных навыков работы «руками», которые пригодятся им в дальнейшей трудовой деятельности. Учебные планы для данной группы студентов наиболее подходят под задачи прикладного бакалавриата, ориентированного на выпуск «студентов-практиков».

Построение трендовой модели работ выпускников

На примере бакалавриата была построена трендовая модель, позволяющая прогнозировать будущую ситуацию. Были построены следующие тренды: линейный, логарифмический и квадратичный. Для каждого из этих трендов были подсчитаны: критерий отклонения и относительные ошибки. В качестве исходных данных для построения трендов использовались два типа данных: обычные данные и данные с программно-устраненными аномалиями.

На рис. 2 представлен линейный тренд с аномальными значениями и без них. Критерий отклонения с аномалиями равен 3,617, относительная ошибка – 25,30 %. Критерий отклонения с устраненными аномалиями равен 1,328, относительная ошибка – 22,74 %.

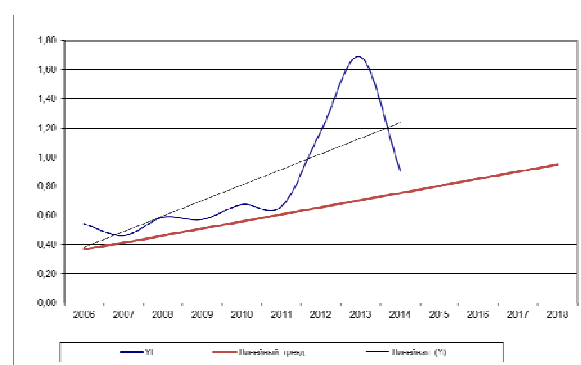
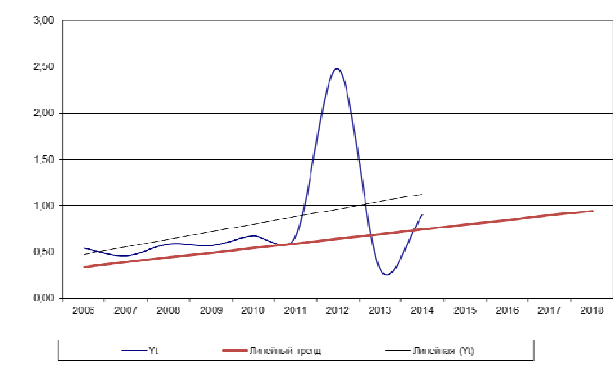


Рис. 2. Линейный тренд

На рис. 3 представлен логарифмический тренд с аномальными значениями и без них. Критерий отклонения с аномалиями равен 2,979, относительная ошибка – 23,24 %. Критерий отклонения с устраненными аномалиями равен 0,727; относительная ошибка – 16,82 %.

На рис. 4 представлен квадратичный тренд с аномальными значениями и без них. Критерий отклонения с аномалиями равен 2,902, относительная ошибка – 22,93 %. Критерий отклонения с устраненными аномалиями равен 0,556, относительная ошибка – 14,71 %.

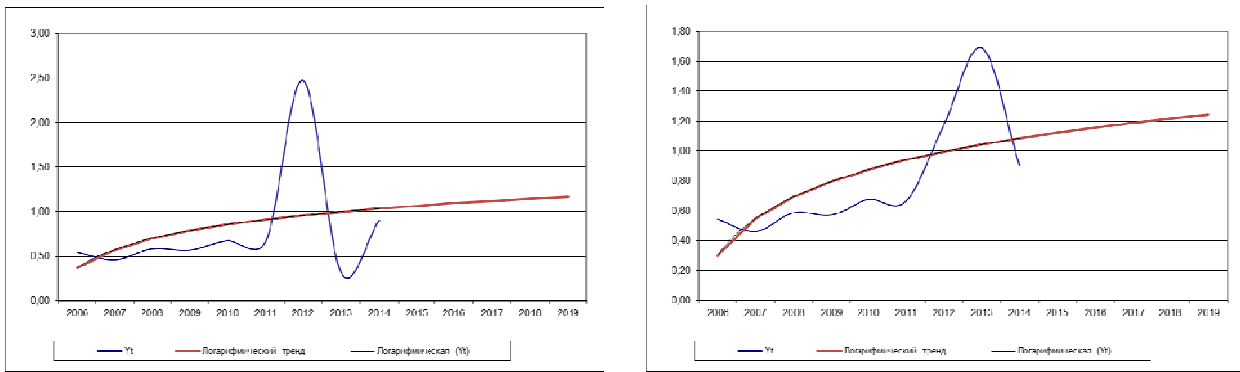


Рис. 3. Логарифмический тренд

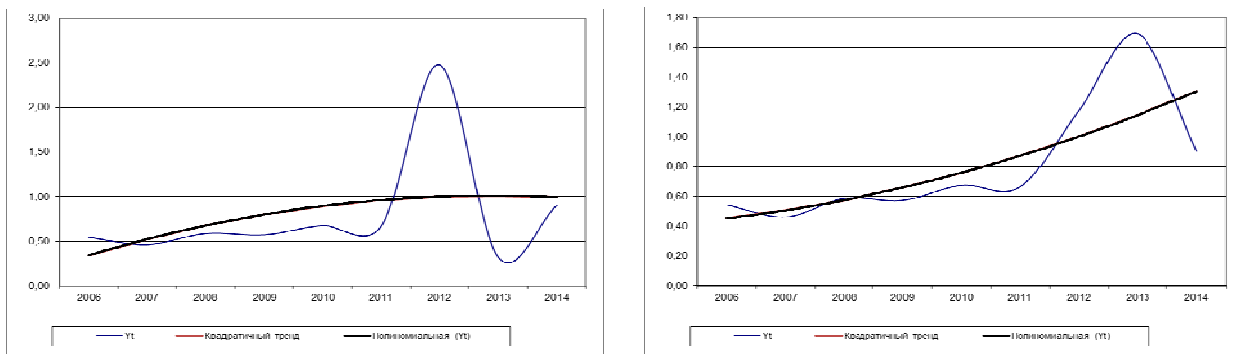


Рис. 4. Квадратичный тренд

Как видно из представленных данных, наименьшие значения критериев отклонений и относительных ошибок, как для трендов с аномальными значениями, так и без них, оказались у квадратичного тренда. Следовательно для

дальнейшего прогнозирования наиболее подходящим является именно квадратичный тренд.

На рис. 5 представлен прогноз по квадратичному тренду с аномальными значениями и без них.

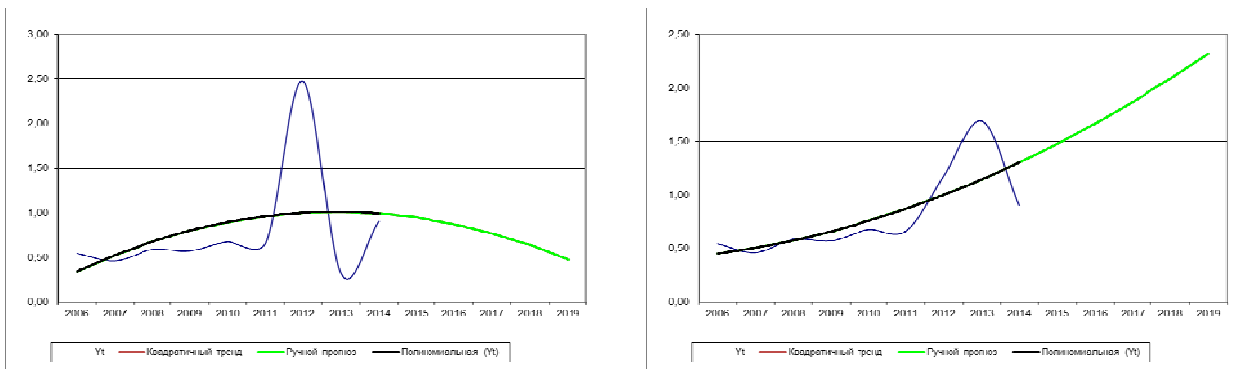


Рис. 5. Квадратичный тренд прогноза

Из графика прогнозирования без аномальных значений можно сделать вывод, что поскольку значение коэффициента продолжает расти, то это значит, что в ближайшие годы процент удовлетворительных оценок по отношению к отличным также будет расти и, как следствие, будет меньше грамотных выпускников.

В качестве рекомендации можно предложить научным руководителям выпускных работ бакалавров большее внимание уделять правильному распределению тем выпускных работ бакалавров, в особенности слабым студентам, с целью их более успешной защиты на хорошие и отличные оценки.

В качестве примера на рис. 6 представлена диаграмма оценок, полученных на защитах выпускниками, руководителя выпускных работ бакалавров и дипломных проектов инженеров, профессора кафедры «Автоматизированные системы обработки информации и управления», Крушель Елены Георгиевны.

Можно проследить тенденцию получения оценок ее выпускниками за период с 1996 по 2015 гг. За эти годы ею было подготовлено к за-

щите 284 выпускника. На гистограмме видно, что за первые годы выпускниками были получены только отличные и хорошие оценки с преобладанием отличных оценок, далее начали появляться удовлетворительные оценки, а за последние годы участились случаи получения удовлетворительных оценок с общим уменьшением отличных. Такая частная картина подтверждает общую тенденцию увеличения количества слабых выпускников.

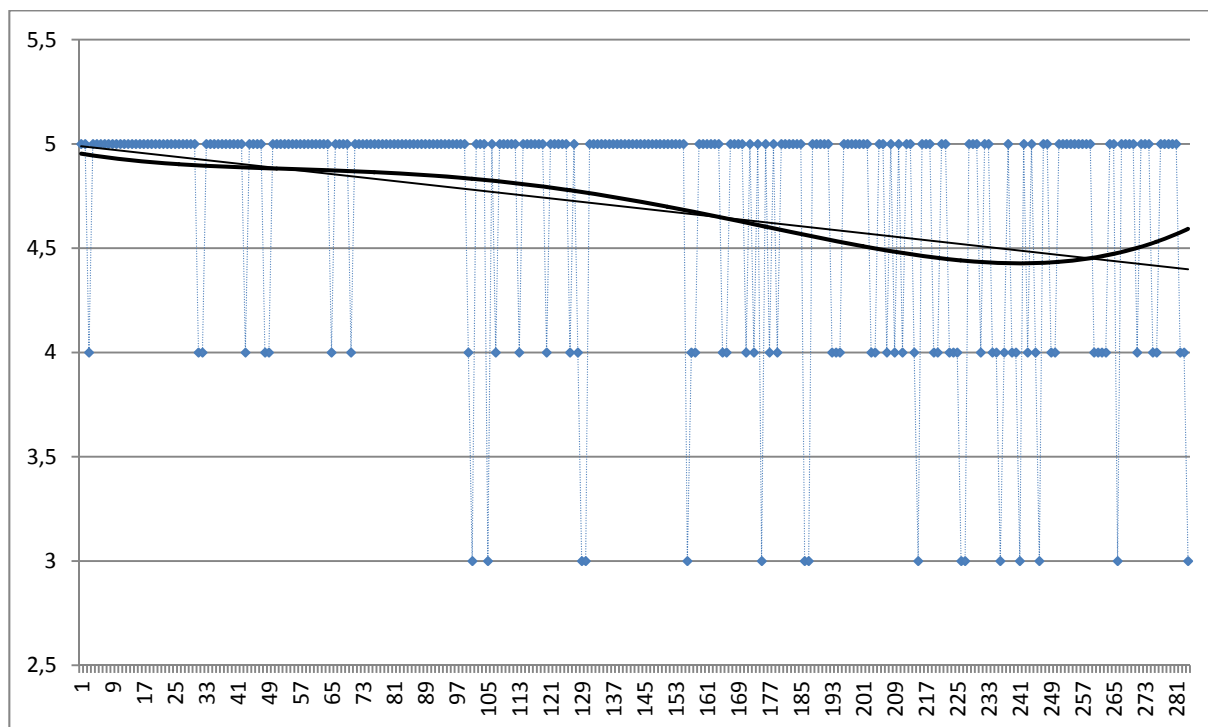


Рис. 6. Диаграмма оценок выпускников

Следует уделить большое внимание не только «подтягиванию» слабых выпускников до уровня хороших, но и, в большей степени, сохранению сильных будущих специалистов в родном городе с целью повышения качества профессиональных сотрудников организаций и предприятий.

Одним из способов сохранения отличных выпускников можно предложить назначение специальных правительственных надбавок отличному молодым специалистам, а также льготы при трудоустройстве на предприятия родного города. Дополнительным стимулом можно было бы предложить введение льгот при получении или покупке жилья для таких специалистов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Харитонов, И. М. Модели и алгоритмы планирования учебного процесса вуза на основе модульно-компе-

тентностного подхода : дис. ... канд. техн. наук / И. М. Харитонов ; Астраханский государственный университет. – Астрахань, 2011. – 202 с.

2. Харитонов, И. М. Моделирование процесса построения учебного плана на основе формализованного представления учебной дисциплины / И. М. Харитонов // Открытое образование. – М., 2011. – № 2. Ч. 1. – С. 21–32.

3. Крушель, Е. Г. Конспект лекций по дисциплине «Обработка экспериментальной информации» [Электронный ресурс]. – Режим доступа : <ftp://ftp.kti.ru/>. – Загл. с экрана. – Яз. рус.

4. Кобзарь, А. И. Прикладная математическая статистика / А. И. Кобзарь. – М.: Физматлит. – 2006. – 816 с.

5. Ходасевич, Г. Б. Обработка экспериментальных данных на ЭВМ / Г. Б. Ходасевич. – СПб.: СПбГУТ. – 2002. – 171 с.

6. Харитонов, И. М. Прогнозирование уровня освоения компетенций выпускниками вуза на основе рейтинговых оценок абитуриентов / И. М. Харитонов, Е. Г. Крушель // Современные проблемы науки и образования. – 2012. – № 6. – С. 43.

УДК 004.942:336.761

*А. В. Петрухин, В. Н. Борозенец, К. А. Шевченко***АВТОМАТИЗАЦИЯ ДИВИДЕНДНОГО МОНИТОРИНГА КОМПАНИЙ-ЭМИТЕНТОВ****Волгоградский государственный технический университет**

htr753@list.ru

В статье рассматривается проблема оперативного получения информации при ведении биржевой деятельности, отдавая предпочтение стратегиям, основанным на фундаментальном анализе. Дается понятие мониторинга, ориентированного на получение различной экономической информации; приводятся примеры такой информации; рассматриваются типы мониторинга по способу автоматизации. Предлагается решение по автоматизации мониторинга экономической информации на примере получения информации о дивидендах компаний-эмитентов. Показаны архитектура системы, макеты пользовательских форм и экранные формы готового прототипа.

Ключевые слова: мониторинг, автоматизированная система, трейдинг, фундаментальный анализ.

*A. V. Petrukhin, V. N. Borozenets, K. A. Shevchenko***AUTOMATION OF DIVIDEND MONITORING OF COMPANIES-ISSUERS****Volgograd State Technical University**

This article discusses the problem of operational information in the conduct of exchange activities, preferring strategies based on fundamental analysis. The article describes the concept of monitoring is focused on obtaining a variety of economic information. The examples of the obtained economic information are given. The types of monitoring by the method of automation are considered. In this article proposes a solution to automate the monitoring of economic information on the example of obtaining information about the dividends of issuing companies. This article shows the system architecture, user form layouts, and screen forms of the finished prototype.

Keywords: monitoring, automated system, trading, fundamental analysis.

Статья посвящена работе с информацией, а конкретно с информацией, которая имеет отношение к торговле на бирже. Важность информации, как значимого ресурса для различных сфер бизнеса в настоящее время не подвергается сомнениям. Не исключением является торговля на бирже. Трейдер, играющий на бирже всегда зависит от информации. Исследования в области автоматизированных способов получения информации имеют особую актуальность. Данное обстоятельство обусловлено, прежде всего, тем, что любая информация может кардинальным образом изменить направление цен на торгуемый актив, и чтобы трейдер не получил убыток, он обязан быть всегда в курсе новостей [2]. Данное исследование актуально в связи с тем, что его результаты позволяют увеличить эффективность работы на бирже путем применения автоматизированной системы для мониторинга фундаментальных показателей компаний-эмитентов.

Во время игры на бирже трейдеру приходится много времени уделять сбору, обработке и анализу различной информации. Процесс мониторинга экономической информации трудоемок, сложен и скоротечен. Именно из-за последнего фактора данный вид деятельности

требует автоматизации всего процесса, так как новости появляются быстро и непредсказуемо и ими воспользоваться можно лишь в короткий срок. Величина затрачиваемого времени зависит от различных показателей, в том числе и от выбора, каким методом пользуется трейдер при принятии решения. Традиционно данный метод может основываться на фундаментальном анализе [4] или на техническом анализе [3]. К техническому анализу мы причислим такие данные как уровни поддержки и сопротивления (исторические минимумы и максимумы), цены закрытия прошлого дня и цены открытия этого дня и прочую информацию, которая отражает историю цены, обычно все эти данные носят чисто числовой характер, которые отлично подходят при создании биржевых роботов [5, 7, 8]. При проведении же фундаментального анализа требуется несколько иные данные, например, финансовый отчет эмитентов и все, что связано с ним [9]. Финансовый отчет эмитента – это документ, в котором отражается финансово-экономическая деятельность компании за определенный срок:

- 1) данные о компании;
- 2) данные о миссиях и стратегиях компании;

3) результаты деятельности (в зависимости от направленности компании, в этом разделе может быть, как доклад о разработке какого-либо продукта, так и о разведке нового месторождения нефти);

4) прочая информация для акционеров и инвесторов.

Также, при проведении фундаментального анализа может использоваться информация, которая влияет на цены активов в неявном виде. В качестве примера такими данными могут быть:

1) календарные данные – дата выборов президента России, дата собраний акционеров, даты международных конференций и симпозиумов, дата обнародования отчета о какой-либо финансово-хозяйственной деятельности;

2) временные данные – время запуска новой линии производства, время открытия выставки или пресс-релиза;

3) новостные данные – по мнению авторов тоже немаловажный показатель, который можно отнести к фундаментальным. Они влияют на финансовую деятельность предприятия. В частности, примерами таких новостей могут служить: выход нового продукта на рынок, введение санкций, чрезвычайное происшествие на фабрике. Каждая из этих новостей может либо обрушить цены на акции компании-эмитента, либо поднять их.

Каждый из этих примеров представляет собой огромный пласт информации, который трейдеру необходимо получить и обработать, чтобы успешно вести свою биржевую деятельность.

Рассмотрим способы мониторинга экономической информации.

Само понятие мониторинг означает процесс непрерывного слежения за появлением новых сведений о деятельности объекта по заданным информационным индикаторам в фиксированном тематическом поле с целью анализа, управления и прогноза его развития.

Существуют различные способы:

1) ручной – все действия выполняет сам трейдер, тратит уйму времени, а учитывая характер некоторых данных, зачастую теряет часть информации или не успевает воспользоваться ими в должной мере;

2) автоматизированный – часть действий выполняется с помощью компьютерной программы, а часть действий выполняет пользователь [1]. Она может производить поиск по заданному критерию, извлекать требуемую ин-

формацию, сохранять ее в необходимом формате и все это делать по установленному таймеру с выставленным повторением [6]. Для такой программы требуется составить некий формуляр, примером может послужить модель, предложенная в этой статье [10], в котором будет указано: на каких сайтах вести мониторинг, классификация данных, соотношение искомых данных и форматы, в которых они будут сохраняться, в каком формате сохранять;

3) автоматический – все действия выполняет программа. От пользователя требуется только нажать кнопку «Пуск» и произвести начальную настройку. Все дальнейшие действия по определению сайтов, соотношение типов данных с форматами и так далее, программа будет выполнять сама.

Рассмотрим один из вариантов реализации в рамках предложенной концепции, а именно – использование автоматизированного способа ведения дивидендного мониторинга различных компаний-эмитентов.

Дивиденды – это часть прибыли акционерного общества или иного хозяйствующего субъекта, распределяемая между акционерами, участниками в соответствии с количеством и видом акций, долей, находящихся в их владении. Компания-эмитент – это организация, которая выпускает (эмитирует) ценные бумаги для развития и финансирования своей деятельности.

Суть этого примера заключается в том, что трейдер, торгующий по определенной стратегии, ориентированной на получение дивидендов (это может быть, как обыкновенная покупка с целью получения дивидендов, так и необходимость избавления от имеющихся акций, чтобы продать их по максимальной цене), должен быть постоянно в курсе любых изменений, таких как дата собраний акционеров или количество, выплачиваемых дивидендов. Стоит учесть еще один немаловажный факт, что данная информация может быть точной и неточной. Официальная информация появляется на сайтах самих эмитентов, либо на сайтах ведущих брокеров. Неточная или неофициальная информация появляется на сайтах некрупных брокеров или на мелких информационно-экономических сайтах. Имея такую программу в своем инструментарии, трейдеру будет легче ориентироваться в том или ином вопросе, связанном с принятием торгового решения.

Рассмотрим последовательность действий пользователя.

Пользователь выбирает, на каких сайтах следует вести мониторинг. По желанию пользователя, он сможет самостоятельно добавлять, интересующие его сайты. Макеты пользовательских форм представлены на рис. 1, 2.

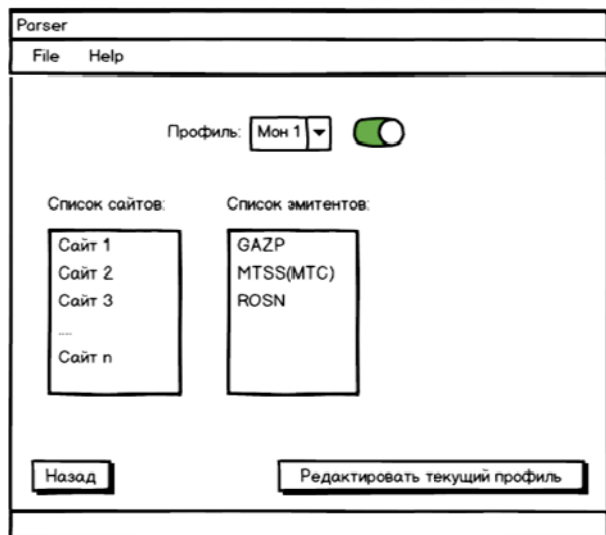


Рис. 1. Макет пользовательской формы выбора профиля

Вторым действием пользователь выбирает, какие эмитенты его интересуют. По умолчанию будет внесена десятка самых крупных эмитентов нашей страны, также, по желанию пользователя, он сам сможет добавлять необходимых ему эмитентов.

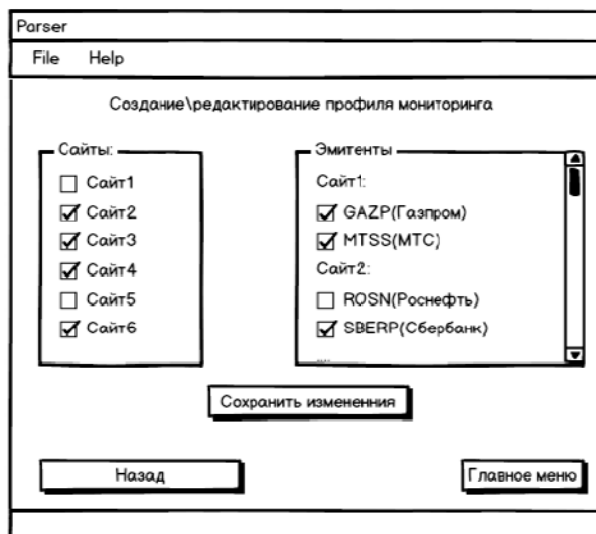


Рис. 2. Макет пользовательской формы редактирования профиля

После выбора сайтов и эмитентов программа проверит запрос на ошибки и, в случае отсутствия последних, приступит к выполнению запроса.

На форме появится запрашиваемая информация, разделенная по сайтам в виде таблицы (данные приведены в качестве примера и не отражают настоящего положения).

Как видно из таблицы, данные будут содержать последнюю точную информацию.

Таблица 1

Вид таблицы формы вывода

Название эмитента	Тикер	Дата собраний	Дата закрытия реестра	Дивиденды
Газпром ао	GAZP	01.01.2018	01.01.2018	1000
Сбербанк	SBER	01.01.2018	01.01.2018	100

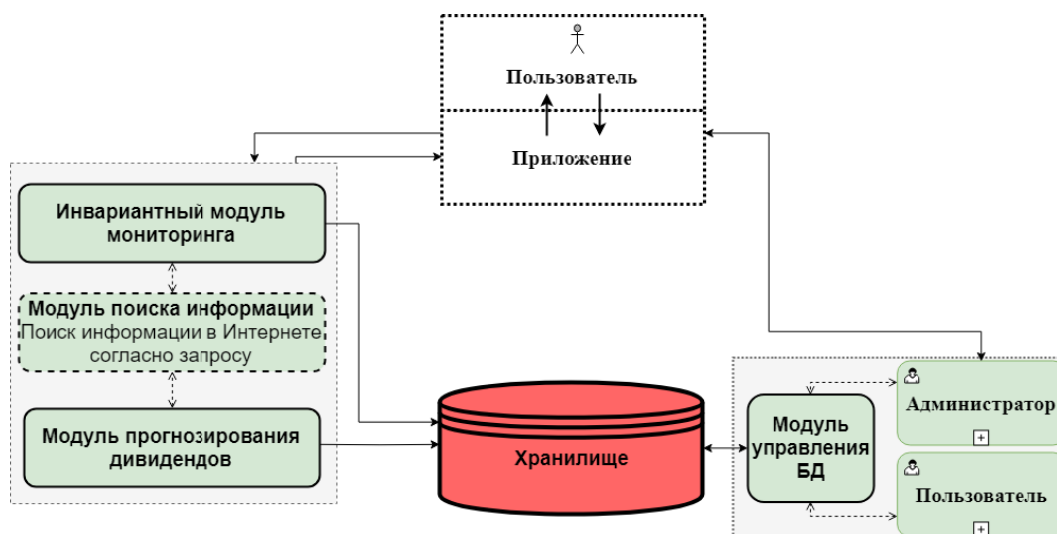


Рис. 3. Архитектура системы

В итоговой форме пользователю будет предоставлен выбор: сохранить полученную информацию, обновить полученную информацию, выставить интервал для автоматическо-

го обновления и сохранения информации.

Архитектура предлагаемого решения представлена на рис. 3.

Рабочий прототип представлен на рис. 4, 5, 6.

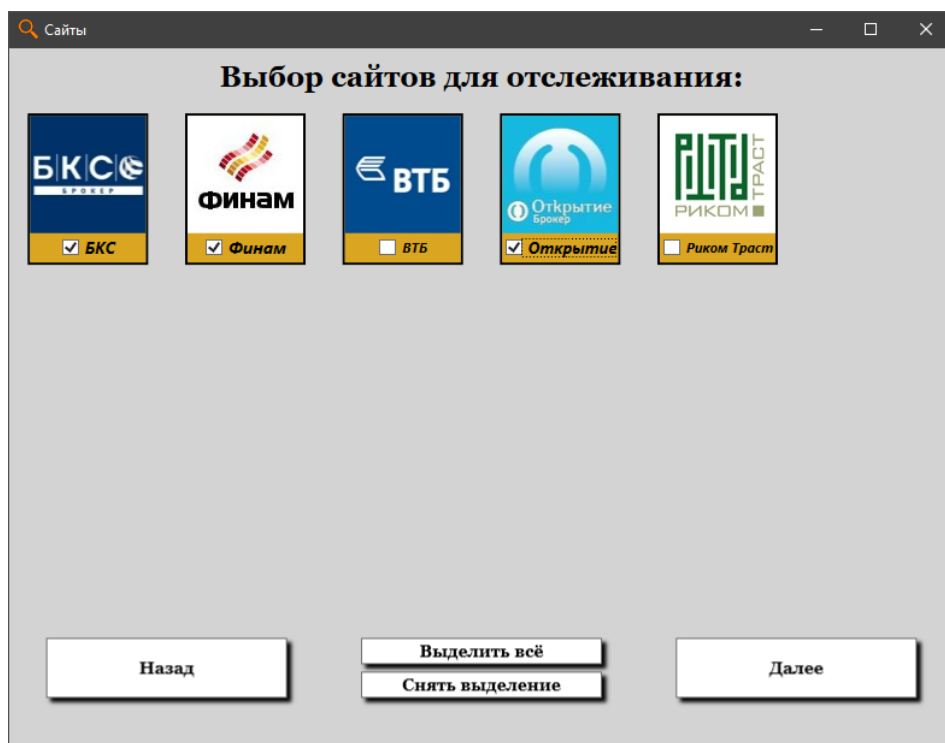


Рис. 4. Выбор сайтов

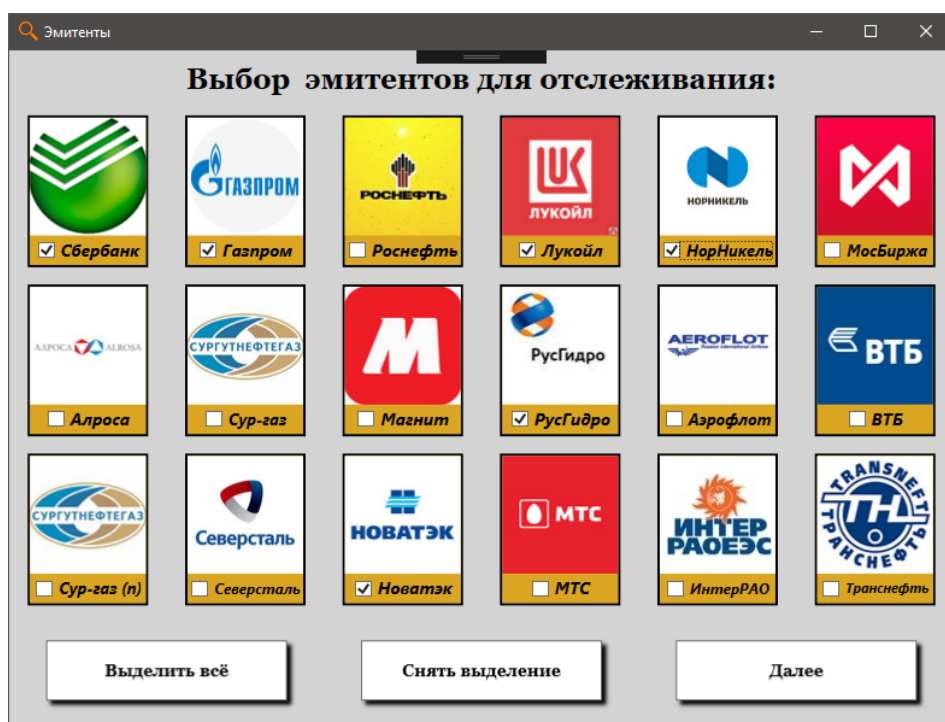


Рис. 5. Выбор эмитентов

Результат

БКС

Название	Посл-ий день	Дата з.реестра	Цена дивид.	Размер дивид.
X5 Retail	—	—	30**	1943,5
Алроса оо	—	—	5,3**	88,72
АФК «Система» оо 2017	—	—	0,51***	12,18
Аэрофлот оо	—	—	10,38**	149,4
Башнефть оо	—	—	370**	2252
Башнефть ал	—	—	370**	1618,5

Сохранить

Обновить

Автосохранение

Интервал сохранения в секундах: 100

☒ Все доступные эмитенты на сайте

ФИНАМ

Открытие

Название	Тикер	Цена акции	Дата з.реестра	Дивиденды
ПАО Россети	RSTIP	1.539	— —	0
ПАО Ростелеком	RTKM	68.1	07.07.2018	4.5
ПАО Ростелеком	RTKMP	60.99	07.07.2018	4.5
ПАО РусГидро	HYDR	0.79	10.07.2018	0.05592
ПАО Сбербанк	SBER	277	14.06.2018	12
ПАО Сбербанк	SBERP	229.29	14.06.2018	12

Сохранить

Обновить

Автосохранение

Интервал сохранения в секундах: 100

☒ Все доступные эмитенты на сайте

Риком Трест

ВТБ

Название	Дата собрания	Дата з.реестра	обыкн.	привил.
Акрон	18.12.2017	23.01.2018	112	
АФК Система	03.11.2017	08.12.2017	0.68	
Башнефть	06.11.2017	12.12.2017	148.31	148.31

Сохранить

Обновить

Автосохранение

Назад

Выход

Рис. 6. Окно формы результата

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Криницкий, Н.А. Автоматизированные информационные системы / Н.А. Криницкий, Г.А. Миронов, Г.Д. Фролов. – М.: Наука, 1982. – 381 с.
2. Федоров, А.В. Анализ финансовых рынков и торговли финансовыми активами. – М.: Питер, 2007. – 240 с.
3. Мэрфи, Д.Д. Технический анализ фьючерсных рынков: теория и практика : пер. с англ. – М.: Альпина Паблишер, 2004. – 479 с.
4. Колмыкова, Л.И. Фундаментальный анализ финансовых рынков. – М.: Питер, 2007. – 288 с.
5. Петрухин, А.В. Архитектура биржевого робота для работы в среде торгового терминала QUIK / А.В. Петрухин // Известия Волгоградского государственного технического университета. – 2015. – № 2 (157). – С. 138–141.
6. Представление физических знаний для автоматизированных систем обработки информации : моногр. / С.А. Фоменков, А.В. Петрухин, В.А. Камаев, Д.А. Давыдов ; ВолгГТУ. – Волгоград, 1998. – 128 с.
7. Петрухин, А.В. Компьютерная визуализация биржевых данных о динамике фондового рынка / А.В. Петрухин, А.С. Стешенко // Известия Волгоградского государственного технического университета. 2015. № 6 (163). С. 124–129.
8. Петрухин, А.В. Мультидисциплинарный подход к реализации инструментальных средств информационного поиска и поддержки принятия решений / А.В. Петрухин, А.С. Колесников, А.А. Фролов // Известия Волгоградского государственного технического университета. 2015. № 6 (163). С. 186–190.
9. Даглас, М. Дисциплинированный трейдер. Бизнес-психология успеха. – М.: Евро, 2004. – 288 с.
10. Петрухин, А.В. Модель инвариантного описания биржевых торговых стратегий / А.В. Петрухин, К.А. Шевченко // Известия Волгоградского государственного технического университета. 2018. № 1 (211). С. 101–106.

УДК 004.81

*В. Л. Розалиев, А. В. Заболеева-Зотова, Ю. А. Орлова,
О. С. Гусынин, А. В. Ражева, Д. О. Недоступов*

**РАЗРАБОТКА МЕТОДОВ ОПРЕДЕЛЕНИЯ ЭМОЦИОНАЛЬНОСТИ ЧЕЛОВЕКА
ПО МИМИКЕ, ТЕКСТУ, ДВИЖЕНИЯМ РУК И ГЛАЗ***

Волгоградский государственный технический университет
vladimir.rozaliyev@gmail.com

В статье описаны подходы, позволяющие увеличить точность распознавания эмоционального состояния человека. Рассмотрены разработанные методы, основанные на анализе видеоизображения и жестов рук, тональности текста, мимики и движения глаз. Проведены эксперименты, показывающие высокие результаты при комбинированном подходе в определении эмоционального состояния человека.

Ключевые слова: эмоциональное состояние, распознавание, веб-камера, реальное время, Leap Motion, Tobii Eye Tracker, мимика, жесты, тональность текста.

*V. L. Rozaliyev, A. V. Zaboileeva-Zotova, U. A. Orlova,
O. S. Gusynin, A. V. Razheva, D. O. Nedostupov*

**DEVELOPING RECOGNITION METHODS OF HUMAN EMOTIONAL
BY FACIAL EXPRESSIONS, TEXT, HAND MOVEMENTS AND EYES**

Volgograd State Technical University

The article describes approaches that allow increasing the accuracy of recognition of a person's emotional state. The developed methods based on: analysis of video images and gestures of hands, tonality of the text, facial expressions and eye movements are considered. Experiments have been conducted that show high results in a combined approach in determining the emotional state of a person.

Keywords: emotional state, recognition, webcam, real time, LeapMotion, Tobii Eye Tracker, facial expressions, gestures, tonality of the text.

Введение

Развитие информационных технологий в современном мире идет невообразимыми темпами, что постепенно приводит к полному погружению человека в технологии. С каждым днем информационные системы становятся все более дружелюбными к человеку, они все проще в использовании, но до сих пор не способны в полной мере распознавать эмоции человека. Выражение эмоций – одна из основных когнитивных функций человека. Технология распознавания эмоций поможет системе, на основе информации о своем пользователе, подстраиваться под него, делая взаимодействие человек-компьютер более естественным и похожим на взаимодействие человек-человек. Достижение достаточной точности в распознавании эмоций в режиме реального времени поможет, если не стереть границы во взаимодействии человек-компьютер, то сделать их максимально размытыми.

**Использование жестов
в определении эмоциональности человека**

Подход, основанный на использовании жестов для увеличения точности прогнозирования эмоций, базируется на использовании сенсора

LeapMotion [1]. Сенсор используется для считывания данных о динамике движения кистей рук человека в пространстве, а также определения жестов. Данный подход, дополняет собой другой метод определения эмоциональности, основанный на анализе мимики лица человека.

Метод распознавания эмоционального состояния по мимике и жестам состоит из двух этапов: сначала происходит считывание карты лица человека и определение его вероятных эмоций, затем эта информация сопоставляется с данными полученными с помощью Leap Motion. При проявлении эмоций человек рефлекторно совершает некоторые действия руками. Это может быть как изменение в динамике его движения, так и формирование определенных жестов [2–3]. На рис. 1 продемонстрированы примеры жестов человека, при проявлении эмоций.

Помимо жестов, для уточнения эмоций, возможно использовать информацию о динамике движения рук. Например, при проявлении страха, у человека могут начать трястись руки, либо резкое изменение положения в пространстве, при испуге. На рис. 2 продемонстрировано резкое изменение положения руки человека в пространстве при испуге.

© Розалиев В. Л., Заболеева-Зотова А. В., Орлова Ю. А., Гусынин О. С., Ражева А. В., Недоступов Д. О., 2018

* Работа выполнена при частичной поддержке РФФИ и администрации Волгоградской области (гранты 16-07-00407, 16-07-00453, 16-47-3403320, 18-07-00220).

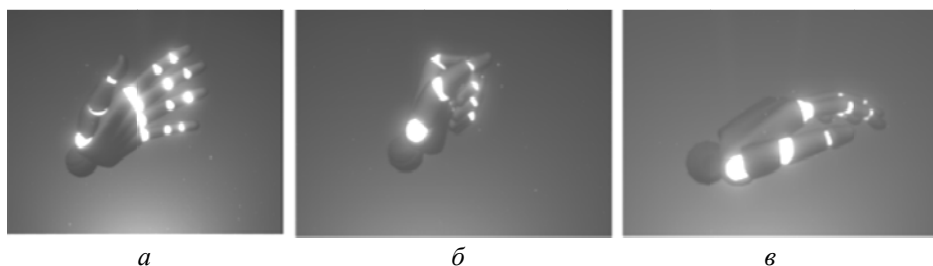


Рис. 1. Жесты при проявлении эмоций:
а – удивление; б – гнев; в – нейтральность

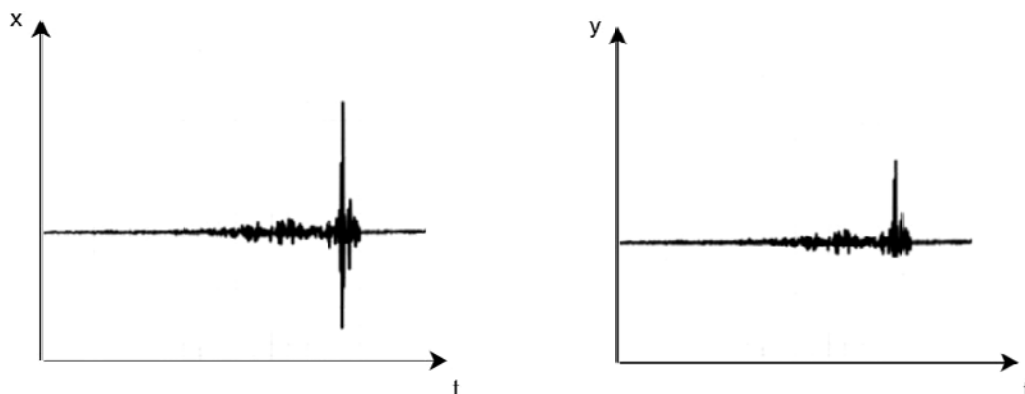


Рис. 2. Резкое изменение положение рук в пространстве при испуге

Данный метод позволяет, в целом, улучшить точность распознавания эмоций, в частности, те эмоции, которые являются наименее выразительными и, следовательно, более труднораспознаваемыми, такие как страх, отвращение и злость.

Для проверки утверждения об увеличении точности распознавания эмоций, при использовании сенсора Leap Motion, было реализовано два приложения. Первое приложение распознает эмоции человека по видеоизображению, которое захватывается с веб-камеры компьютера. Для его реализации была использована сверточная нейронная сеть [4], так как именно она наиболее подходит для задач обработки изображений. Для проверки теории, данная нейронная сеть была обучена на достаточно небольшом количестве входных данных, порядка 3–4 тысяч изображений для каждой эмоции [5]. Точность распознавания эмоций сильно варьируется в зависимости от эмоций. Выразительные эмоции, такие как удивление и радость имеют достаточно высокую степень распознавания, порядка 90 %, несмотря на небольшую обучающую выборку. Не выразительные эмоции, такие как печаль и отвращение, имели достаточно низкую степень распознавания, порядка 35 %.

Для уточнения прогнозов, относительно проявляемых эмоций, был применен LeapMotion. Было проведено несколько экспериментов, в рамках которых, проявляемые эмоции подкреплялись различными движениями рук. На рис. 3 изображены результаты первого эксперимента, в котором была продемонстрирована эмоция страха (Fear). Как видно из рисунка, приложение, определяющее эмоции по видеоизображению, показывает проявление эмоции удивления (Surprise), с точностью примерно 60 %. Однако, проявление данной эмоции сопровождалось дрожанием рук (Tremor), которое захватывалось LeapMotion. В результате, после уточнения данных, окончательный прогноз был обновлен.

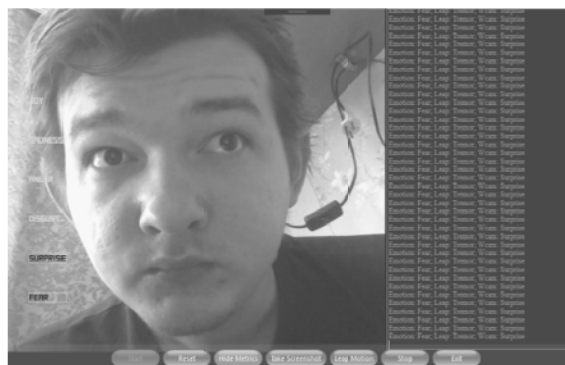


Рис. 3. Эмоция страха. Широко открытые глаза (удивление) сопровождаются дрожанием рук

На рис. 4 показана та же ситуация, но руки находились в спокойном состоянии.

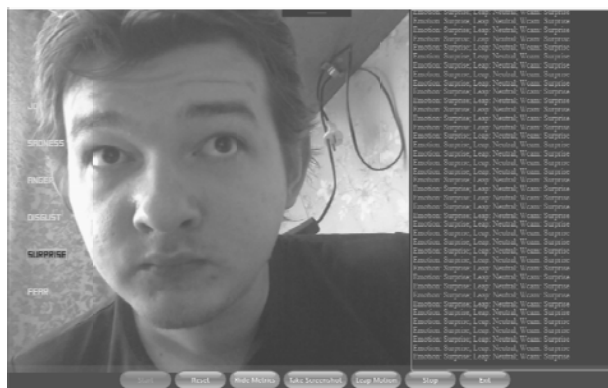


Рис. 4. Широко раскрытые глаза сопровождаются спокойным состоянием рук, результат – эмоция удивление

По результатам эксперимента можно сделать вывод, что подход, включающий уточнение информации, относительно проявляемых эмоций, при помощи динамики движения рук показывает неплохие результаты. Реализация метода анализа движения рук при проявлении человеком различных эмоций, позволяет добиться высоких показателей точности прогнозирования эмоций, и применяется при разработке комплексного подхода к определению эмоциональных состояний по речи, движениям, мимике [7].

Анализ тональности текста

В данный момент существует несколько аналогов, для определения тональности текстовой информации. За последнее время чувствуется прирост в потребности инструментов, позволяющих провести обработку отзывов пользователей, основанной на тональности текста. В основном, все они опираются на методологию, основывающуюся на тональных словарях, с применением дополнительных методов анализа и классификации. Эти словари содержат в себе различные фразы и слова, а также их тональную оценку. Каждый язык и каждая область, предназначенная для тонального анализа, должны иметь свой собственный эмоциональный словарь, позволяющий учесть индивидуальные особенности, для более точной оценки текста.

Анализ тональности текста является одной из задач компьютерной лингвистики. Существует несколько методов тональной классификации текста. Условно все эти методы можно разделить на следующие категории:

– методы, основанные на правилах. Состоит из набора правил, с помощью которых производится анализ на наличие тональности в тексте;

– методы, основанные на словарях. Использует тональные словари, для текстового анализа;

– машинное обучение с учителем. Основная идея заключается в обучении классификатора на заранее известной коллекции текстов. Затем эту модель можно будет применять для анализа и тональной классификации новых текстов;

– машинное обучение без учителя. Метод опирается на правила, по которым выявляет эмоционально окрашенные слова. На основе их тональности, определяется общая тональность всего текста;

– метод, основанный на теоретико-графовых моделях основывается на предположении, что не все слова вносят одинаковый вклад в общую тональность текста. На основе наиболее сильного и слабого влияния строится граф, по исследуемому тексту.

Для разработки программного продукта были выделены следующие этапы:

– Импорт документа в текстовом формате с удалением ненужной информации (изображений, разметки, гиперссылок и т. д.).

– Создание текстового классификатора. Этот процесс разбивается на следующие шаги:

- выявление субъекта и объекта текста;
- выделение базы однокоренных слов для приведения их к одинаковому виду («стемминг»);
- выделение наиболее значимых слов, которые сильнее всего влияют на рейтинг отзыва при помощи метода тональных словарей;
- обучение нейронной сети;
- тестирование полученной программы.

В качестве базы текстов использовались наборы отзывов с различных сайтов. Для анализа веб-страниц сайтов используется библиотека HtmlAgilityPack. Данная библиотека написана на языке программирования C# и позволяет анализировать и генерировать HTML-файлы. Также существует возможность анализа синтаксически неверных HTML-файлов. Для приведения слов к нормальной форме была применена библиотека морфологического анализа.

Программное обеспечение, разработанное на языке программирования C#, позволяет выбрать комментарии пользователей с различных сайтов. Приложение также выделяет объект и субъект комментариев. Присутствует возможность сохранения и загрузки проанализированной информации в файл. Сохранение данных

необходимо для хранения проанализированных комментариев и дальнейшей работы с ними. На

рис. 5 приведен пример проанализированных программным обеспечением данных.

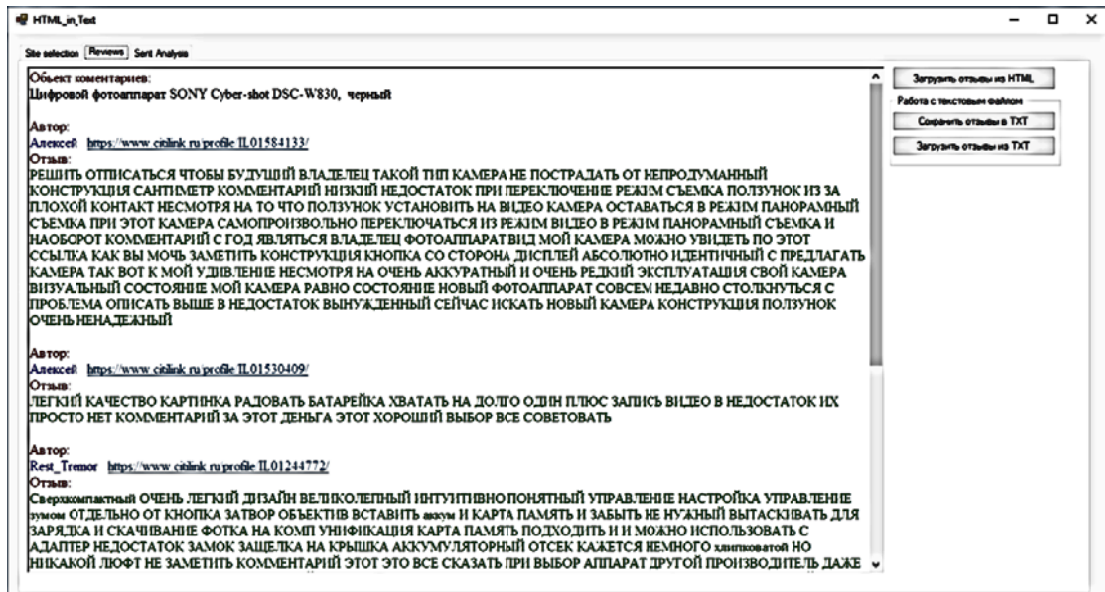


Рис. 5. Результат работы программы после выявления комментариев о камерах, с применением лемматизации

Программное обеспечение позволяет дать тональную оценку выявленных комментариев. Для этого подаются обработанные ранее отзывы пользователей, автор отзыва, объект, о котором выражается эмоция, и одна из пяти тематик, а именно: цифровые камеры, фильмы, рестораны, книги, автомобили.

Программа реализована методом тональных словарей. Каждый из словарей, на которых строится программа, содержит в себе 10000 слов. Словам приводится оценка от 0 до 5 в следующих форматах: позитивный, негативный, ней-

тральный, неизвестный. Все эти лексиконы были взяты из работы [6] и бесплатны для некоммерческого использования.

В данный момент каждое слово в комментарии сверяется со словами в лексиконе. Если слово присутствует в тональном словаре, ему присваивается величина одной из квалификаций. На основе всех выделенных слов дается оценка комментарию. На рис. 6 представлен пример проанализированного отзыва на предмет тональной составляющей.

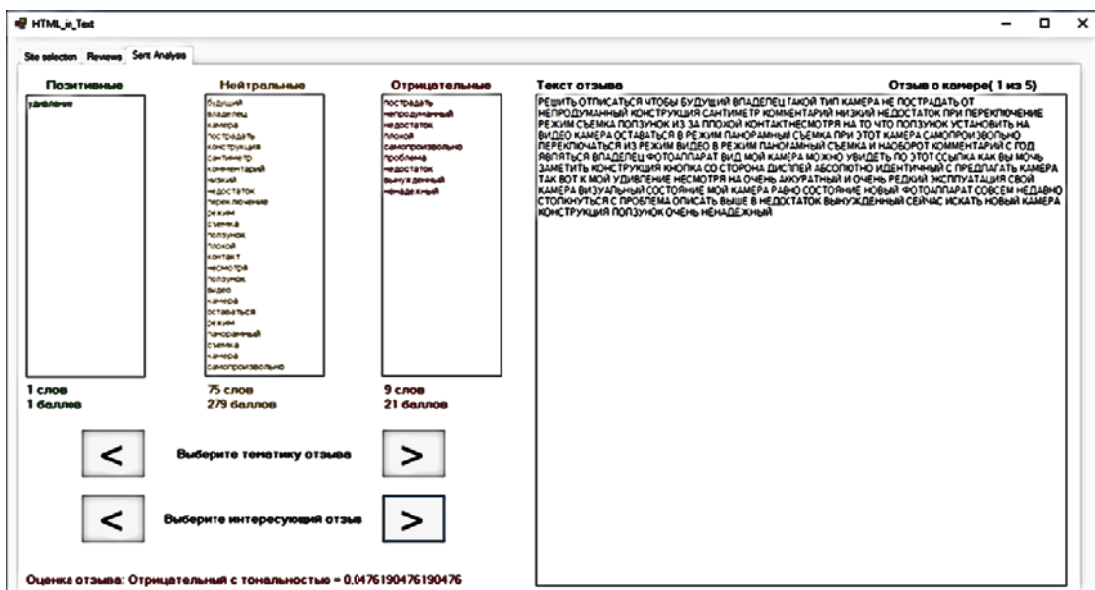


Рис. 6. Результат работы программы после применения тонального анализа

В данной работе было проведено несколько экспериментов. Их суть заключалась в выявлении комментариев, и присвоение им тональной оценки (положительной, отрицательной или нейтральной) при помощи программного обес-

печения. Затем проверялась точность распознавания эмоций. Также была дана количественная оценка успешно приведенных к нормальной форме слов. В табл. 1 приведен пример результатов работы программы.

Таблица 1

Результат анализа комментариев о фильме
«Человек-паук: Возвращение домой»

Наименование объекта	Человек-паук: Возвращение домой
Количество положительных комментариев	32
Количество нейтральных комментариев	25
Количество отрицательных комментариев	21
Количество верно определенной тональности комментариев при помощи программы	67
Процентное соотношение верно определенной тональности к общему количеству комментариев	86 %

Из результатов экспериментов было выявлено, что комментарии пользователей находились без потери информации, а также с выявлением объекта и субъекта. Приведение к нормальной форме слова проходило удачно в 94 % случаев. Метод анализа комментариев по тематике различным тематикам определил тональность верно по каждой тематике не менее чем в 84 % случаев.

В дальнейшем планируется добавить искусственную нейронную сеть для анализа комментариев и использовать ранее разработанные подходы для автоматического выявления схожих текстов [8]. Для обучения нейросети будет подаваться полученная ранее с помощью лексикона тональная оценка и соответствующий комментарий пользователя. На выходе планируется получить финальную тональную оценку комментария.

Для оценки работы нейронной сети необходимо привлечение экспертов. Их главной задачей будет определение точности проведенного анализа комментариев.

Восприятие эмоциональных экспрессий лица человека

Проблема восприятия состояния человека по мимике лица активно разрабатывается современной наукой. В психологии получен боль-

шой эмпирический материал, раскрывающий природу мимических экспрессий, условия и способы их распознавания, роль выражений лица в регуляции взаимодействия людей; предложен ряд концепций, описывающих различные аспекты межличностного восприятия.

Для оценки особенностей восприятия различных экспрессий лица был разработан аппаратно-программный комплекс, представляющий собой приложение для просмотра, фиксации и визуализации перемещения взгляда пользователя на языке C# с использованием камеры Tobii Eye Tracker 4C.

Были проведены эксперименты, целью которых было выявление закономерностей в восприятии человеком базовых эмоций с помощью анализа перемещения взгляда в зонах интереса при просмотре видео разного эмоционального окраса.

Для проведения эксперимента был составлен набор из 6 тестовых стимулов, каждый из которых представляет собой видео файл, содержащий последовательность кадров изменения мимики человека от нейтрального положения до целевой эмоции. Естественный фон обрезан по контуру головы. Длина каждого видео составляет 1 с – 30 кадров. Графическое представление видеофайла отображено на рис. 7.

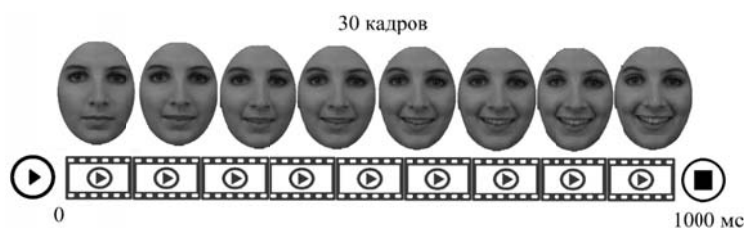


Рис. 7. Последовательность кадров видео

Были выделены четыре зоны интереса для исследования восприятия эмоций: левый глаз, правый глаз, нос, рот. Исследуемые зоны интереса представлены на рис. 8.

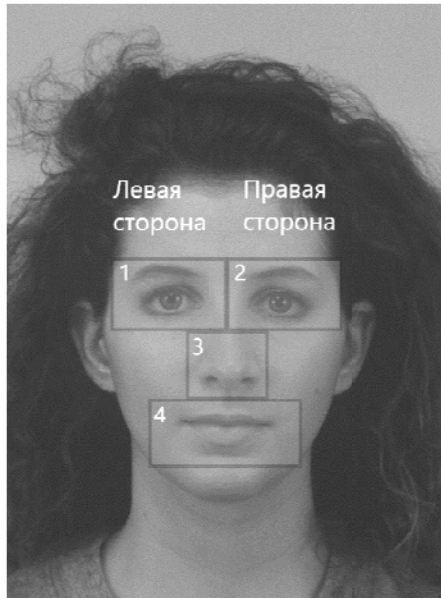


Рис. 8. Зоны интереса:
1 – левый глаз; 2 – правый глаз; 3 – нос; 4 – рот

Исследуемой величиной является общая продолжительность в зоне интереса, измеряемая в миллисекундах.

Также исследованию подлежал анализ правильности распознавания продемонстрированной эмоции посредством выбора соответствующего тега. В интерфейсе программы пользователю предлагалось выбрать тег, описывающий какая эмоция была продемонстрирована.

Для визуализации перемещения взгляда был использован метод графический представления данных тепловая карта.

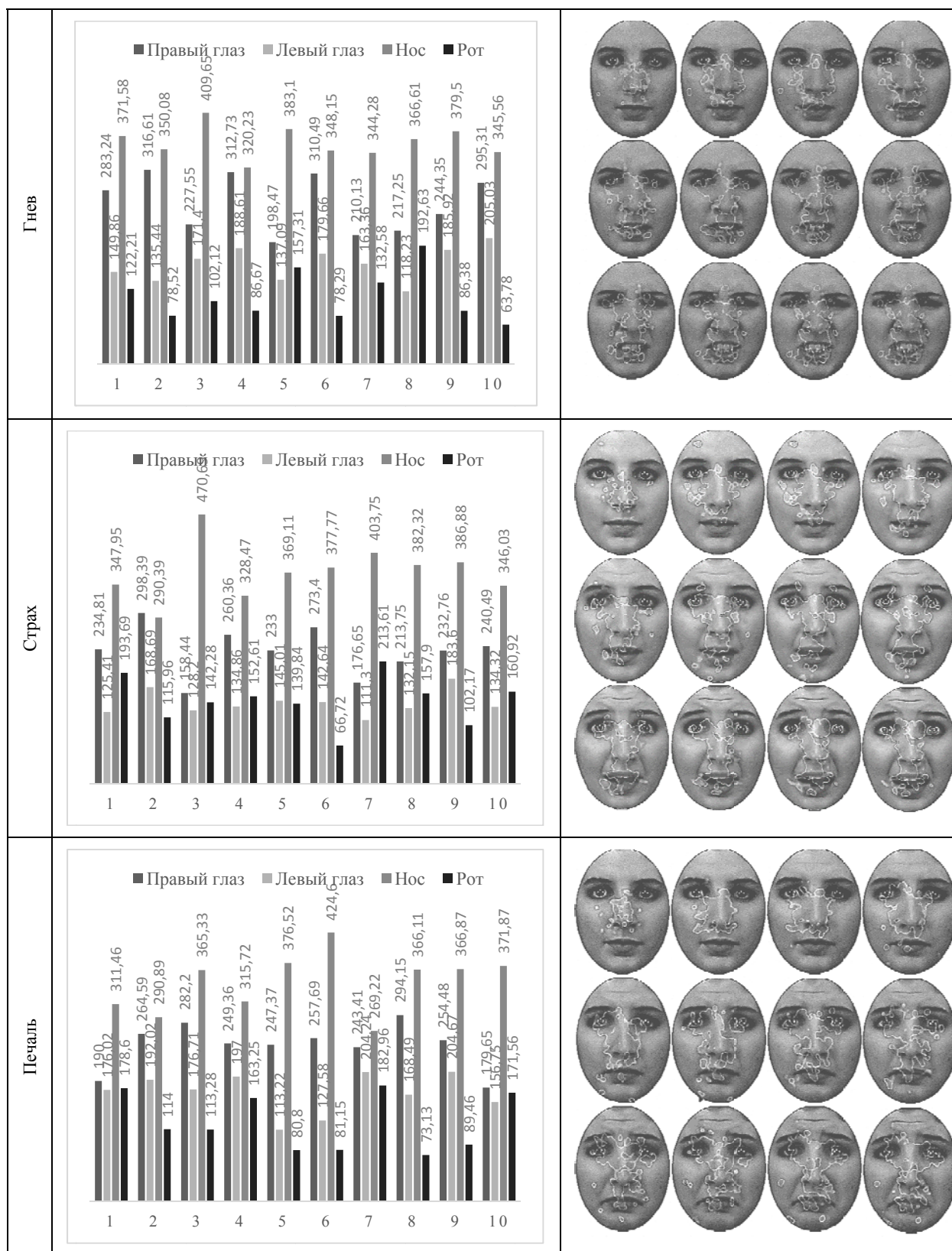
В тестировании принимало участие 10 человек, 5 мужчин и 5 женщин в возрасте от 20 до 25 лет. Согласно инструкции, испытуемым было необходимо зафиксировать взгляд на представляемом ему стимуле, во время чего, посредством камеры Tobii eye tracker 4с фиксировались перемещения взгляда и записывались в файл для анализа. По завершении каждой пробы необходимо было выбрать тег в окне программы, соответствующий продемонстрированной эмоциональной экспрессии.

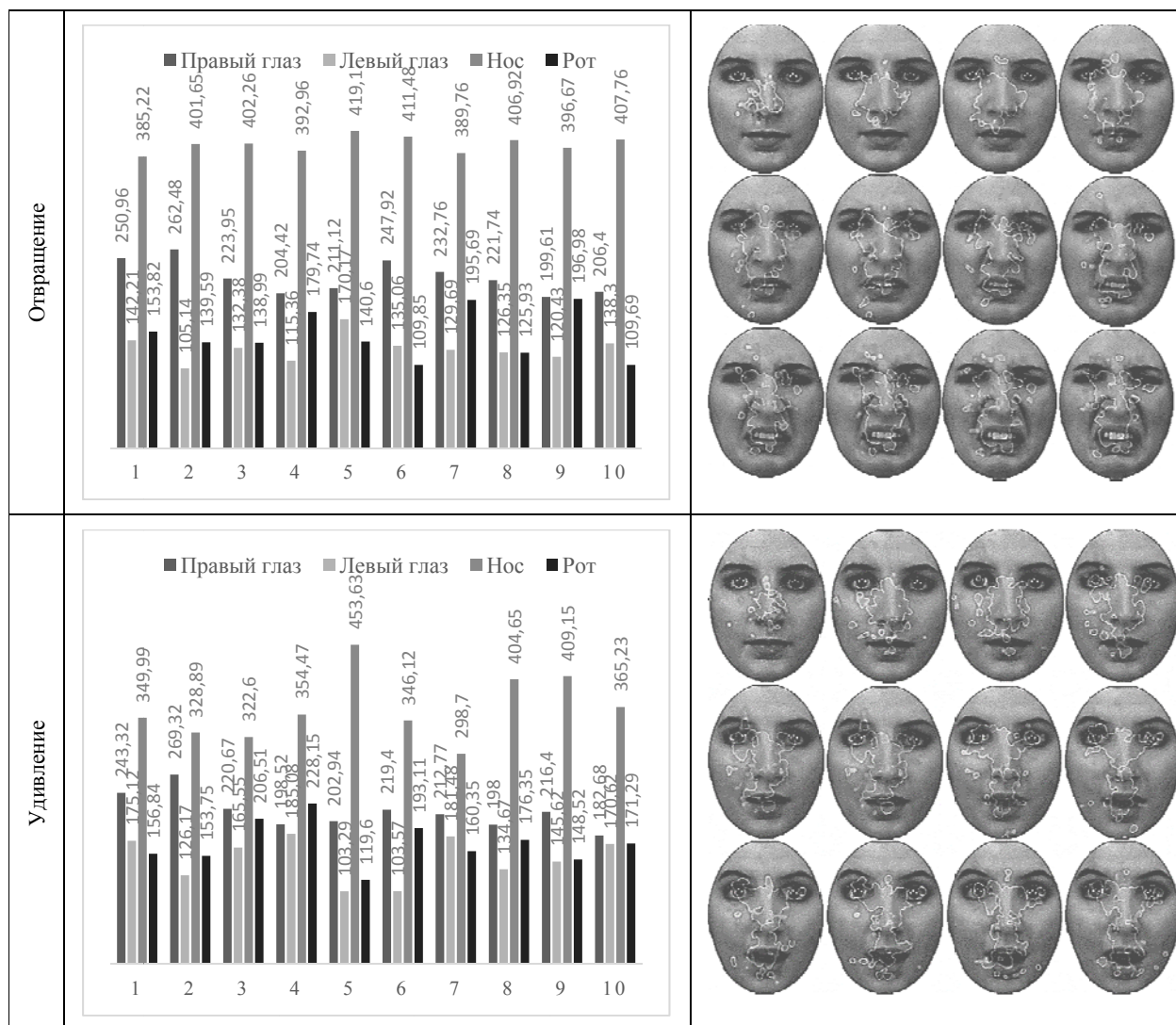
В результате тестирования были получены данные, приведенные в табл. 2.

Таблица 2

Общая продолжительность фиксаций в каждой лицевой области каждой из шести эмоциональных экспрессий

	Общая продолжительность фиксаций в зоне интереса	Тепловая карта																																																							
Радость	<table><thead><tr><th>Триал</th><th>Правый глаз</th><th>Левый глаз</th><th>Нос</th><th>Рот</th></tr></thead><tbody><tr><td>1</td><td>211,05</td><td>106,62</td><td>385,02</td><td>210,79</td></tr><tr><td>2</td><td>156,95</td><td>105,63</td><td>310,14</td><td>283,41</td></tr><tr><td>3</td><td>145,55</td><td>141,61</td><td>193,63</td><td>418,3</td></tr><tr><td>4</td><td>182,26</td><td>97</td><td>230,16</td><td>387,91</td></tr><tr><td>5</td><td>113,07</td><td>159,69</td><td>235,81</td><td>402,5</td></tr><tr><td>6</td><td>182,73</td><td>94,18</td><td>226,29</td><td>369,44</td></tr><tr><td>7</td><td>177,53</td><td>133,78</td><td>221,11</td><td>356,81</td></tr><tr><td>8</td><td>169,56</td><td>92,53</td><td>239,1</td><td>337,37</td></tr><tr><td>9</td><td>196,02</td><td>148,09</td><td>202,16</td><td>351,48</td></tr><tr><td>10</td><td>181,28</td><td>97,34</td><td>214,82</td><td>423,56</td></tr></tbody></table>	Триал	Правый глаз	Левый глаз	Нос	Рот	1	211,05	106,62	385,02	210,79	2	156,95	105,63	310,14	283,41	3	145,55	141,61	193,63	418,3	4	182,26	97	230,16	387,91	5	113,07	159,69	235,81	402,5	6	182,73	94,18	226,29	369,44	7	177,53	133,78	221,11	356,81	8	169,56	92,53	239,1	337,37	9	196,02	148,09	202,16	351,48	10	181,28	97,34	214,82	423,56	
	Триал	Правый глаз	Левый глаз	Нос	Рот																																																				
	1	211,05	106,62	385,02	210,79																																																				
	2	156,95	105,63	310,14	283,41																																																				
	3	145,55	141,61	193,63	418,3																																																				
	4	182,26	97	230,16	387,91																																																				
	5	113,07	159,69	235,81	402,5																																																				
	6	182,73	94,18	226,29	369,44																																																				
	7	177,53	133,78	221,11	356,81																																																				
	8	169,56	92,53	239,1	337,37																																																				
	9	196,02	148,09	202,16	351,48																																																				
10	181,28	97,34	214,82	423,56																																																					





По завершении каждой пробы испытуемому необходимо было выбрать тег в окне программы, соответствующий продемонстрированной

эмоциональной экспрессии. Результаты анализа правильности распознавания испытуемым базовой эмоции приведены в табл. 3.

Соотношение числа верно определенных эмоций к правильно определенным

Таблица 3

Выбираемая эмоция	Демонстрируемая эмоция					
	Радость	Гнев	Страх	Печаль	Отвращение	Удивление
Радость	0,81	0,02	0,03	0,02	0,04	0,03
Гнев	0,02	0,54	0,07	0,08	0,18	0,08
Страх	0,03	0,07	0,71	0,06	0,08	0,12
Печаль	0,02	0,08	0,06	0,56	0,05	0,09
Отвращение	0,04	0,18	0,01	0,05	0,62	0,03
Удивление	0,03	0,08	0,12	0,09	0,03	0,61

По результатам проведенного исследования можно сделать вывод, что при экспозиции изо-

бражений экспрессий лица количество фиксаций происходит в левой половине в большей

степени. На выделенные зоны интереса затрачивается большая часть времени восприятия. Максимальная продолжительность осмотра приходится на зону носа и переносицы, в меньшей степени на зону рта и глаз. Также нетрудно заметить из табл. 2, что сильная экспрессия, такая как радость, страх и удивление распознается лучше, чем остальные. Полученные результаты и предложенные методы оценки интереса участников изображения применяются в работе по распознаванию эмоциональных состояний [7].

Заключение

Системы распознавания эмоций человека – это развивающиеся системы, используемые во многих областях деятельности человека, таких как научные исследования, маркетинг, психология, медицина, usability-тестирование, оценка эффективности печатной рекламы, дизайн и др. В данной работе были рассмотрены три подхода к определению эмоциональности человека:

1) метод, основанный на анализе видеоизображения и жестов рук. Использует данные о распознанном жесте для увеличения точности прогнозирования эмоций с использованием сенсора Leap Motion. Была описана модель жестов человека при проявлении таких эмоций, как удивление, счастье, страх, злость и печаль. В результате проведенного исследования было показано, что комбинирование традиционного подхода к определению эмоциональности человека по видео изображению с подходом, основанным на анализе движений рук человека, показывает высокую эффективность в задачах распознавания эмоций. С развитием методов, захвата, анализа движений и жестов рук человека, возможно добиться высокой точности прогнозирования всех, проявляемых человеком, эмоций;

2) метод на основе анализа тональности текста. Описывается программное обеспечение, позволяющее дать тональную оценку комментариев пользователей на заданном сайте. Тональная оценка определялась с помощью тональных словарей объемом 10000 слов. Тональность текста классифицировалась на три группы: позитивная, отрицательная или нейтральная. В результате проведенного эксперимента данный метод показал высокую точность определения эмоционального окраса текста;

3) метод, основанный на анализе мимики и движения глаз. Данный метод описывает закономерности перемещения взгляда человека при восприятии эмоциональных экспрессий. Описывается эксперимент, проведенный с участием 10 человек на аппаратно-программном ком-

плексе, представляющий собой приложение для просмотра, фиксации и визуализации перемещения взгляда пользователя на языке C# с использованием камеры Tobii Eye Tracker 4С. В результате проведенного эксперимента была получена общая продолжительность фиксаций в каждой выделенной лицевой области, а также выявлена точность распознавания испытуемыми эмоциональных экспрессий.

В дальнейшем для улучшения метода определения эмоций по анализу видеоизображения и жестов рук планируется улучшить алгоритмы захвата и анализа движения рук человека, что существенно понизит требования к обучению нейронных сетей, использующихся при распознавании эмоций на основе мимики человека. Для улучшения анализа тональности текста планируется обучить искусственную нейронную сеть для анализа комментариев, что позволит увеличить точность определения эмоционального окраса текста. Перспективой развития метода анализа мимики человека и движения глаз при восприятии эмоциональных экспрессий является применения других камер для детектирования зрачков человека с различной точностью и частотой распознавания и проведения сравнительного анализа полученных результатов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Leap Motion overview [Электронный ресурс] // Официальный сайт Leap Motion: [сайт]. URL <https://www.leapmotion.com/product/vr#113> (дата обращения: 16.01.2018)
2. Karpouzis, K., Raouzaoui, A. Facial Expression and Gesture Analysis for Emotionally-Rich Man-Machine Interaction // National Technical University of Athens, 2004.
3. Lhomme, M., Marsella, S. Expressing emotion through posture and gesture // Oxford Library of Psychology. 2014. 21 pages. (<http://people.ict.usc.edu/~gratch/CSCI534/Readings/ACIHandbook-GestureSyn.pdf>).
4. Duncan, D., Shine, G., English, C., Facial Emotion Recognition in Real Time // Stanford. 2015. 7 pages (http://cs231n.stanford.edu/reports/2016/pdfs/022_Report.pdf).
5. Challenges in Representation Learning: Facial Expression Recognition Challenge [Электронный ресурс] // Официальный сайт Kaggle: [сайт]. URL <https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge> (дата обращения: 16.01.2018).
6. Arkhipenko, K., Kozlov, I., Trofimovich, J., Skorniakov, K., Gomzin, A., Turdakov, D. (2016), Comparison of neural network architectures for sentiment analysis of Russian tweets. Proceedings of the International Conference "Dialogue 2016".
7. Rozaliev V.L., Orlova, Yu.A. Recognizing and Analyzing Emotional expressions in Movements / E-Learning Systems, Environments and Approaches. Theory and Implementation / ed. by P. Isaias [et al.]. – [б/м] : Springer, 2015. – Part II, § 9. – P. 117–131.
8. Soloshenko, A.N., Orlova, Yu.A., Rozaliev, V.L., Zaboloeva-Zotova, A.V. Establishing Semantic Similarity of the Cluster Documents and Extracting Key Entities in the Problem of the Semantic Analysis of News Texts // Modern Applied Science. – 2015. – Vol. 9, No. 5. – C. 246–268.

УДК 004.8

*В. А. Частикова, С. А. Жерлицын, Я. И. Воля***НЕЙРОСЕТЕВОЙ ПОДХОД К РЕШЕНИЮ ЗАДАЧИ
ПОСТРОЕНИЯ ФОТОРОБОТА ПО СЛОВЕСНОМУ ОПИСАНИЮ****Кубанский государственный технологический университет**

chastikova_va@mail.ru, kpytooooo@gmail.com, volya_y@mail.ru

Процедура составления фоторобота на данный момент является слабо оптимизированной и проводится на основе специальных программ-конструкторов, имеющих ряд недостатков. В статье описывается разработанный с помощью современных средств машинного обучения нейросетевой подход к решению задачи построения фоторобота человека по словесному описанию его внешности. Приведены принципиальные описания используемых технологий: word2vec – алгоритм представления слов в векторной форме таким образом, чтобы расстояние между полученными значениями было пропорционально смысловой разнице соответствующих слов; LSTM – рекуррентная нейронная сеть, способная учитывать долговременные и кратковременные зависимости во входном сигнале и отражать их как в своем выходе, так и во внутреннем состоянии; нейронная сеть транспозиционной свертки, позволяющая производить операции, обратные свертке для увеличения размерности данных и используемая в настоящем проекте для восстановления графического изображения из вектора чисел, так называемого контекста. Предложена структурная схема, состоящая из вышеописанных блоков, каждый из которых выполняет свои функции; описан принцип взаимодействия компонентов системы в целом, составляющий аппарат построения фоторобота.

Ключевые слова: фоторобот, рекуррентная нейронная сеть, разверточная нейронная сеть, word2vec, машинное обучение.

*V. A. Chastikova, S. A. Zherlitsyn, Y. I. Volya***NEURAL NETWORK APPROACH TO THE IDENTIKIT
COMPILATION PROBLEM THROUGH VERBAL DESCRIPTION****Kuban State Technological University**

The procedure of an identikit compilation is poorly optimized at the moment. It is performed with the help of special constructor programs which have a row of shortcomings. This article describes a technique of constructing a person's sketch by a verbal description of his appearance based on modern machine learning tools. The principal descriptions of used technologies are provided in the article. Word2vec is an algorithm of words' vectorial form representation that provides distance between obtained values proportional to semantic difference of their original words. LSTM is the recurrent neural network capable of considering long-term and short-term dependencies of input signal and reflecting them in both its output, and its internal state. The neural network of a transpositional convolution allows to make operations opposite to convolution such as increasing dimensionality of data which is used in this project for recovery of the graphics image from a vector of numbers, so-called context. The structural scheme, which consists of the blocks described above, is proposed. The principle of system components interaction, which constitutes the device for constructing the identikit, is described.

Keywords: identikit, recurrent neural network, transpositional neural network, word2vec, machine learning.

В настоящее время процедура составления фоторобота производится с помощью специальных программ, в которых оператору необходимо собирать лицо человека из множества составных частей – фрагментов лиц.

Данные элементы составляют довольно объемную базу, просмотр и примерка которой – крайне трудоемкий и длительный процесс. С возрастанием количества просмотренных элементов снижается внимательность и точность выбора [12]. Больше количество элементов начинают казаться похожими на оригинал, кроме того, может возникнуть эффект наду-

манности некоторых черт, важных для распознавания описываемого человека. Комплекс данных погрешностей может стать причиной неправдоподобного описания и, как следствие, затруднением в задержании преступника или поиска разыскиваемого лица.

Таким образом, с учетом высокой рабочей нагрузки на сотрудников, обязанных выполнять составление фоторобота, целесообразна автоматизация данного процесса.

Исследование направлено на решение задачи генерации изображения на основе текстовых данных, то есть на реализацию тексто-графи-

ческого преобразования с целью получения изображения лица человека из неформализованного словесного описания в свободной форме.

В статье предлагается использовать современные достижения в области машинного обучения для построения нейросетевого подхода к решению указанной проблемы [1]. Алгоритмическую базу составляют:

- Технология Word2vec – позволяет перевести слова русского языка в численные векторы таким образом, чтобы расстояние между полученными значениями было пропорционально смысловой разнице соответствующих слов.

- LongShortTermsMemory (LSTM) сеть – рекуррентная нейронная сеть с особой архитектурой, способная к обучению долгосрочным зависимостям.

- TransposedConvolutionNeuralNetwork (TCNN) – нейронная сеть транспозиционной свертки, производящая действия, противоположенные работе сверточной сети. В предложенной работе используется для получения изображения по вектору его признаков.

Входные параметры предлагаемой системы являются текстом описательного характера на русском языке в свободной форме. Единственное ограничение, вводимое для адекватности обучения, – это объем слов, который должен лежать в диапазоне от 100 до 200 на один пример.

На выходе происходит генерация черно-белого изображения лица фиксированного размера, принятого равным 300x400 пикселей, обеспечивающего приемлемый уровень детализации представленного на нем фоторобота.

Все входящие в состав системы сети обучаются на основе алгоритма градиентного спуска и его модификаций.

Word2vec

Данная технология была разработана компанией Google на основе анализа поисковых запросов, совершаемых пользователями. Подобный набор алгоритмов используется для расчета векторных представлений слов, причем таких, что семантически более близкие слова имеют наименьшее расстояние в векторном представлении. Похожими по смыслу считаются слова, находящиеся в похожих контекстах. Формально нахождение таких векторов сводится к максимизации косинусной близости, то есть скалярного произведения векторов, которым соответствуют схожие слова и минимизации для слов, не появляющихся в схожем контексте [2, 3].

В формульном виде максимизируемая функция будет иметь следующий вид:

$$\frac{(w_v \cdot w_c)}{\sum (w_{c1} \cdot w_v)},$$

где w_c – вектор рассматриваемого слова контекста; w_v – вектор целевого слова; w_{c1} – другое слово, которого нет в анализируемом контексте. Таким образом, в числителе вычисляется близость слов контекста, а в знаменателе – близость всех других слов. Однако вычисление знаменателя для каждого другого слова из всех возможных контекстов неэффективно и ресурсоемко, поэтому для оптимизации этого процесса применяются такие методы, как, негативное сэмплирование, удаление редких слов, изъятие наиболее частотных слов из анализа, построение дерева Хаффмана.

В рамках данного исследования использование технологии Word2vec обусловлено необходимостью преобразования информации, содержащейся в описании внешности человека, в вид, понятный для нейронной сети. Такое свойство, как схожесть векторных представлений слов, имеющих похожий смысл, делает систему более устойчивой к использованию синонимов при описании.

LongShortTermsMemory

Рекуррентная нейронная сеть представляет из себя обычную нейронную сеть [11], но имеющую обратные связи [4]. В развертке любая такая сеть является цепью одинаковых блоков, состояние каждого из которых зависит от предыдущего.

Потенциальный эффект от использования рекуррентной нейронной сети заключается в ее умении учитывать полученную ранее информацию и способности принимать на вход данные практически произвольной размерности. Однако по мере роста расстояния между наиболее важной информацией способность ее учета ослабевает.

Долгая краткосрочная память – разновидность архитектуры рекуррентных нейронных сетей, способная к обучению долгосрочным зависимостям [5]. Ее структуру все еще можно представить в виде цепочки одинаковых блоков, однако каждый из них должен иметь минимум четыре взаимодействующих между собой слоя:

- слой фильтра забывания – имеет сигмоидальную функцию активации и используется для определения, какую информацию сеть

должна забыть. Применяется к входному вектору сети, а его выход находится в диапазоне $[0, 1]$, где 1 означает полное сохранение информации, а 0 – полное удаление. Выполнение функции забывания реализуется с помощью поэлементного перемножения вектора состояния из предыдущего блока с выходом названного слоя;

- слой входного фильтра – работает аналогично слою фильтру забывания, и также применяется к неизменному входу сети на данном шаге, однако перемножение результата происходит с выходом следующего слоя – слоя обновления;

- слой обновления – использует в качестве функции активации гиперболический тангенс и принимает на вход то же самое, что и оба выше-

описанных слоя. Его выход поэлементно перемножается с выходом слоя входного фильтра; результат поэлементно прибавляется к вектору состояния, уже прошедшему фильтр забывания;

- выходной слой – имеет сигмоидальную функцию активации и определяет, какая часть входных данных будет являться выходом на текущем шаге аналогично слою фильтра забывания. На него подается все тот же неизменный вход данного шага, а выход поэлементно перемножается с вектором состояния, уже подвергнутым всем вышеописанным трансформациям, поэлементно пропущенным через гиперболический тангенс, чтобы его значения лежали в диапазоне от -1 до 1 .

Схематично система взаимодействия представленных блоков приведена на рис. 1.

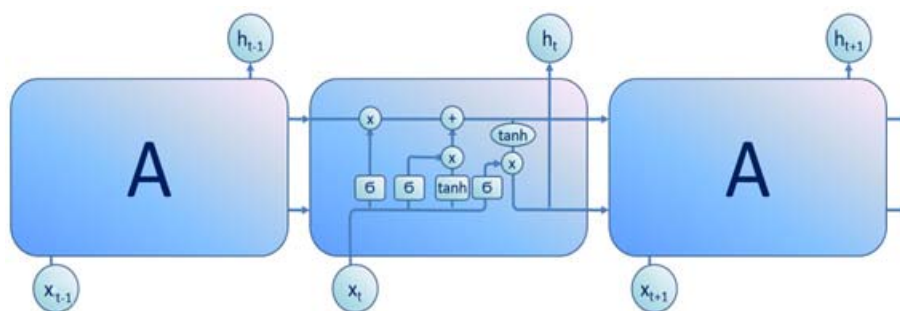


Рис. 1. Схема LSTM нейронной сети

На приведенном изображении стрелками обозначены пути перемещения векторов от одной операции к другой. Расходящиеся стрелки иллюстрируют копирование одних и тех же данных в двух направлениях. Операции в круглых (овальных) рамках означают, что описанная функция производится над векторами почленно: $\times$ – умножение; $+$ – сложение; $\tanh$ – гиперболический тангенс. Прямоугольные рамки соответствуют слоям нейронных сетей с указанной внутри функцией активации: σ – сигмоидальная функция.

Существуют и другие вариации долгой краткосрочной памяти, однако в рамках настоящего исследования рассматривается приведенная, так как именно такая структура является классической.

Основной причиной использования LSTM сети является не только переменная длина входных данных. Подобная сеть применяется в качестве инструмента формирования опорного вектора признаков для разверточной нейронной сети, поэтому важно получить вектор, который нес бы в себе как можно больше ин-

формации о внешности человека. Для этого предлагается считать только последний выход сети, в котором сжата вся необходимая информация благодаря использованию технологии долгой краткосрочной памяти.

Transposed Convolution Neural Network

Сверточная нейронная сеть – разновидность глубоких нейронных сетей, состоящая из трех базовых элементов, которые могут повторяться, менять свою последовательность и настройки в каждой отдельно взятой архитектуре [6–8]:

Слой свертки представляет из себя набор нейронов, также называемых ядрами или фильтрами, каждый из которых является матрицей, накладываемой на исходные данные. При наложении числа в матрице данных и матрице ядра на соответствующих позициях перемножаются, а результаты умножения складываются и записываются в соответствующую ячейку выхода слоя. Такой блок всегда является первым в любой сверточной нейронной сети; результатом его работы является карта признаков, отображающая, в какой части исходных данных выделенный признак проявляется сильнее.

ReLU-преобразование (rectifiedlinearunit) – подвергает каждый элемент матрицы данных $x_{i,j}$ преобразованию вида $\max(0, x_{i,j})$. Таким образом гарантируется отсутствие отрицательных значений признака. ReLU-преобразование проводится после каждой свертки.

Pooling-слой – существует несколько разновидностей пулинга (объединения): по максимальному значению, по среднему значению и другие, однако наиболее распространенным на практике является первый из выше перечисленных. Его сущность заключается в уплотнении и уменьшении размерности карты признаков путем отбрасывания в равных, непересекающихся и покрывающих всю матрицу данных, прямоугольниках всех значений, кроме максимальных.

В конце сверточной сети обычно располагают многослойный персептрон [9, 10], интерпретирующий полученные данные.

Нейронная сеть транспозиционной свертки основана на принципе работы сверточной сети, модифицированной таким образом, что принцип ее работы меняется на противоположный. Данная сеть принимает на вход вектор и, про-

изводя ряд преобразований, выдает на выходе матрицу, которая может быть интерпретирована как изображение.

В отличие от сверточной сети, сеть транспозиционной свертки производит операции, обратные свертке и пулингу вместо них самих. При этом для unpooling необходимо знать, в какую область данных необходимо поместить известное максимальное значение, что невозможно без наличия симметричной сверточной сети. Операция, обратная свертке, осуществляется благодаря искусственному равномерному расщеплению данных и применению к ним аналогичных разверточных фильтров.

В рассматриваемом проекте предлагается использовать сеть, состоящую из персептрона, ReLU-преобразования и транспозиционной свертки, за счет которой и будет осуществляться повышение размерности данных из вектора признаков до полноценного изображения.

Таким образом, предлагаемый нейросетевой подход для построения фоторобота по словесному описанию включает в себя следующие блоки (рис. 2):



Рис. 2. Блочное представление системы построения фоторобота по словесному описанию

Каждое слово описания внешности человека преобразуется в вектор по словарю, полученному с помощью технологии Word2vec, после этого список векторов поочередно поступает на вход LSTM сети.

Данная сеть на своем последнем выходе формирует описание признаков внешности в виде вектора. Он, в свою очередь, подается на вход полносвязного слоя транспозиционной сети, формирующей итоговое изображение.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Хайкин, С. Нейронные сети. Полный курс ; пер. с англ. – 2-е изд. – М.: Вильямс, 2006.
2. Tomas, Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, Jeffrey Dean. Distributed Representations of Word and Phrases and their Compositionality. In Proceedings of Workshop at The Twenty-seventh Annual Conference on Neural Information Processing Systems (NIPS), 2013.
3. Word2Vec. How does it work URL: <https://code.google.com/archive/p/word2vec/> (дата обращения: 20.11.2017).
4. Hochreiter, S., Schmidhuber, J. Long short-term memory. Neural computation, 1997. Vol. 9, no. 8. Pp. 1735–1780.

5. *PoghosyanAghasi, SarukhanyanHakob*. RNN with additional constant memory for image caption generation task. Киев: Web of Scholar, 2017.
6. *LeCun Y., Bengio Y.* Convolutional networks for images, speech, and time series. The handbook of brain theory and neural networks, 1995. Т. 3361. №. 10. P. 1995.
7. *Quoc V. Le, JiquanNgiam, Zhenghao Chen, Daniel Chia, Pang Wei Koh, Andrew Y. Ng.* Tiled convolutional neural networks. Computer Science Department, Stanford University, 2010.
8. *Alex Krizhevsky, IlyaSutskever, Geoffrey E. Hinton.* ImageNet Classification with Deep Convolutional Neural Networks. Advances in neural information processing systems, 2012, Pp. 1097–1105.
9. *Частикова, В.А., Малыхина, М.П., Жерлицын, С.А., Воля, Я.И.* Сравнительный анализ некоторых алгоритмов роевого интеллекта при обнаружении сетевых атак нейросетевыми методами // Политематический сетевой электронный научный журнал Кубанского государственного аграрного университета, 2017. № 129. С. 106–115.
10. *Частикова, В.А., Власов, К.А., Картамышев, Д.А.* Обнаружение DDoS-атак на основе нейронных сетей с применением метода роя частиц в качестве алгоритма обучения // Фундаментальные исследования. 2014. № 8–4. С. 829–832.
11. *Малыхина, М.П., Бегман, Ю.В.* Нейросетевая экспертная система на основе прецедентов для решения проблем обслуживания абонентов сотовой сети // Известия высших учебных заведений. Северо-Кавказский регион. Серия : Технические науки. 2009. № 3. С. 6–9.
12. *Белов, Д.Л., Антипова, О.Ю., Частикова, В.А.* Методы решения задач с конфликтными ситуациями в системах принятия решений // Труды Кубанского государственного технологического университета. 2000. Т. 7. № 1. С. 153–159.

УПРАВЛЕНИЕ В СОЦИАЛЬНЫХ И ЭКОНОМИЧЕСКИХ СИСТЕМАХ

УДК 004.42
X 20

Е. Г. Крушель, А. Э. Панфилов, И. М. Харитонов

АНАЛИЗ ЭФФЕКТИВНОСТИ ИЗУЧЕНИЯ ПРОФИЛИРУЮЩИХ УЧЕБНЫХ ДИСЦИПЛИН В ВУЗЕ НА ПРИМЕРЕ КТИ (ФИЛИАЛА) ВолгГТУ

Камышинский технологический институт (филиал)
Волгоградского государственного технического университета
elena-krushel@yandex.ru, pansanja@yandex.ru, wisdom_monk@mail.ru

Предлагается анализ успеваемости студентов направления 09.03.01 – «Информатика и вычислительная техника» Камышинского технологического института (филиала) Волгоградского государственного технического университета по двум профилирующим учебным дисциплинам учебного плана. Анализ дисциплин, изучаемых на первом и последнем курсе позволяет проследить динамику изменения успеваемости. Группы студентов разбиваются на классы, в зависимости от их успеваемости. Производится анализ «перетекания» классов студентов по годам обучения. Выявляется проблема недостаточной эффективности использования преподавательского ресурса, в связи с чем предлагаются пути ее решения с целью повышения эффективности подготовки выпускников вуза.

Ключевые слова: профилирующие дисциплины, классы студентов, анализ успеваемости, эффективность подготовки выпускников.

E. G. Krushel, A. E. Panfilov, I. M. Kharitonov

THE RESULTS OF PROFILING EDUCATION DISCIPLINES STUDYING EFFICIENCY ANALYZING IN THE HIGHER EDUCATION INSTITUTE BY THE EXAMPLE OF KTI (BRANCH OF VSTU)

Kamyshin Technological Institute (branch)
Volgograd State Technical University

The analysis of the 09.13.01 “Computer science and the computer engineering” specialty students’ education performance is presented based on the education results of the Kamyshin technologic higher education institute (branch of the Volgograd state technical university) students. Two profiling education disciplines were chosen for the investigation of their studying efficiency relationship with the education results obtained at the preceding semesters. The clusters of the students’ educational rating values achieved at the first semesters were determined, and the clusters’ dynamic changes at the following semesters were analyzed. The professors’ resource employment insufficiency problem was detected, and the directions of its overcoming are proposed for the purposes of the high school graduates future successful activity raise.

Keywords: profiling educational disciplines, students’ clusters, educational results analyze, graduates’ education quality.

Введение

Одним из важнейших направлений российской образовательной профессиональной политики в современных условиях является обеспечение высокого качества профессиональной подготовки студентов на основе сохранения ее фундаментальности, соответствия актуальным

и перспективным потребностям личности, общества и государства.

Особую значимость и актуальность в современных условиях приобрела проблема повышения эффективности подготовки студентов в высших учебных заведениях к профессиональной деятельности. Результаты исследова-

ния свидетельствуют о том, что данная проблема является первостепенной для большинства вузов страны. В связи с этим, особую актуальность в настоящее время приобретают работы, направленные на исследование процесса эффективности подготовки выпускников вузов [1].

Анализ качества подготовки выпускников вуза показал [2], что в последние годы качество выпускающихся будущих специалистов снижается. В связи с этим необходимо изучение складывающейся ситуации на более ранних этапах учебы.

Авторами предлагается анализ успеваемости студентов направления 09.03.01 – «Информатика и вычислительная техника» Камышинского технологического института (филиала) Волгоградского государственного технического университета по двум основным дисциплинам учебного плана – «Информатика» и «Основы теории управления» в периоды с 2011 по 2014 годы. Дисциплины читаются на первом и четвертом курсе обучения одним и тем же преподавателем и относятся к одной группе профилирующих дисциплин, что позволит частично избежать субъективности в получаемых оценках. Анализ данных дисциплин показывает динамику

изменения успеваемости студентов за время их обучения в вузе.

Анализ оценок по дисциплине «Информатика»

Данная дисциплина читается на первом курсе в первом семестре, и является одной из приоритетных для направления обучения. Дисциплину в течение нескольких лет читает один и тот же преподаватель. Зачет проходит в виде тестирования, поэтому, по данной дисциплине студенты получают заработанные, справедливые оценки. Данные взяты из рейтинга деканата по результату сессии за первый семестр.

В таблице занесены данные: группа, количество студентов в группе, год набора и количество студентов получивших определенные оценки по предмету. На рисунке черным жирным цветом отмечено количество студентов, получивших минимальный балл – 61 (он приравнивается к оценке «2») по предмету; тонким серым цветом – от 62 до 75 баллов; пунктиром – от 76 до 89 баллов и штрихами – от 90 до 100 баллов. Также указаны колонки с количеством студентов в группе.

На рис. 1 представлена успеваемость по дисциплине «Информатика» за период с 2011–2014 гг.

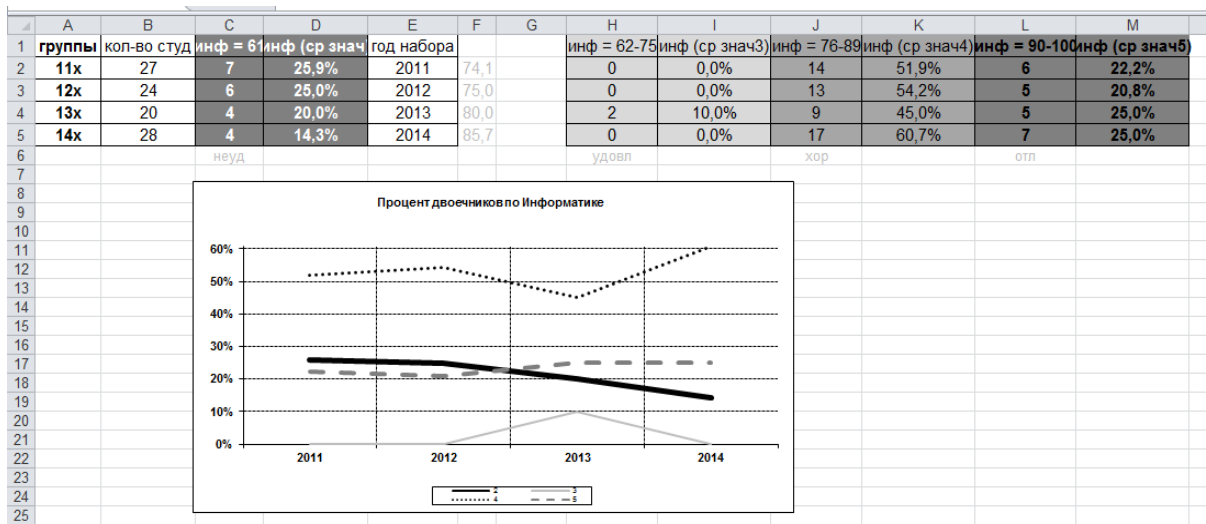


Рис. 1. Успеваемость по дисциплине «Информатика»

Гистограммы на рис. 2 показывают, что за последние годы по данной дисциплине наблюдается тенденция «размытия» среднего класса студентов-«троечников» и происходит их «перетекание» либо в класс «двоечников», либо в класс «хорошистов».

Таким образом, для большинства вузов (не считая самые престижные, выпускающие са-

мых сильных студентов) встает проблема неэффективности использования преподавательского ресурса. В основном, учебные программы рассчитаны на средних студентов, учащихся на «хорошо» и «удовлетворительно», которые после выпуска и составляют основу для большинства рабочих мест на рынке труда.

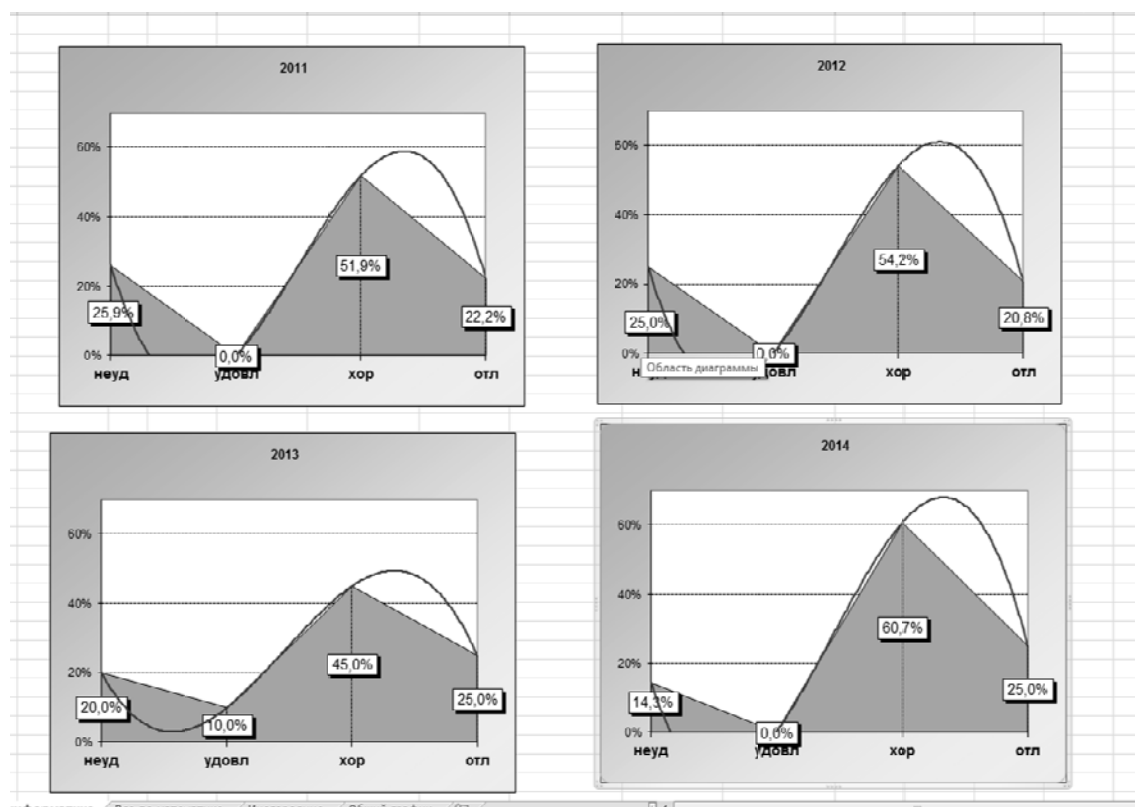


Рис. 2. Построение гистограмм по годам («Информатика»)

Исчезновение этой основной группы студентов ставит преподавателя перед необходимостью перестраивать процесс преподавания дисциплины либо для слабых студентов (но тогда сильным студентам будет неинтересно учиться и они получают меньше знаний), либо наоборот для сильных (но тогда слабые студенты не будут ничего понимать в дисциплине и уйдут из института). Если же разбивать процесс преподавания дисциплины на два разных потока, то это ведет к серьезному перегрузу самого преподавателя, имеющего и так очень высокую нагрузку после изменения часовых нормативов.

Одним из решений данной проблемы для малого вуза может быть безфакультетская структура. Авторами предлагается создать общетехнический факультет, на котором будут обучаться все студенты технических специальностей на протяжении первых двух лет обучения.

Разделения на учебные группы будут производиться не по признаку специальности, а по возможностям восприятия сложного материала. Лекции для всех групп можно читать одинаково, а практики и лабораторные разделить, в зависимости от способностей студентов. Поскольку получатся большие потоки, то количе-

ство преподавателей хватит на всех и не нужно, например, будут разделения преподавателей между факультетами.

После второго курса уже необходимо будет передавать студентов по направлениям обучения – на кафедры, где они и будут продолжать обучение в зависимости от своих предпочтений к выбранной специальности и от своих способностей в выбираемой области знаний.

Дополнительным достоинством такого решения является выполнение требований «дорожной карты» по увеличению числа студентов на одного преподавателя. В среднем, «плохих» студентов можно распределить на преподавателя в большем количестве, т. к. уровень их способностей не позволяет успешно овладеть сложным теоретическим материалом, а для «хороших» студентов можно выделить лучших преподавателей и, таким образом, снизить нагрузку на них за счет уменьшения числа студентов, приходящихся на этих преподавателей [3].

Анализ оценок по дисциплине «Основы теории управления»

Данная дисциплина читается на последнем курсе и является одной из приоритетных для направления обучения. Дисциплину в течение

нескольких лет читает один и тоже преподаватель, не изменяющий свои требования в рамках данной дисциплины.

На рис. 3 представлены исходные данные для анализа успеваемости по данной дисциплине.

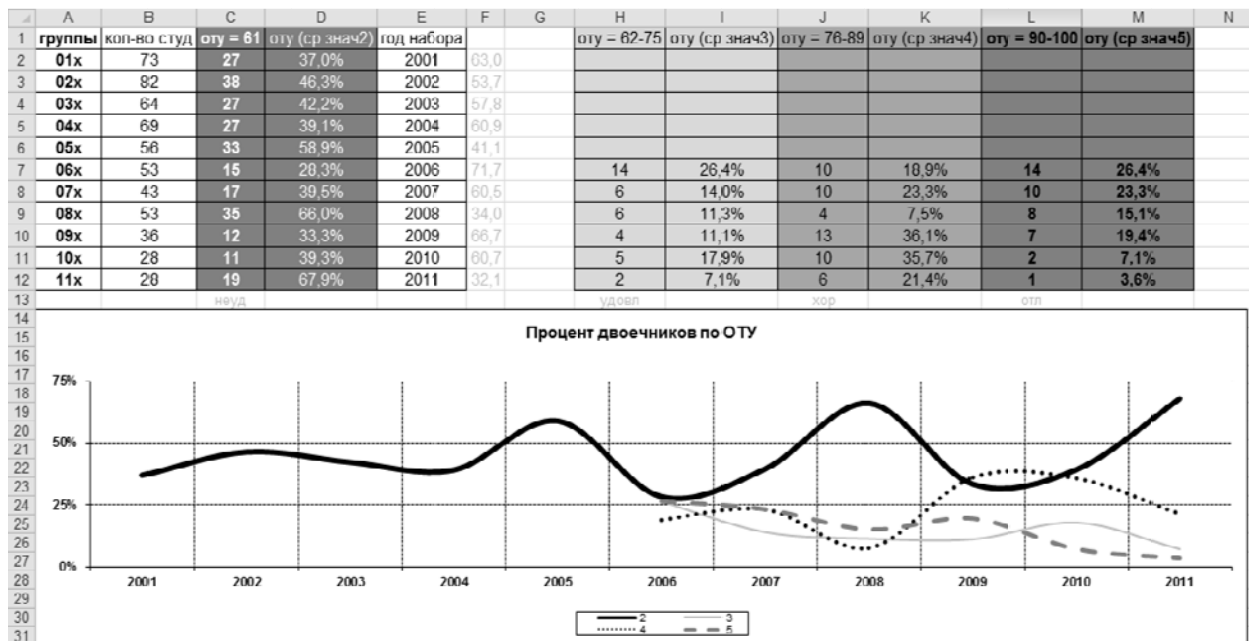


Рис. 3. Успеваемость по дисциплине «Основы теории управления»

На рис. 4 показаны гистограммы, которые отображают успеваемость по данной дисциплине.

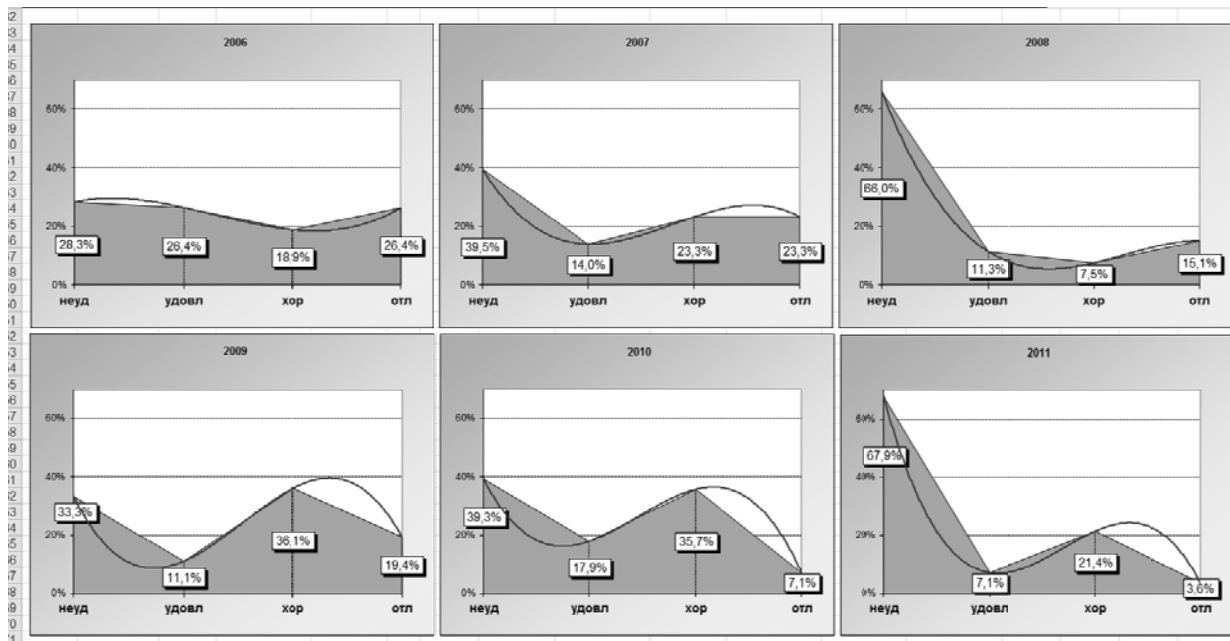


Рис. 4. Построение гистограмм по годам («Основы теории управления»)

Данные взяты из архива сводных ведомостей по результату сессии.

В таблицу занесены следующие данные: группа, количество студентов в группе, год набора и количество студентов, получивших оп-

ределенные оценки по предмету. На рисунке черным жирным цветом отмечено количество студентов, получивших минимальный балл – 61 (он приравнивается к оценке «2») по предмету; тонким серым цветом – от 62 до 75 баллов;

пунктиром – от 76 до 89 баллов и штрихами – от 90 до 100 баллов. Также указаны колонки с количеством студентов в группе.

Гистограммы на рис. 4 показывают, что за последние годы по данной дисциплине также наблюдается тенденция «размытия» среднего класса студентов-«троечников» и происходит их «перетекание» либо в класс «двоечников», либо в класс «хорошистов».

Но по сравнению с более простой в изучении дисциплиной «Информатика», изучаемой на первом курсе, где «перетекание» «троечников» наблюдалось в основном в класс «хорошистов», в данной, более сложной для изучения, дисциплине наблюдается «перетекание» «троечников» в класс «двоечников», что не только негативно влияет на «размытие» среднего класса студентов, но и на общее снижение эффективности процесса обучения.

Вывод

Анализ основных профилирующих дисциплин показал одинаковую картину – «размытие» среднего класса успевающих студентов. Все это подтверждает актуальность работ, направленных на исследование эффективности процесса обучения в вузе [4].

В ходе проведения анализа также было установлено, что эффективность качества подготовки выпускников в Камышинском технологическом институте (филиале) Волгоградского государственного технического университета падает.

Главной причиной такого явления можно назвать нежелание сильных абитуриентов продолжать обучение в родном городе, считая такой вариант непрестижным. Из-за этого происходит «отток» потенциальных хороших будущих специалистов в другие города, снижая тем самым общий уровень профессиональных работников города. Результаты «оттока» подтверждаются собранной статистикой средних

рейтинговых оценок абитуриентов по ЕГЭ за последние годы. В результате такого «оттока» не только снижается количество будущих грамотных выпускников института (что также подтверждается проведенными исследованиями), но и один из критериев оценки эффективности вуза, отвечающий за средний балл ЕГЭ абитуриентов. Все это дополнительно ставит вуз и город в сложное положение.

Одним из способов улучшения текущей ситуации является повышение привлекательности вуза, для чего можно предложить следующие рекомендации:

1. Назначение лучшим выпускникам института дополнительных материальных надбавок при трудоустройстве в родном городе.
2. Льготы лучшим выпускникам института при трудоустройстве в организациях и на предприятиях родного города.
3. Помощь в приобретении или получении жилья для молодых специалистов – лучших выпускников института.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Харитонов, И. М. Моделирование процесса построения учебного плана на основе формализованного представления учебной дисциплины / И. М. Харитонов // Открытое образование. – М., 2011. – № 2. Ч. 1. – С. 21–32.
2. Харитонов, И.М. Применение контент-анализа для оценки терминологической связи между учебными дисциплинами / И.М. Харитонов, Е.Н. Скрипченко // Известия Волгоградского государственного технического университета. – 2010. – № 11(71). – С. 114–117.
3. Харитонов, И.М. Прогнозирование уровня освоения компетенций выпускниками вуза на основе рейтинговых оценок абитуриентов / И. М. Харитонов, Е. Г. Крушель // Современные проблемы науки и образования. – 2012. – № 6. – С. 43.
4. Харитонов, И.М. Модели и алгоритмы планирования учебного процесса вуза на основе модульно-компетентного подхода : дис. ... канд. техн. наук / И. М. Харитонов ; Астраханский государственный университет. – Астрахань, 2011. – 202 с.

УДК 004.94

*Т. М. Петрова, Т. К. Смыковская, Ю. А. Машевская***ИСПОЛЬЗОВАНИЕ GOOGLE-ДОКУМЕНТОВ ПРИ КОНСТРУИРОВАНИИ
ИНДИВИДУАЛЬНЫХ ОБРАЗОВАТЕЛЬНЫХ ТРАЕКТОРИЙ****Волгоградский государственный социально-педагогический университет**

profptm@gmail.com, smikov_t@mail.ru, mash_mif@mail.ru

В статье представлена модель использования Google-документов как инструментального средства при конструировании преподавателем индивидуальных образовательных траекторий. Описаны примеры экспериментальной работы, показаны возможности использования в образовательной практике. Представлено авторское видение проблемы конструирования индивидуальных образовательных траекторий в рамках дисциплины, учебной темы, учебного занятия, что позволяет решить задачу частичной автоматизации конструирования индивидуальных образовательных траекторий.

Ключевые слова: индивидуальная образовательная траектория, конструирование педагогических объектов, Google-документы, ИКТ.

*T. M. Petrova, T. K. Smykovskaya, Yu. A. Mashevskaya***USING GOOGLE DOCUMENTS TO CONSTRUCT
INDIVIDUAL EDUCATIONAL TRAJECTORIES****Volgograd State Socio-Pedagogical University**

The article presents a model of using Google documents as a tool in the construction of individual educational trajectories by a teacher. The examples of experimental work are described, the possibilities of using in educational practice are shown. The author's vision of the problem of constructing individual educational trajectories within the discipline, educational theme, educational lessons is presented, which allows solving the problem of partial automation of constructing individual educational trajectories.

Keywords: individual educational trajectory, construction of pedagogical objects, Google documents, ICT.

В последние годы актуализировалась проблема создания и реализации индивидуальных образовательных траекторий в образовательном процессе. Проблеме конструирования и реализации индивидуальных образовательных траекторий посвящены исследования В.А. Слестенина, А.В. Хуторского и др. (личностно-ориентированный подход); В.В. Афанасьева, И.М. Осмаловской и др. (вариативный подход); Н. Б. Лаврентьевой, М.Д. Миронова и др. (модульный подход); В.А. Кузнецовой, В.М. Монахова и др. (дифференцированный подход); Т.К. Смыковской и др. (теория проектирования педагогических объектов).

Индивидуальная образовательная траектория (ИОТ) – это траектория, по которой каждый конкретный обучающийся продвигается в развитии ИКТ-компетентности; обеспечивает выбор обучающимся при педагогической поддержке преподавателя модели освоения элемента содержания (дисциплина, учебная тема, учебное занятие), форм организации собственной учебной деятельности и логики выполнения практических заданий на лабораторных работах [3].

В качестве инструмента автоматизации процесса конструирования индивидуальных образовательных траекторий нами были определены Google-документы и Google-сервисы, что согласуется с позициями, представленными в статье [2]. Представим оценку возможностей Google-документов для решения различных задач конструирования индивидуальных образовательных траекторий.

Одним из инструментов, позволяющих обрабатывать и хранить данные, является система облачного хранения данных – Google Диск (не требуется установки на компьютер пользователя; доступ к данным осуществляется с любого пользовательского компьютера или гаджета, подключенного к сети Интернет; содержит различные приложения, которые дают возможность пользователю создавать и редактировать документы, таблицы, презентации и формы, при этом проблема сохранности файлов не является для пользователя первоочередной). GoogleDocs функционально текстовому и табличному процессорам. Мы предлагаем использовать при конструировании индивидуальных образовательных траекторий следующие функции:

– загрузка документов Word и преобразование их в документы Google, что позволяет пользователю работать в привычных форматах текстовых документов процессора Word, не заботясь при переносе файла в Документы Google о несоответствии типов файлов, а соответственно о потере данных;

– совместный доступ к документу, что позволяет другим пользователям работать с созданным конкретным пользователем документом: читать, редактировать, комментировать, при этом возможна настройка уровней доступа для каждого пользователя;

– параллельное online-редактирование в режиме реального времени, что создает условия для просмотра истории изменений конкретного документа (например, учитель может просмотреть, как выполнялись задания учащимися в логике освоения индивидуальной образовательной траектории);

– добавление различных элементов в Google-документы: изображений, рисунков, формул, таблиц, ссылок, специальных символов;

– перевод документов на другие языки, что дает возможность учителю и ученикам расширять кругозор, а так же повышать степень коммуникаций.

Существуют десятки бесплатных дополнений к Документам Google, которые расширяют

функционал сервиса. Чтобы их установить, нужно перейти в пункт верхнего меню **Дополнения** → **Установить** → **Дополнение** и выбрать необходимые дополнения из представленного списка. Так, можно установить: MathType, чтобы вставить уравнение или график; TagCloudGenerator, чтобы создать в правом сайдбаре словарное облако; Kaizena, чтобы оставлять голосовые комментарии; LucidChart, который позволяет вставлять графики, диаграммы и интеллект-карты и др.

При построении индивидуальной образовательной траектории для каждого обучающегося важно создать такую ситуацию выбора, которая стала бы для него стимулом для движения вперед при освоении учебных дисциплин [3]. Организация индивидуальной работы требует быстрого и корректного взаимодействия как между учителем и учеником, так и между учащимися. Решение этой проблемы при создании индивидуального маршрута движения наиболее продуктивно можно осуществить с помощью приведенных выше Google-сервисов. Например, с помощью Google-таблиц (рис. 1) можно проследить продвижение ученик при освоении темы: в соответствующей ячейке ставится отметка о прохождении учебной темы или выполнении конкретного задания по теме или учебному занятию.

	A	B	C	D	E	F
1	Параграф	4.1		4.2		
2	Задание	Сравнение "бумажных" и "компьютерных" технологий	Мини-исследование "Электронные устройства для работы с текстами"	Набор текста	Устранение ошибок в готовом тексте	Работа с фрагментами текста
3	Алексеев Н.					
4	Анкудинов С.					
5	Богатырев А.					
6	Быков В.					
7	Воеводин М.					
8	Востриков А.					

Рис. 1. Таблица продвижения по ИОТ

Данная таблица может быть дополнена связями с другими объектами: шаблонами, опросниками, тестами и т. п.

Так, в Галерее шаблонов, предлагаемых приложением **Документы**, в разделе «Образование» учитель может выбрать для своей работы шаблоны лабораторных работ, планов уроков. Предлагаемые шаблоны уже содержат стандартные разделы, которые чаще всего встречаются в соответствующих документах.

Например, в шаблоне «Доклад» структура подразумевает содержание таких разделов, как аннотация, содержание, результаты, список литературы и т. д.

Сервис Google Формы позволяет с помощью шаблонов создать анкету, опросник или другую форму, подразумевающую получение обратной связи. При создании опросников могут быть использованы следующие типы вопросов: 1) короткий текст, который позволяет

отвечающему вписать короткий ответ; 2) длинный текст для внесения развернутых ответов; 3) выбор «один из списка»; 4) выбор «несколько из списка»; 5) раскрывающийся список; 6) шкала, которая позволяет оценивать указанный критерий в промежутке от 0 до 10; 7) сетка, дающая возможность выбрать точки в сетке, состоящей из столбцов и строк.

При работе с Google-сервисами учащимся предоставляется возможность более активно осваивать учебный материал, так как они могут самостоятельно выбирать траекторию своего движения по теме (разделу), уровень заданий в зависимости от уже приобретенных умений. Использование предлагаемых технологий показывает учащимся, что они становятся «соучастниками» в построении учебного занятия, привлекает их к современным доступным информационным технологиям, дает возможность создавать свои документы в таком формате, чтобы они стали доступны в любой момент обращения к ним с любого электронного устройства, подключенного к сети Интернет [3].

Опираясь на работы В. И. Загвязинского [1], можно выделить следующие методы конструирования индивидуальных образовательных траекторий:

- метод аналогий, построенный на использовании опыта проектирования, предусматривающий выработку типовых структур, определение различных рамок, условий и механизмов их применения;

- экспертно-аналитический метод, базирующийся на изучении рекомендаций и предложений экспертов, ориентированный на выявление специфики структур, возможных недостатков;

- структуризация целей, предусматривающая выработку системы целей, ориентиров, последующее совмещение их с разрабатываемыми структурами;

- организационное моделирование, представляющее собой разработку формализованных математических, графических и других отображений, распределения полномочий и ответственности проектировщиков, являющееся базой для построения вариаций структур.

При рассмотрении вопроса форм организации учебной деятельности учащихся, соответствующих содержанию изучаемых тем и позволяющих повысить познавательную активность каждого ученика при освоении учебного материала с учетом собственных индивидуальных

особенностей, учитель может использовать следующие формы: мини-исследования, динамические четверки, самостоятельная работа, групповая работа, индивидуально-групповая работа, парная работа.

После выбора формы организации учебной деятельности учащихся учитель разрабатывает систему заданий для практических и лабораторных работ по изучаемому разделу. При этом используются любые дидактические материалы.

Для отслеживания прохождения каждым учеником его индивидуальной траектории учителю рекомендуется создать систему документов. В качестве наиболее эффективных инструментов ведения подобной документации мы предлагаем использовать сервисы Google (Документы, Таблицы и др.).

Одним из подходов при построении индивидуальных образовательных траекторий при вхождении в тему является выявление у учащихся связи между понятиями и операциями, которые связаны с этим понятием. Выявленные связи мы можем разделить на 3 уровня: 1) восприятия (теоретическое представление о связи между понятием и действием, которое надо выполнить для решения поставленной задачи); 2) воспроизведения (выполнение заданий по образцу с затруднениями, если отсутствует четкая последовательность действий); 3) оперирования (владение понятиями и знание действий, приводящих к решению поставленной задачи при полном отсутствии инструкции). Определяется максимальный порог, на котором может работать обучающийся: на уровне восприятия, воспроизведения или оперирования. В соответствии с полученными результатами учитель строит индивидуальные образовательные траектории по уровню сложности содержания для каждого уровня.

Другой подход построения индивидуальных образовательных траекторий осуществляется через использование различных видов предлагаемых заданий [3].

Рассмотрим систему организации изучения темы «Обработка текстовой информации» (7 класс) с использованием индивидуальных образовательных траекторий, при этом построение ИОТ целесообразно осуществлять через использование заданий различного уровня сложности.

Работа с текстовыми файлами выполнялась учащимися и раньше. Начиная с младшей школы, учащиеся делают доклады, готовят презентации, участвуют в конкурсах и проектах, тре-

бующих оформления документов. Поэтому данная тема не является абсолютно новой для большей части учащихся.

Начать изучение темы можно с задания, которое будет выполняться в малых группах по 6–8 человек, на сравнение возможностей «бумажных» и «компьютерных» технологий: «В чем сильные и слабые стороны документа, представленного на бумаге и в электронном виде?». На обсуждение в группах дается 5 минут, после чего выводы, сделанные учениками, систематизируются и сводятся в единую таблицу.

На следующем этапе изучения темы на этом же уроке в качестве мини-исследования учащимся можно предложить анализ электронных устройств, работающих с документами: их достоинства и недостатки. Итоги исследования оформить в виде таблицы. При этом удобнее использовать Google Таблицу, так как она дает возможность работы с одним документом сразу нескольким пользователям одновременно.

После этого учитель переходит к разделу «Создание текстовых документов на компьютере», основными рассматриваемыми вопросами которого являются: 1) набор (ввод) текста (зоны «ответственности» пальцев на клавиатуре, правила при вводе текста); 2) редактирование текста (клавиши и сочетания клавиш для перемещения курсора по тексту, прокрутка документа, режим вставки, проверка правописания, способы устранения ошибок, поиск и замена в документе); 3) работа с фрагментами текста.

При изучении этого раздела учитель может провести диагностику имеющегося уровня связи между понятиями и операциями, которые связаны с этим понятием (назовем эти уровни – *уровнями готовности к работе с учебным материалом*), у учащихся. Такой анализ позволит учителю выстроить индивидуальные траектории обучения в рамках освоения учебной темы.

Так как учитель должен быть уверен в успешном освоении материала по теме, то построение индивидуальных образовательных траекторий мы предлагаем осуществлять по уровням знаний и умений, имеющимся у учащихся.

Для определения уровня готовности к работе с учебным материалом предлагаем учителю провести небольшой опрос, который можно организовать с помощью Google Форма, направленный на выявление основных знаний по изучаемой теме.

Учащимся предлагается ответить на следующие вопросы: 1) как осуществляется переход на новую строку в текстовом редакторе? 2) с помощью каких клавиш можно написать заглавную букву? 3) чтобы скопировать фрагмент текста, нужно... На первый и третий вопросы предлагается выбор одного варианта из нескольких, на второй вопрос – два варианта из нескольких. Каждый верный полный выбор оценивается в один балл.

После прохождения опроса учитель может просмотреть результаты и разделить учащихся на следующие группы: 0 баллов – учащиеся не имеют представления о работе с текстовыми документами на компьютере; 1 балл – учащиеся имеют первичные представления о работе с текстовыми документами на компьютере, могут набрать простой, не структурированный текст; 2 балла – учащиеся способны работать с текстовым документом в расширенном объеме: набирать, работать с фрагментами текста.

С помощью инструментов Google Форма учитель может просмотреть ответы учащихся не только по баллам, но и оценить их по содержанию. Для этого нужно перейти на закладку «Ответы» используемой формы, расположенной сверху окна, и на вкладке «Сводка» посмотреть общие результаты в виде гистограмм, а на вкладке «Отдельный пользователь» ознакомиться с результатами конкретного ученика. Автоматизация процесса проверки ответов с помощью встроенных инструментов Google Форма значительно ускоряет и упрощает процесс обработки результатов опроса.

В то время, когда учитель работает с результатами опроса, ученикам дается следующее задание (набрать небольшой текст) для того, чтобы учитель мог определить уровень умения набора текста и разделить учащихся на группы: высокая скорость набора и знание правил набора текста, средняя скорость набора и знание не всех правил набора текста, низкая скорость набора текста и незнание правил набора текста.

Для распределения учащихся на группы предлагается текст, который потребует знания тех правил, которые рассматриваются в тексте параграфа: переход на новую строку, использование прописных букв, написание знаков препинания (запятых, точек, знаков вопроса и восклицания, тире, дефиса).

После организации групп учитель предлагает учащимся выполнить задания в зависимости от группы, в которую включается ученик. При

этом движение по теме учитель может отмечать с помощью Таблицы Google.

Рассмотрим построение индивидуальных образовательных траекторий при движении по разделу «Создание текстовых документов на компьютере» для различных групп учащихся.

1-я группа (высокая скорость набора и знание правил набора текста):

1-е задание (1 балл). «В предложенном тексте исправьте ошибки, которые были допущены при наборе. Для проверки орфографии используйте соответствующие опции текстового процессора». В предлагаемом тексте ученикам необходимо вставить пробелы, устранить лишние символы, «склеить» разорванные строки и т.д. Можно приготовить несколько вариантов «неправильных» текстов, чтобы каждый работал со своим фрагментом.

Это же задание можно оформить в виде се-

тевого документа, над которым ученики работают совместно;

2-е задание (дифференцировано по уровням). В зависимости от уровня притязаний ученика он может выбрать выполнение одного из предложенных заданий, либо оба. Во втором случае баллы за выполнение задания суммируются.

(1 балл) «В предложенном тексте поменяйте местами 2-й и 3-й абзацы; продублируйте первый абзац, вставив его после 2-го и после 3-го абзацев».

(2 балла) «Составьте памятку для своих товарищей по использованию горячих клавиш и кнопок на панели инструментов для работы с фрагментами текста и оформите в соответствующем стиле».

Таким образом, в этой группе можно сконструировать следующие траектории (рис. 2):

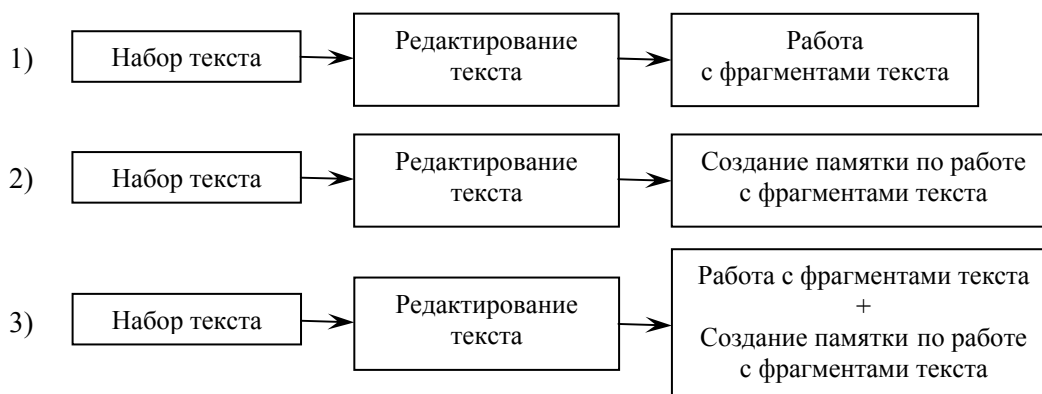


Рис. 2. Пример ИОТ

Для 2-й группы (средняя скорость набора и знание не всех правил набора текста) организуется следующая схема работы на уроке:

1. Просмотр презентации и выполнение контрольных заданий (<http://lbz.ru/metodist/authors/informatika/3/eor7.php> – сайт электронной поддержки учебника) под руководством учителя.

2. Выполнение заданий, приведенных выше для 1-й группы № 1 и № 2 (1 балл). Объективно

оценивая возможности учащихся, выполнение более сложной части задания № 2 (2 балла) в течение урока будет невозможным.

3. В качестве домашнего задания – работа с тренажером клавиатуры.

Таким образом, для этой группы учащихся индивидуальная образовательная траектория будет выглядеть следующим образом (рис. 3):

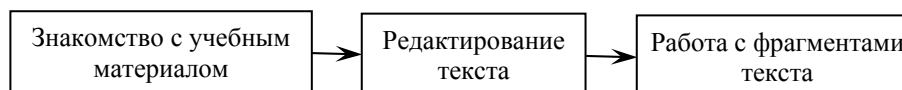


Рис. 3. Пример ИОТ

Для 3-й группы (низкая скорость набора текста и незнание правил набора текста) последовательность работы на уроке будет следующей:

1. Просмотр презентации и выполнение контрольных заданий (<http://lbz.ru/metodist/authors/informatika/3/eor7.php> – сайт электронной поддержки учебника) под руководством учителя.

2. Работа с клавиатурным тренажером под руководством учителя.
3. Выполнение заданий № 1 и № 2 (1 балл) – по возможности.
4. Домашняя работа – набор нескольких тек-

стов по образцу, работа с клавиатурными тренажерами.

Индивидуальная образовательная траектория этой группы будет выглядеть следующим образом (рис. 4):

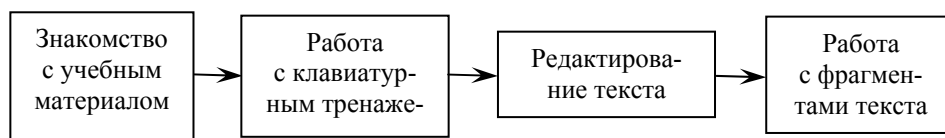


Рис. 4. Пример ИОТ

Изучение параграфа учитель фиксирует в сетевом документе (рис. 5). Выполнение каж-

дого задания отмечается в соответствии со шкалой оценивания в баллах.

Обработка текстовой информации							
Файл Правка Просмотр Вставка Формат Данные Инструменты Дополнения Справка Все изменения сохранены на Диске							
100% p. % .00 123 Arial 10 B I U A							
fx							
A	B	C	D	E	F	G	H
2		Набор текста (1 балл)	Устранение ошибок в готовом тексте (1 балл)	Работа с фрагментами текста (1 балл)	Работа с фрагментами текста (2 балла)	Сумма баллов	Итоговая оценка за урок
3	Алексеев Н.	1	1		2	4	отлично
4	Анкудинов С.	1				1	нужно еще поработать
5	Богатырев А.	1	1			2	удовлетворительно
6	Быков В.	1				1	нужно еще поработать
7	Воеводин М.	1	1	1		3	хорошо
8	Востриков А.	1	1			2	удовлетворительно
9	Гаврилоаа Н.	1	1		2	4	отлично
10	Головин Д.	1	1	1	2	5	отлично
11	Дикарев О.	1	1	1		3	хорошо
12	Ермакова О.	1	1			2	удовлетворительно
13	Елфимова Т.	1	1	1	2	5	отлично

Рис. 5. Пример фиксации результатов

В конце урока подводятся итоги выполнения заданий, которые фиксируются в двух ячейках таблицы: «Сумма баллов» и «Итоговая оценка за урок». В первой ячейке суммируются все баллы, набранные учеником за выполнение заданий. В ячейке итоговой оценки по критериям выставляется оценка: 4–5 баллов – «отлично»; 3 балла – «хорошо»; 2 балла – «удовлетворительно»; балл – не оцениваем.

Мы привели пример построения индивидуальных образовательных траекторий на материале первого урока по теме «Создание текстовых документов на компьютере» 7 класса.

Такой подход к освоению учебного материала позволит учителю повысить качество знаний каждого обучающегося, так как они движутся по собственным траекториям в соответствии со своим уровнем знаний, постепенно переходя на более высокий уровень, что соответствует требова-

ниям ФГОС в области использования информационно-коммуникационных технологий.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Загвязинский, В.И. Противоречия процесса обучения. – Свердловск, 1971.
2. Жукова, И.Г., Кульцова, М.Б., Аникин, А.В. Модели и инструментальные средства интеграции распределенных ресурсов в открытых образовательных сетях // Известия Волгоградского государственного технического университета. 2010. Т. 9. № 11 (71). С. 91–94.
3. Машевская, Ю.А. Теория и практика проектирования индивидуальных образовательных траекторий освоения информатических дисциплин будущими учителями : учеб.-метод. пособие [Электронный ресурс] / Ю.А. Машевская, Т.К. Смыковская, А. М. Коротков. – Волгоград: ВГСПУ, 2016. URL : <http://www.iprbookshop.ru/57787.html> (дата обращения: 02.05.2018).
4. Садовникова, Н.П. Особенности проектирования инструментальных средств поддержки учебного процесса для второго высшего образования // Актуальные вопросы профессионального образования. 2007. Т. 4. № 7 (33). С. 181–182.

АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ

УДК 004.42

В. А. Егунов, А. А. Князев

ПРОЕКТИРОВАНИЕ ОПТИМИЗИРОВАННОЙ АРХИТЕКТУРЫ WEB-КЛИЕНТА, ВЗАИМОДЕЙСТВУЮЩЕГО С REST API

Волгоградский государственный технический университет

vegunov@mail.ru, akanew@mail.ru

Рассматривается возможность повышения скорости загрузки web-страниц приложений, построенных на REST-архитектуре, сопоставимой аналогичным приложениям, осуществляющим прямые запросы к серверу. Предлагается вариант оптимизированного архитектурного решения для взаимодействия с WEB API через REST.

Ключевые слова: REST, WEB API, AngularJS, ReactJS, VueJS.

V. A. Egunov, A. A. Knyazev

DESIGNING OPTIMIZED ARCHITECTURE OF A WEB CLIENT THAT INTERACTS WITH THE REST API

Volgograd State Technical University

The article presents an analysis of variants developing applications on the REST architecture, which have a close speed of loading web-pages in comparison with applications built on the architecture that performing direct requests to the server. In addition, the article offers a variant of the implementation of an optimized application architecture that combines WEB API with REST.

Keywords: REST, WEB API, AngularJS, ReactJS, VueJS.

Web-среда является одним из наиболее популярных направлений разработки программных продуктов. Это область, которая имеет большую целевую аудиторию и определяет значительную долю современного рынка. Последние тенденции в развитии мобильных платформ и интернета серьезно изменили подход к web-разработке в связи с необходимостью охвата большого количества разнообразных устройств. Новым веянием стал отказ от MVC (Model-View-Controller) структуры в пользу альтернативных MV* архитектур [1, 2, 3]. Это повлияло на способ взаимодействия клиентской и серверной стороны web-приложения, приведя к появлению технологий WEB API и REST.

Архитектурный стиль REST предоставляет возможность создания кроссплатформенных, масштабируемых систем, гибких к изменению структуры и повышению производительности [4]. Он устанавливает характер взаимодействия

с WEB API посредством базовых операций (CRUD-create, read, update, delete) с использованием HTTP протокола. REST подход разделяет web-приложение на клиентскую и серверную часть, определяя их как обособленные архитектурные компоненты, связанные формой организации WEB API.

Архитектура клиента имеет здесь приоритетное значение, поскольку формат данных сервера строго формализован API и условно статичен, а форма их представления определяется логикой работы клиентской стороны.

Существующие решения для реализации web-клиента не нацелены на оптимизацию относительно применения REST. В них в большей степени упор сделан на «реактивность» интерфейса (быстрое изменение представления при изменении связанных с ним данных) и компонентный подход. Это определяет проблему увеличения времени загрузки web-страниц: получается, что приложение, построенное

на WEB API, обладает большим временем отклика на запрос в сравнении с приложением, производящим прямые запросы к серверу. В этом можно убедиться, проанализировав ре-

зультаты анализа среднего времени выполнения CRUD-запросов различных реализаций web-клиента, созданного на базе наиболее популярных фреймворков (рис. 1).

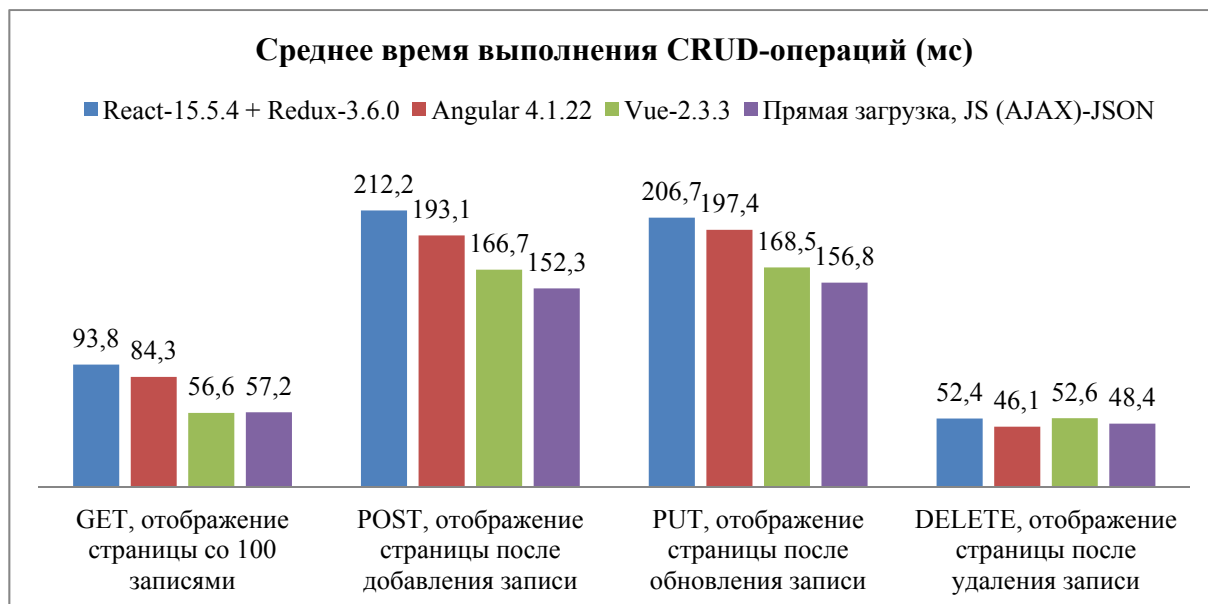


Рис. 1. Среднее время выполнения CRUD-операций в web-приложении (одностраничном TaskManager-e), реализованном посредством различных библиотек и фреймворков

В качестве решения данной проблемы был разработан концепт архитектуры компонентно-

го web-приложения с «реактивной» обработкой данных, совместимой с REST (рис. 2).

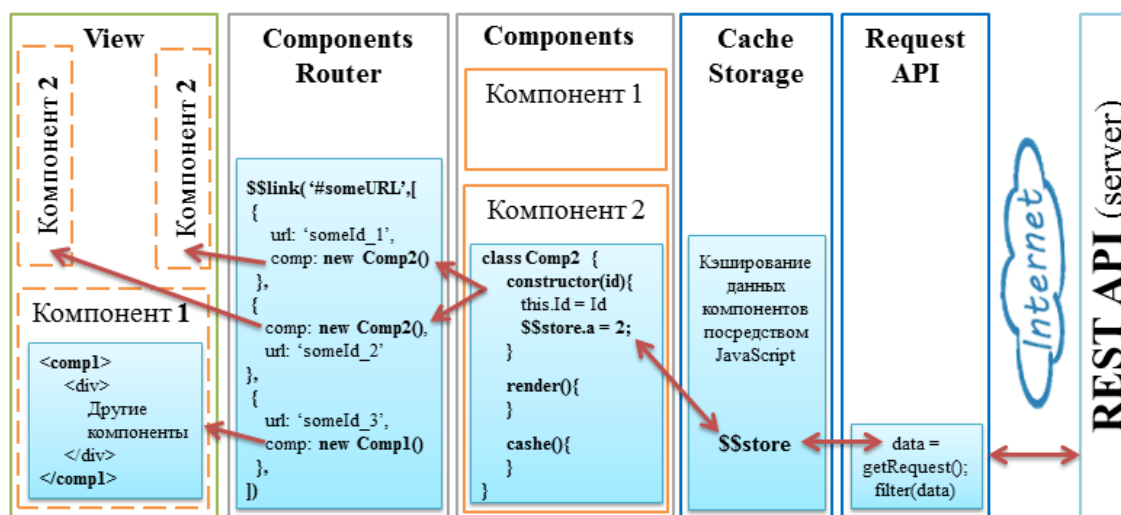


Рис. 2. Структура архитектурного решения web-клиента (стрелками указан вид взаимодействия блоков)

Основными структурными единицами архитектуры являются:

- блок представления (View);
- модуль управления переадресацией (ComponentsRouter);

– набор компонентов – объекты блока представления (Components);

– модуль управления изменением данных (CacheStorage);

– блок абстракции AJAX взаимодействия с WEB API (RequestAPI).

Архитектурное решение опирается на модель объектно-ориентированного программи-

рования, диаграмма классов прототипа архитектурного решения приведена на рис. 3.

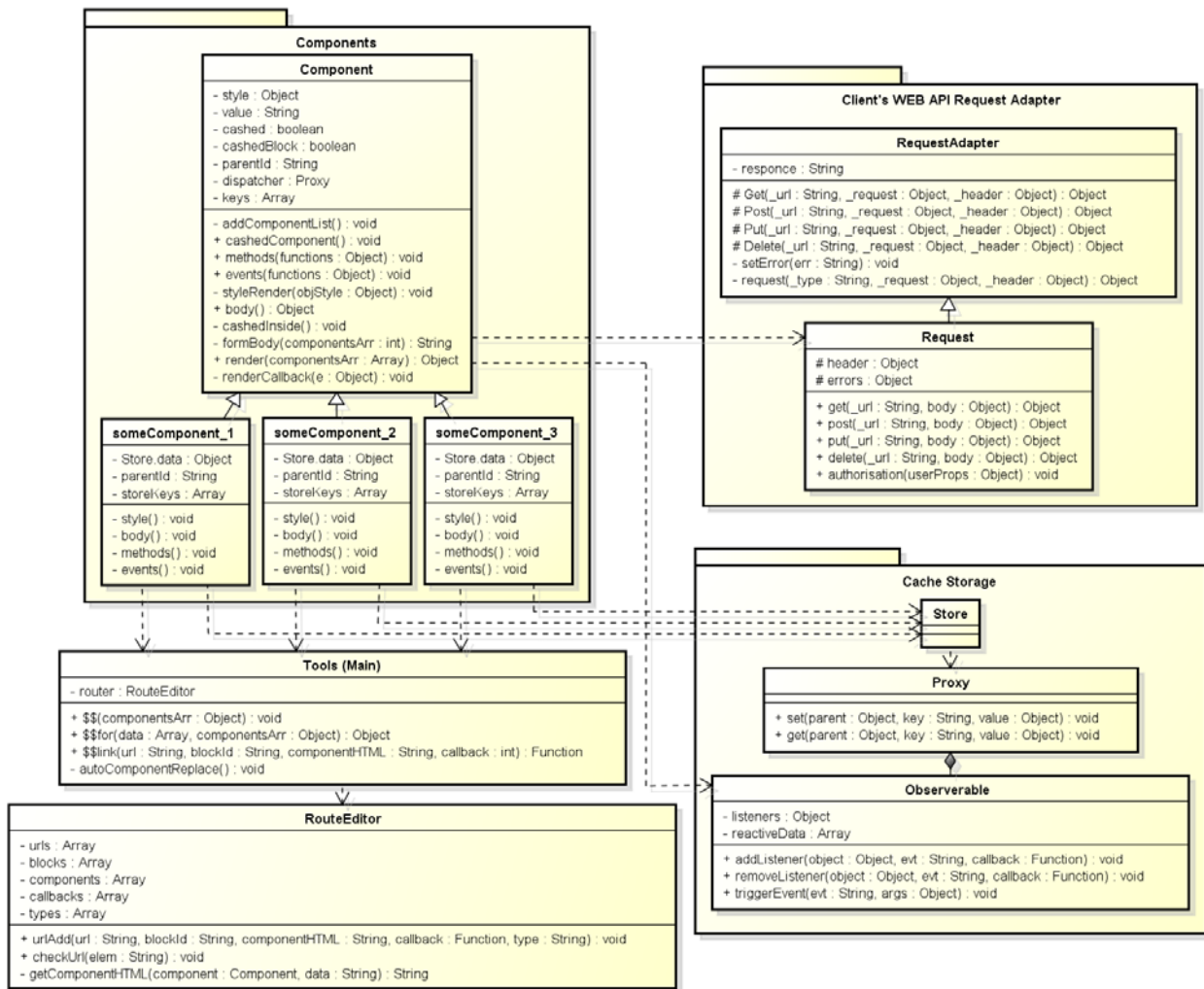


Рис. 3. Диаграмма классов прототипа архитектурного решения

Блок RequestAPI включает полиморфный класс с базовыми методами определения CRUD-операций, при этом осуществляется абстрагирование от специфики взаимодействия с WEB API. Протокол обмена данными с характерными особенностями процедур авторизации запросов, обработки ошибок (и т. д.) остается скрытым для последующих блоков архитектуры, представляя на выходе методов класса только конечные данные или сообщение об ошибке.

Модуль управления изменением данных (CacheStorage) реализуется в виде класса с методами обратного оповещения (callback-функциями) об изменении состояния данных. Это позволяет «реактивно» управлять значениями переменных компонентов при их модификации, как со стороны представления, так и со стороны модели.

Блок Components определяет множество компонентов представления, включая логику обработки данных и способ их отображения в интерфейсе пользователя. Все компоненты являются наследниками класса Component, который полиморфно предоставляет им набор методов для рендеринга представления компонента, хранения его данных, обработки событий внутренних элементов, реализации собственной логики и кэширования (частичного и полного). Фактически, каждый компонент является обособленной единицей страницы представления, обобщающей некоторое множество более простых элементов, используемых в совокупности, в рамках решаемой ими задачи.

Модуль управления переадресацией (ComponentsRouter) осуществляет отображение компонентов на странице интерфейса пользователя

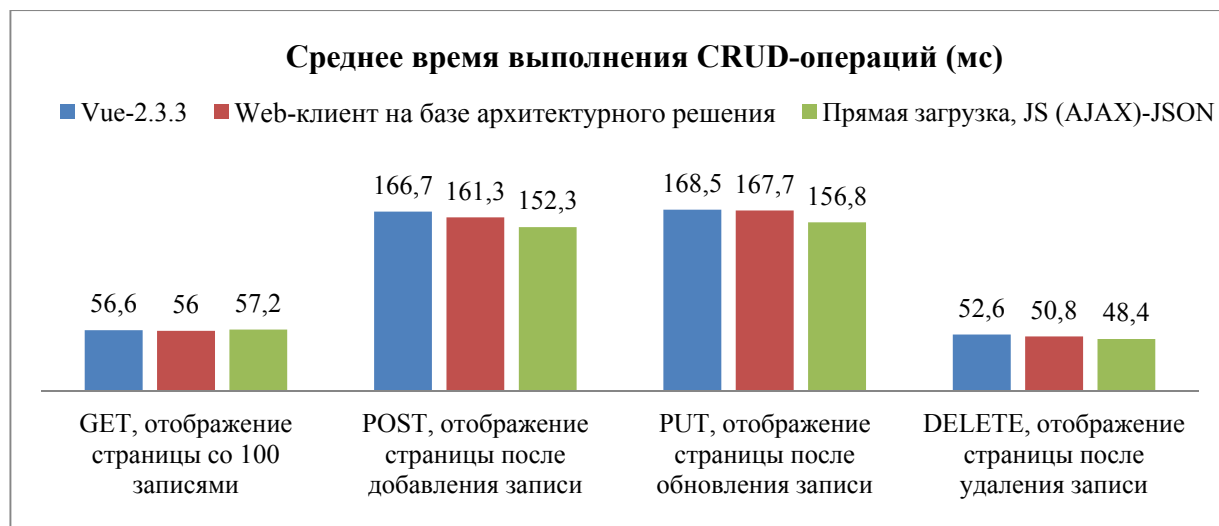


Рис. 5. Среднее время выполнения CRUD-операций в web-приложении (одностраничном TaskManager-e), реализованном на базе архитектурного решения

Таким образом, можно считать предложенное архитектурное решение вполне конкурентоспособным при использовании в приложениях с архитектурным стилем REST и WEBAPI, по отношению к популярным фреймворкам (при этом более оптимальным с точки зрения минимизации временных задержек при отклике на запрос пользователя).

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. React - A JavaScript library for building user interfaces [Электронный ресурс]. – 2018. – Режим доступа : reactjs.org
2. Angular. Tutorial: Tour of Heroes [Электронный ресурс]. – 2018. – Режим доступа : <https://angular.io/tutorial>
3. Vue.js – Прогрессивный JavaScript-фреймворк [Электронный ресурс]. – 2018. – Режим доступа : <https://ru.vuejs.org>
4. Leonard Richardson, Mike Amundsen, RESTful Web APIs / Simon St. Laurent, Meghan Blanchette. – Sebastopol: O'Reilly Media, 2013. – P. 94.

УДК 004.89

Д. М. Коробкин¹, С. А. Фоменков¹, С. Г. Колесников¹, Ф. А. Х. Аль-Хадша²

СИНТЕЗ И АНАЛИЗ ФИЗИЧЕСКИХ ПРИНЦИПОВ ДЕЙСТВИЯ ТЕХНИЧЕСКИХ СИСТЕМ С ИСПОЛЬЗОВАНИЕМ СЕТЕЙ ПЕТРИ*

¹ Волгоградский государственный технический университет

² University of Saba Region-Yemen-Marib City

dkorobkin80@mail.ru

В работе рассматривается синтез и анализ структур физического принципа действия технических систем на основе математического аппарата сетей Петри в программной среде CPN Tools.

Ключевые слова: физический принцип действия, техническая система, сеть Петри, CPN Tools.

D. M. Korobkin¹, S. A. Fomenkov¹, S. G. Kolesnikov¹, F. A. H. Al-Hadsha²

SYNTHESIS AND ANALYSIS OF THE PHYSICAL PRINCIPLES OF OPERATION OF TECHNICAL SYSTEMS BASED ON PETRI NETS

¹ Volgograd State Technical University

² University of Saba Region- Yemen-Marib City

The authors developed a synthesis and analysis of physical operation principle of technical systems based on Petri nets. Implemented a software of synthesis and analysis of POP structures in the software's environment CPN Tools.

Keywords: physical operation principle, technical system, Petri nets, CPN Tools.

© Коробкин Д. М., Фоменков С. А., Колесников С. Г., Аль-Хадша Ф. А. Х., 2018

* Работа выполнена при финансовой поддержке РФФИ (проекты № 16-07-00534-а, № 18-07-01086-а).

В настоящее время одной из важнейших задач является создание новых высокоэффективных технических систем (ТС). Известен ряд подходов к реализации начальных этапов проектирования ТС, среди них одним из наиболее перспективных является подход, связанный с привлечением структурированных физических знаний в форме физических эффектов (ФЭ) [1–9] для автоматизированного формирования и выбора физического принципа действия (ФПД) разрабатываемой ТС.

Анализ работ, связанных с использованием физических эффектов на начальных этапах проектирования ТС, показал ряд проблем, тормозящих эффективное и широкое внедрение данных методов [10–13]. Одной из таких проблем является представление результатов синтеза структур ФПД. Наиболее распространена выдача цепочек ФЭ в порядке их получения. Актуальной проблемой автоматизации выбора ФПД ТС является разработка методов оценки синтезируемых структур, то есть способов определения их физической реализуемости.

При помощи разработанной автоматизированной системы извлечения описаний физических эффектов из русско- и англоязычных текстов документов был произведен подсчет частот встречаемости описаний ФЭ в патентном массиве. В качестве веса ФЭ используется инверсная частота IDF , значение которого тем меньше, чем чаще описание ФЭ встречается в патентном массиве.

$$IDF = \log \frac{|D|}{|D_i|}, \quad (1)$$

где $|D|$ – число всех патентов в массиве; $|D_i|$ – число патентов, содержащих описание некоторого ФЭ t .

$$w(POP_k) = \sum_{i=1}^{N_k} IDF_i, \quad (2)$$

где w – вес структуры ФПД POP_k ; N_k – длина цепочки (структуры) ФПД POP_k ; IDF_i – обратная документная частота описания i -го ФЭ, участвующего в цепочке ФПД POP_k .

На основе значения веса ФЭ, отражающего его практическую реализуемость (встречаемости описания в патентном массиве), определяется оптимальная структура ФПД как частный случай решения проблемы нахождения кратчайшего пути на взвешенном графе.

Пусть имеется некоторое конечное множество физических эффектов $PhE = \{PhE_i\}$, $i = 1, N$. Сопоставим этому множеству ориентированный граф:

$G(V, E)$, где V – конечное непустое множество вершин графа (физических эффектов из БД); E – множество дуг графа.

Ребро e_{ij} существует тогда и только тогда, когда i -й ФЭ совместим с j -м ФЭ ($i, j = 1, N$). Вес ребра e_{ij} определяется весом j -го ФЭ. ФПД – это путь $v_1, e_{1,2}, v_2, e_{2,3}, \dots, e_{k-1,k}, v_k$ в графе G длины $k-1$ (налагаемое пользователем ограничение на длину искомого ФПД). Путь $v_1, v_2, v_3, \dots, v_k$ определяется заданием начального воздействия ФПД A_1 , которое эквивалентно входному воздействию эффекта v_1 , и результирующего воздействия ФПД C_k (выход эффекта v_k эквивалентен воздействию C_k). Задача оценки практической реализуемости синтезированных ФПД новых технических систем состоит в нахождении всех таких кратчайших путей в графе G , удовлетворяющих заданным ограничениям.

Для решения данной задачи было принято решение использовать теорию сетей Петри. Введем соответствие между разработанной моделью представления ФЭ и сформированным процессом синтеза ФПД с одной стороны и имитационным процессом, созданным на основе сетей Петри.

Процесс синтеза ФПД в виде цветных сетей Петри будет характеризоваться следующим:

множество маркеров представляют собой множество ФЭ;

множество позиций – база физических эффектов, входное и выходное воздействие ФПД согласно заданию на синтез, множество синтезированных ФПД;

условия срабатывания перехода – соответствие ФЭ входному и выходному воздействию ФПД согласно заданию на синтез, а также совпадение выходного воздействия одного и входного воздействия другого ФЭ из структуры ФПД.

Физический эффект представлен сложным составным типом POP , который состоит из входного и выходного воздействия (voz) (первое и четвертое поле маркера), качественных характеристик входных (второе поле маркера) и выходных (пятое поле маркера) воздействий ($char$), физических величин входа и выхода (третье и шестое поле маркера) ($fizvel$), номера физического эффекта ($numfe$) (седьмое поле маркера), веса (критерия практической реализуемости) физического эффекта (восьмое поле маркера) ($weight$). Список $plist$ (девятое поле маркера) используется для сохранения номеров ФЭ, составляющих цепочку (структуру) ФПД. На выходе имитационного процесса получаем множество маркеров, представляющих собой син-

тезированные структуры ФПД. Маркер содержит на каждом этапе моделирования информацию о текущем ФЭ (добавленном в ФПД), список всех используемых в данной структуре

ФПД физических эффектов, а также результирующий вес ФПД. Описания множеств цветов (colset) используемые в имитационной модели в среде CPN Tools, представлены на рис. 1.

```

▼ colset voz = with voz1|voz2|voz3|voz4|voz5|voz6|voz7|voz8|voz9|voz10|voz11|voz12|voz13;
▼ colset vrem = with vrem1|vrem2|vrem3|vrem4|vrem5|vrem6;
▼ colset pros = with pros1|pros2|pros3|pros4|pros5|pros6|pros7|pros8;
▼ colset spec = with spec1|spec2|spec3|spec4|spec5|spec6|spec7;
▼ colset fizvel = with fizvel1|fizvel2|fizvel3|fizvel4|fizvel5|fizvel6|fizvel7|fizvel8|fizvel9|fizvel10|fizvel11|fizvel12;
▼ colset char = product vrem * pros * spec;
▼ colset numfe = INT;
▼ colset plist = list INT;
▼ colset weight = REAL;
▼ colset POP = product voz*char*fizvel*voz*char*fizvel*numfe*weight*plist;
▼ colset PhE = product voz*char*fizvel;
▼ var count:INT;
▼ var x:POP;
▼ var y:POP;
▼ var z:PhE;

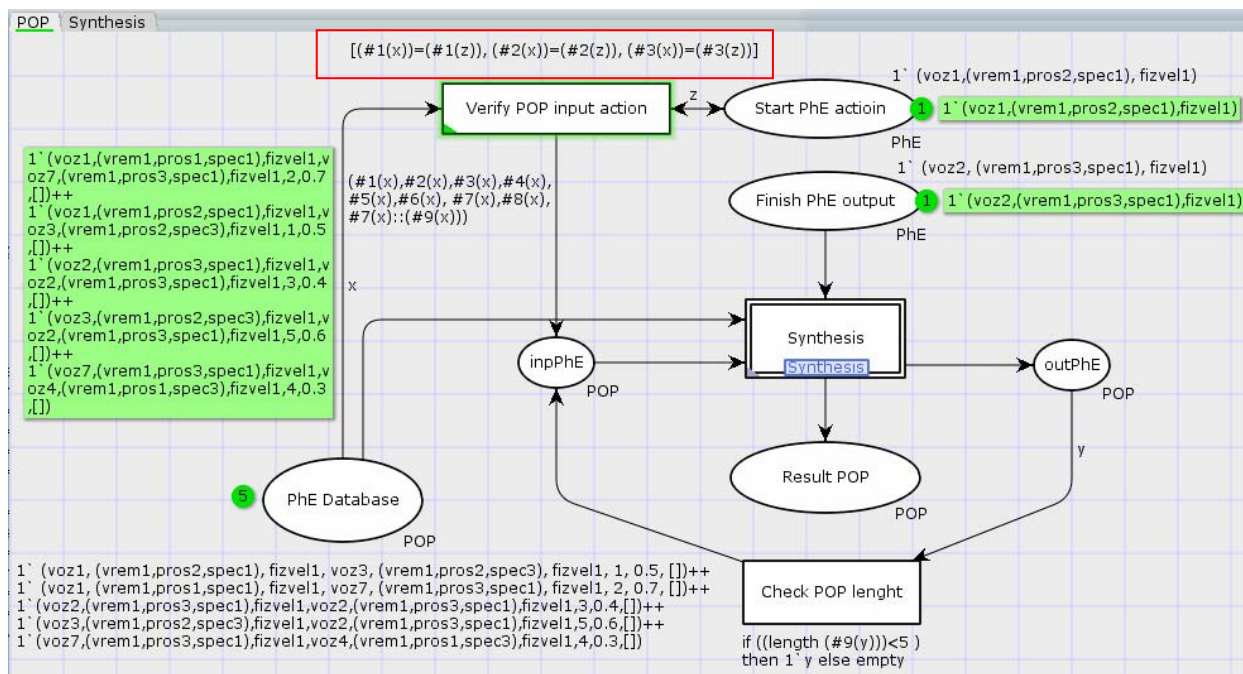
```

Рис. 1. Описания множеств цветов

Иерархическая модель синтеза ФПД на основе сетей Петри приведена на рис. 2.

Элементами модели являются подмодель «Synthesis»; позиция «PhE Database» – БД ФЭ, «Start PhE action» – задание на синтез (входное воздействие), «Finish PhE action» – задание на синтез (выходное воздействие), «Result POP» –

ФПД, полученные в результате синтеза; переход «Verify POP input action» – отбор из БД ФЭ маркеров (ФЭ), имеющих входное воздействие, совпадающее с заданием на синтез, «Check POP length» – проверка условия останова процесса синтеза (длина цепочки ФПД).



Задание на синтез:

Вход: электрическое поле (voz1), постоянное (vrem1), однородное (pros1), слабое (spec1), напряженность электрического поля (fizvel1). **Выход:** электрический ток (voz2), постоянное (vrem1), однородное (pros1), электронный (spec1), сила электрического тока (fizvel1). **Длина пути:** 5.

Рис. 2. Проверка на входное воздействие ФПД

В процессе синтеза требуется получить цепочку ФПД с заданным входным и выходным воздействиями и не больше заданной длины. В позиции «PhE Database» находятся маркеры, представляющие собой все множество ФЭ. Переход «Verify POP input action» активен, если входное воздействие маркера (первое, второе и третье поле) из позиции «PhE Database» соответствует заданному входному воздействию синтезируемой ФПД. Используется подстановка

перехода «Synthesis» на сеть более низкого уровня иерархии «Synthesis» (рис. 3). В позицию «PhE» через переход «Verify input & output PhE action» из всего множества маркеров позиции «PhE Database» переходят только те фишки, для каждого из которых выполняется условие: входное воздействие (первое, второе и третье поле маркера) переходящей фишки должно быть равно выходному воздействию (четвертое, пятое и шестое поле) маркера из позиции «inpPhE».

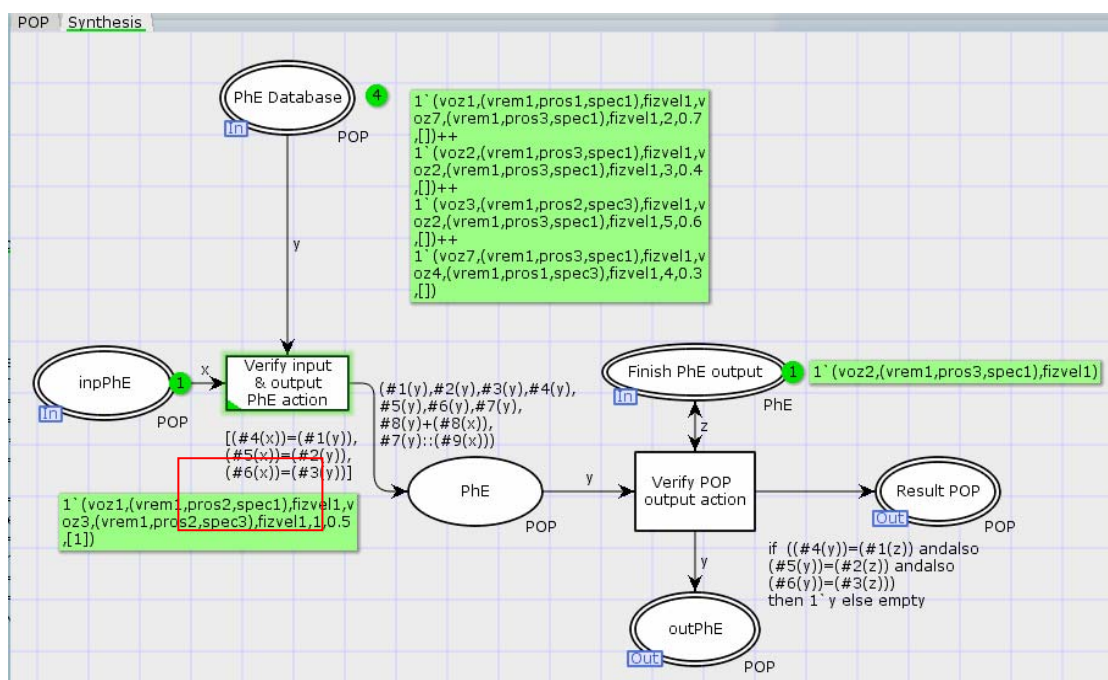


Рис. 3. Проверка на совместимость ФЭ

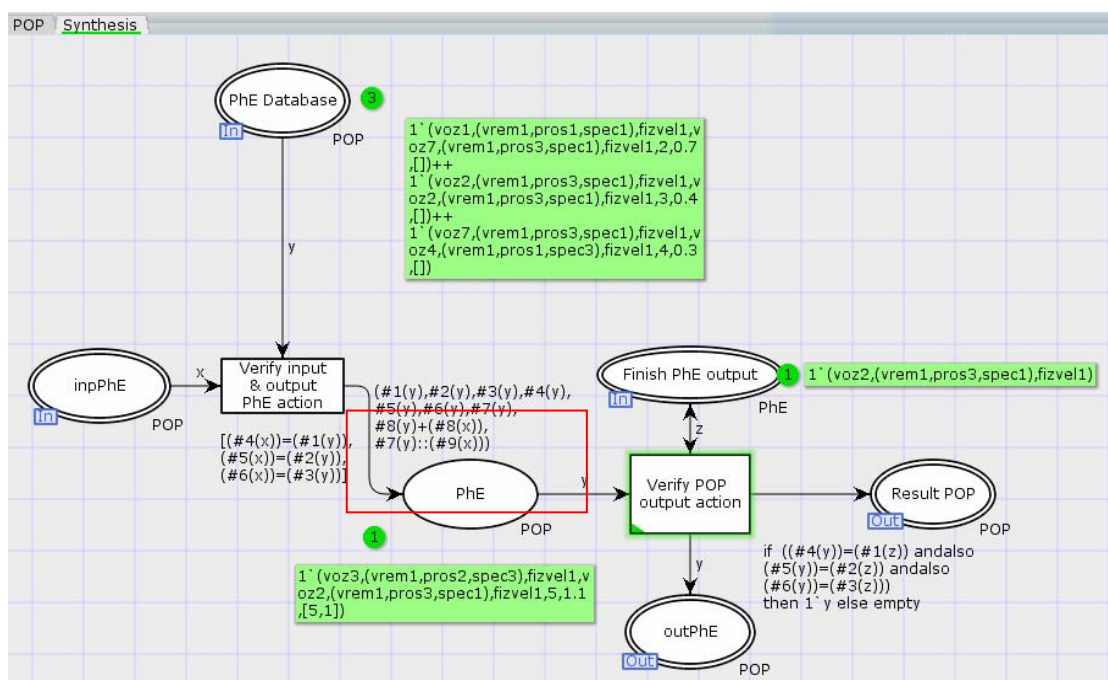


Рис. 4. Порождение цепочки ФПД

На выходной дуге перехода «Verify input & output PhE action» происходит порождение новых маркеров (рис. 4). Заменяются первое – седьмое поля маркера на значения совместимого ФЭ из БД, т. е. теперь он представляет собой

следующий ФЭ в цепочке ФПД. Цепочка ФПД до текущего элемента хранится в поле «plist» маркера. Вес всех ФЭ в ФПД суммируется и сохраняется в поле «weight».

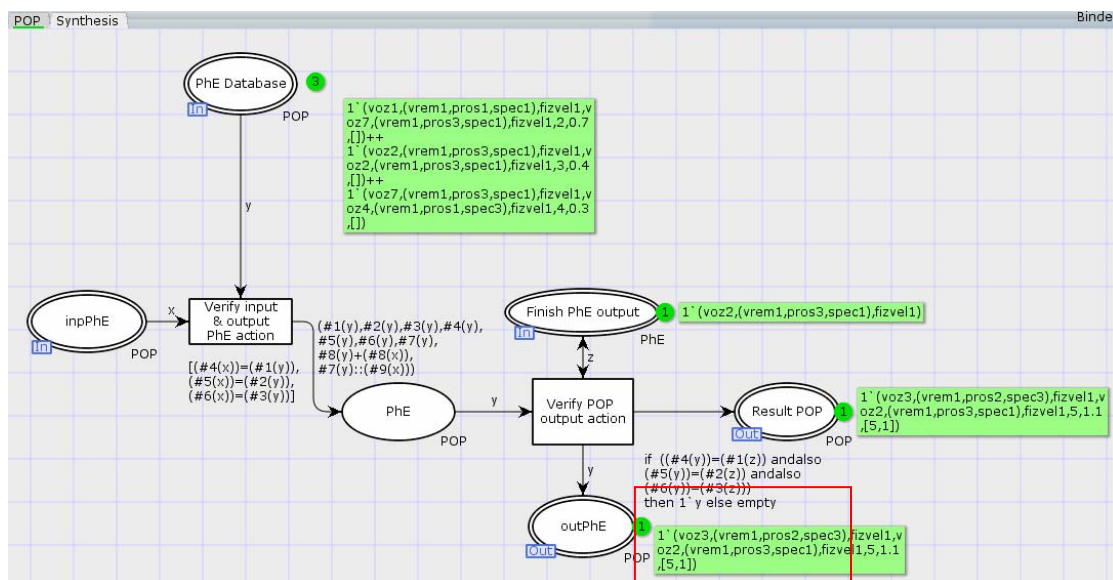


Рис. 5. Проверка на выходное воздействие ФПД

На выходной дуге перехода «Verify POP output action» осуществляется проверка условия выхода из процесса синтеза ФПД. В позицию «Result POP» (рис. 5) попадают маркеры, выходное воздействие (четвертое, пятое и шестое поля) которых соответствует выходному воздействию искомой цепочки ФПД (задание на синтез). Данный маркер представляет собой последний ФЭ цепочки ФПД, в первом – седьмом полях такой фишки содержится информация о последнем завершающем ФЭ, в поле «weight» находится значение веса всей цепочки ФПД, в поле «plist» перечислены номера всех ФЭ данной цепочки ФПД в обратном порядке. В позицию «outPhE» помещаются все маркеры, удовлетворяющие и не удовлетворяющие данному условию (т. е. не достроенные структуры ФПД), и переходят в позицию «inpPhE» для продолжения синтеза.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Фоменков С.А., Давыдов Д.А., Колесников С.Г. Автоматизированный информационный поиск физических эффектов // Информационные технологии. 2004. № 7. С. 30–34.
2. Фоменков С.А., Колесников С.Г. Представление физических знаний в автоматизированном банке физических эффектов // Известия вузов. Машиностроение. 1998. № 1–3. С. 55–61.
3. Фоменков С.А., Колесников С.Г. Информационное наполнение баз данных по физическим эффектам // Информационные технологии. 2004. № 6. С. 60–62.
4. Fomenkov S., Korobkin D., Kolesnikov S. Method of Ontology-Based Extraction of Physical Effect Description from Russian Text // Communications in Computer and Information Science. 2014. Т. 466 CCIS. С. 321–330.
5. Фоменков С.А., Колесников С.Г., Коробкин Д.М. Методика модификации информационного обеспечения базы данных физических эффектов // Известия Волгоградского государственного технического университета. 2012. Т. 10. № 14. С. 166–170.
6. Фоменков С.А., Колесников С.Г., Деврянкин А.М. Использование структурированных физических знаний для прогнозирования новых нанотехнических систем // Известия Волгоградского государственного технического университета. 2012. Т. 4. № 13. С. 80–82.
7. Коробкин Д.М., Фоменков С.А., Колесников С.Г. Автоматизация процесса формирования информационного обеспечения базы данных физических эффектов // Вестник компьютерных и информационных технологий. 2005. № 3. С. 22–25.
8. Петрухин А.В., Фоменков С.А., Колесников С.Г. Архитектура автоматизированной системы концептуального проектирования технических объектов и технологий с использованием структурированного описания физической информации (СОФИ) // Известия высших учебных заведений. Машиностроение. 1998. № 4–6. С. 52–56.
9. Korobkin D.M., Fomenkov S.A., Kolesnikov S.G., Voronin Y.F. System of Physical Effects Extraction from Natural Language Text in the Internet // World Applied Sciences Journal (WASJ). 2013. Т. 24. № 24. С. 55–61.

10. Korobkin D.M., Fomenkov S.A., Kolesnikov S.G., Orlova Y.A. A Multi-Stage Algorithm for Text Documents Filtering Based on Physical Knowledge // WORLD APPLIED SCIENCES JOURNAL. 2013. T. 24. № 24. С. 91–97.

11. Fomenkov S.A., Korobkin D.M., Kolesnikov S.G., Dvoryankin A.M., Kamaev V.F. Procedure of Integration of the Systems of Representation and Application of the Structured Physical Knowledge. // Research Journal of Applied Sciences. 2014. T. 9. № 10. С. 700–703.

12. Korobkin D.M., Fomenkov S.A., Kolesnikov S.G. Ontology-based extraction of physical effect description from

russian text // В сборнике: Proceedings of the European Conference on Data Mining 2014 and International Conferences on Intelligent Systems and Agents 2014 and Theory and Practice in Modern Computing 2014 - Part of the Multi Conference on Computer Science and Information Systems, MCCSIS 2014. С. 260–262.

13. Korobkin D.M., Fomenkov S.A., Kolesnikov S.G. Semantic network of physical effects descriptions in natural language context // В сборнике: Proceedings of the IADIS International Conference WWW/Internet 2013, ICWI 2013. С. 342–346.

УДК 004.5

А. С. Поцелуйко, М. Б. Кульцова, А. М. Дворянкин

РЕДАКТОР ДЛЯ СОЗДАНИЯ АДАПТИВНЫХ ИНТЕРФЕЙСОВ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ ДЛЯ ЛЮДЕЙ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ

Волгоградский государственный технический университет

poas@vstu.ru

В статье описано программное средство – редактор интерфейсов, предназначенный для создания адаптивных интерфейсов мобильных приложений для людей с ограниченными возможностями, в частности, архитектура программного средства, его основные компоненты, экранные формы и процессы взаимодействия пользователя и приложения. Кроме этого, представлены результаты тестирования редактора в группах разработчиков интерфейсов, а также тестирования разработанных в редакторе мобильных приложений в группах пользователей с ограниченными возможностями.

Ключевые слова: вспомогательные компьютерные технологии, пользователи с ограниченными возможностями, адаптивный интерфейс пользователя, онтологическая модель, паттерны интерфейса, процесс адаптации интерфейса, механизм рассуждений, двухэтапный алгоритм адаптации интерфейса.

A. S. Potseluko, M. B. Kultsova, A. M. Dvoryankin

THE SOFTWARE TOOL FOR DEVELOPMENT OF ADAPTIVE MOBILE USER INTERFACES FOR PEOPLE WITH SPECIAL NEEDS

Volgograd State Technical University

This paper presents a software tool – interface editor, designed for development of adaptive mobile interfaces for users with special needs, namely, the software architecture, its main components, screen forms and interaction processes between user and application. Also, the results of testing the interface editor in groups of interface developers and people with special needs are presented, as well as the results of testing the mobile applications developed in the editor in groups of users with special needs.

Keywords: assistive technologies, users with special needs, adaptive user interface, ontological model, interface patterns, user interface adaptation, ontological reasoning, two-phase algorithm of interface adaptation.

Введение

В настоящее время разработано огромное число мобильных приложений, призванных облегчить повседневную жизнь человека. Зачастую такие приложения могут быть полезны и для людей с ограниченными возможностями, но в силу физических или ментальных особенностей таких людей использование ими мобильных приложений, не спроектированных должным образом, не представляется возможным. Чтобы уменьшить социальные барьеры для

данной группы людей, разработчики программного обеспечения создают так называемые ассистивные (вспомогательные) технологии. Согласно отчету Всемирной Организации Здравоохранения (ВОЗ) ассистивные технологии – это вспомогательные средства и услуги, предназначенные для поддержания на прежнем уровне или повышения функциональных возможностей и автономности людей с ограниченными возможностями, тем самым способствуя их благополучию. Ассистивные техноло-

гии снижают потребность в формальных услугах здравоохранения и поддержке, долгосрочном уходе и услугах опекунов [1].

Рекомендации для разработчиков интерфейсов, описанные в [2], предполагают постоянный контакт разработчика с конечным пользователем в процессе создания интерфейса, в противном случае разработка интерфейса может провалиться по ряду причин, среди которых:

- проблема сбора информации о заболеваниях пользователей и их последствиях;
- проблема понимания ожиданий конечных пользователей;
- проблема настроек элементов интерфейса;
- проблема понимания важности персонализации интерфейса для пользователя с ограниченными возможностями.

Чтобы избежать данных проблем, в ходе исследования применялась технология User-CenteredDesign [3], что обеспечило непосредственное участие целевой аудитории, а именно разработчиков интерфейсов и людей с ограниченными возможностями. Описание подхода к адаптации, описание онтологической модели, механизма рассуждения, модели паттерна были представлены в более ранних работах [7, 8, 11], поэтому далее будут приведены архитектурные решения и результаты тестирования разработанного программного средства.

1. Описание предметной области разработанной системы

В процессе дизайна интерфейса для людей с ограниченными возможностями разработчик

проходит несколько итераций. Обязательным требованием является соответствие разрабатываемого интерфейса критериям качества, представленным в стандартах, например, в [3]. В данном случае разработчик вынужден затрачивать большое количество времени на доработку имеющегося интерфейса или на создание нового, подходящего для конкретного типа пользователей. С применением автоматизированного средства для адаптации интерфейса часть операций, связанных с доработкой и изменением интерфейса, будет осуществляться автоматически, в частности, поиск проблемных элементов интерфейса для конкретного пользователя с ограничениями и генерация рекомендаций по соответствующим интерфейсным решениям. Это позволит сократить время на создание и доработку интерфейса для людей с ограниченными возможностями, а также снизить трудоемкость создания таких интерфейсов.

Концепция создания адаптивных интерфейсов для людей с ограниченными возможностями представлена на рис. 1. Данная концепция решает проблему повышения эффективности создания адаптивных интерфейсов за счет использования онтологической модели представления знаний и логического вывода на онтологии [8]. Концепция также решает проблему персонификации интерфейсов для людей с ограниченными возможностями за счет применения двухэтапного метода адаптации интерфейса на основе применения паттернов интерфейса [7].

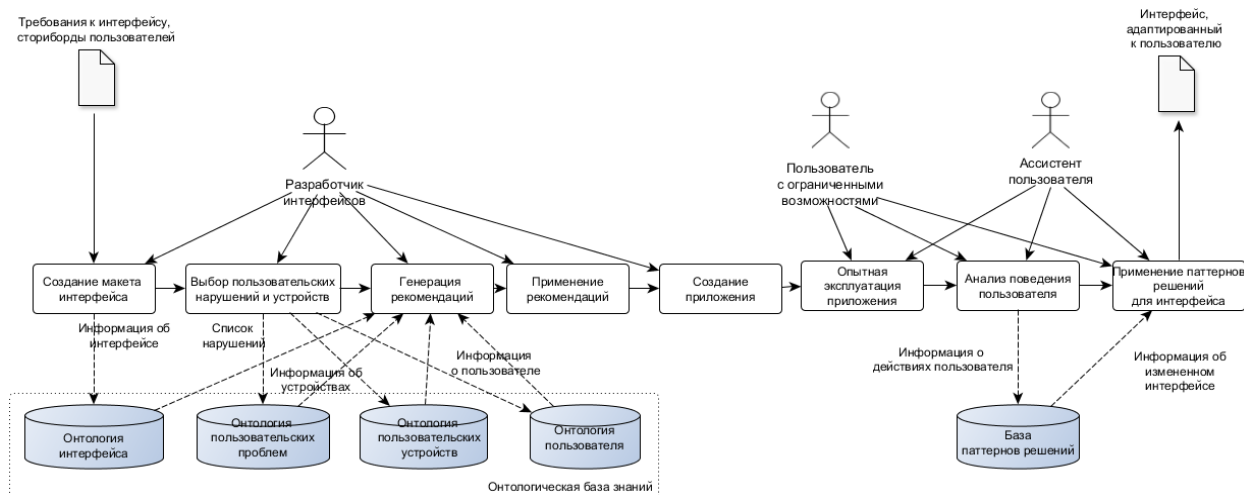


Рис. 1. Концепция создания адаптивных интерфейсов для пользователей с ограниченными возможностями

В ходе разработки концепции автоматизированного создания адаптивных интерфейсов для людей с ограниченными возможностями,

были выдвинуты функциональные требования к разрабатываемой системе:

- система должна предоставлять возмож-

ность разработчику добавлять, удалять и редактировать элементы интерфейса;

- система должна сохранять измененные свойства элементов интерфейса;

- система должна позволять разработчику выбрать проблемы пользователя, вызванные его заболеванием;

- система должна генерировать рекомендации по улучшению разрабатываемого интерфейса для пользователя с определенными нарушениями;

- система должна отобразить разработчику найденные рекомендации к интерфейсу;

- система должна дать возможность разработчику применить рекомендацию или отклонить ее;

- система должна добавлять возможность менять настройки для разработанных интерфейсов.

Также были определены некоторые ограничения для разрабатываемой системы:

- хранение информации об интерфейсе и пользователе должно осуществляться в соот-

ветствующих онтологических моделях предметной области;

- вывод рекомендаций для разработчика должен осуществляться с помощью онтологий с применением логического вывода на правилах;

- изменение настроек интерфейсов готовых приложений должно осуществляться с помощью базы паттернов интерфейса.

2. Архитектура программного средства

В ходе проектирования редактора интерфейсов были выявлены некоторые ограничения, в частности, невозможность использования библиотеки для работы с онтологиями для версии .NET ниже, чем 4. Поэтому было принято решение о создании отдельного приложения для работы с онтологиями, которое путем обмена данными с редактором интерфейса будет передавать список рекомендаций. Полная диаграмма компонентов разрабатываемой системы представлена на рис. 2.

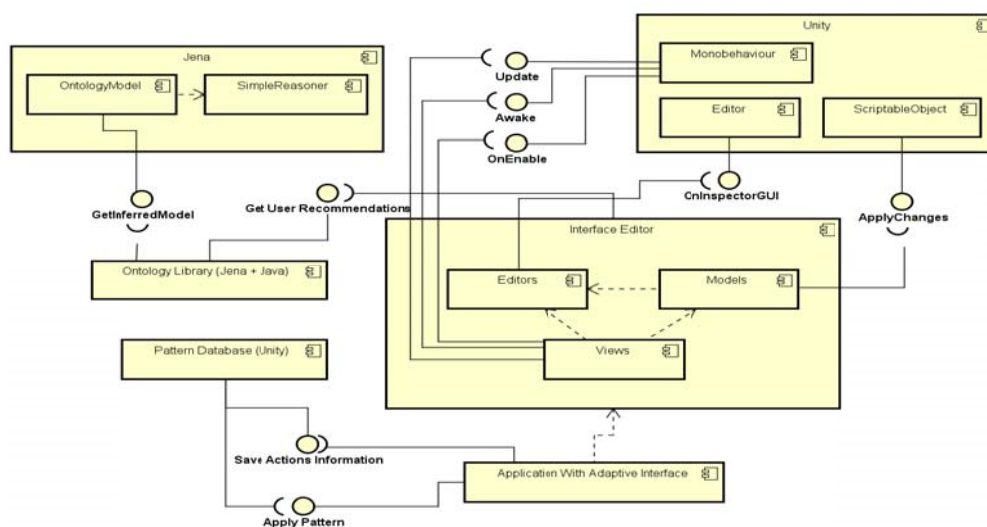


Рис. 2. Диаграмма компонентов разработанной системы

Главным компонентом является **Interface Editor** – редактор интерфейсов, который создается в среде **Unity3D** с использованием классов **Unity3D**. Редакторы свойств для каждого виджета, содержащиеся в компоненте **Editors**, должны реализовать интерфейс **OnInspector GUI** для создания собственных окон редактирования свойств виджетов, содержащихся в компоненте **Models**. Модели виджетов должны использовать интерфейс класса **ScriptableObject** для сохранения текущих параметров. Представления виджетов должны реализовывать интерфейсы

Update, **OnEnable** и **Awake** для инициализации, обновления и реакции на нажатие пользователя на представление виджета.

Рекомендации для пользователя генерируются с помощью обращения редактора интерфейса к библиотеке для работы с онтологиями. Данная библиотека разработана на языке **Java** и использует фреймворк **Jena**[5] для обработки моделей и получения новых знаний с помощью логического вывода на правилах.

Более детально архитектура редактора интерфейсов представлена на рис. 3. Главный

класс программы (InterfaceEditor) позволяет создать профиль пользователя с определенным набором заболеваний. У пользователя имеется интерфейс, представленный полем mainWin-

dow – главным окном интерфейса. Главное окно содержит в себе все виджеты, которые добавляются на сцену с помощью редактора интерфейсов.

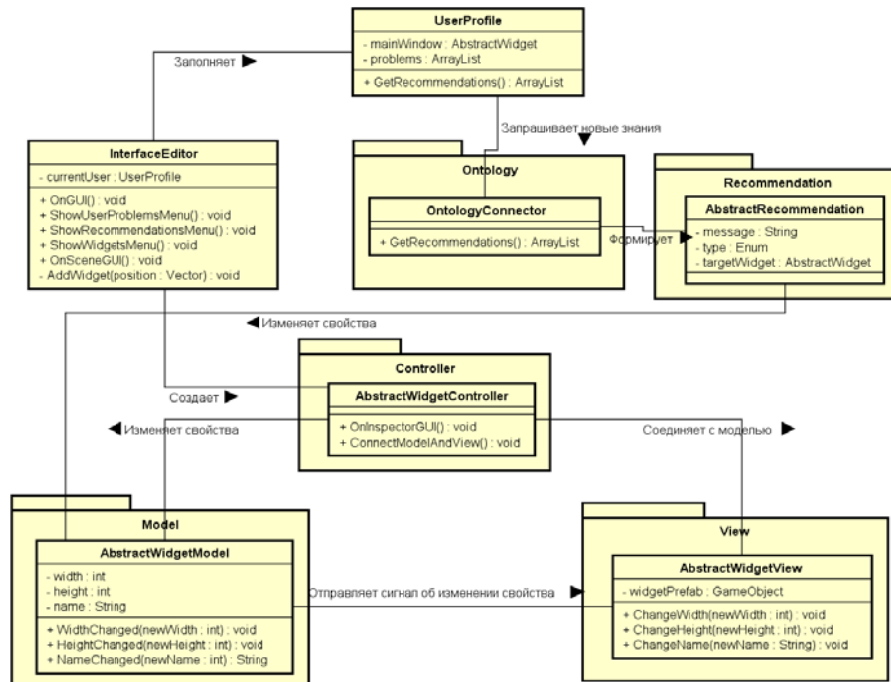


Рис. 3. Архитектура редактора интерфейса

Класс Ontology Connector позволяет связаться через редактор интерфейсов с онтологической моделью пользователя и получить список рекомендаций. Рекомендация содержит в себе сообщение по улучшению виджета, тип рекомендации и ссылку на виджет, к которому она относится.

Один виджет представлен тремя классами, согласно паттерну MVC: модель, представление и контроллер. Контроллер генерирует меню для редактирования свойств виджета, после чего изменения направляются в модель, а модель испускает события о том, что ее свойства изменились. После этого представление ловит события от модели и перерисовывается.

Вывод новых знаний на онтологии производится путем вызова отдельного приложения, написанного на языке Java. Основным классом для работы с онтологиями является Ontology Connector. Данный класс считывает входные файлы, содержащие информацию о текущем интерфейсе и пользовательских заболеваниях, а также принимает на вход файл с правилами для вывода на онтологии. Основная функция данного класса – получить и распределить имеющиеся виджеты по трем группам: разрешен-

ные, запрещенные и требующие обновления. Также данный класс занимается определением запрещенных и разрешенных типов компонентов интерфейса, которые можно или нельзя использовать для пользователя с заявленными проблемами.

3. Процесс адаптации интерфейса

Процесс получения рекомендаций для пользователя представлен на рис. 4.

Класс Ontology Connector создается с помощью главного класса приложения для работы с онтологией, само приложение запускается как отдельный процесс через редактор интерфейсов. Сначала класс Ontology Model создает модель предметной области с помощью считывания онтологий из файлов, получения информации о пользователе и текущем интерфейсе из файлов, сгенерированных редактором интерфейсов. Для выводов на модели подгружаются два файла с правилами: в первом файле правила для выявления разрешенных и запрещенных типов элементов интерфейса, а во втором правила для выявления разрешенных, запрещенных или требующих обновления текущих элементов интерфейса.

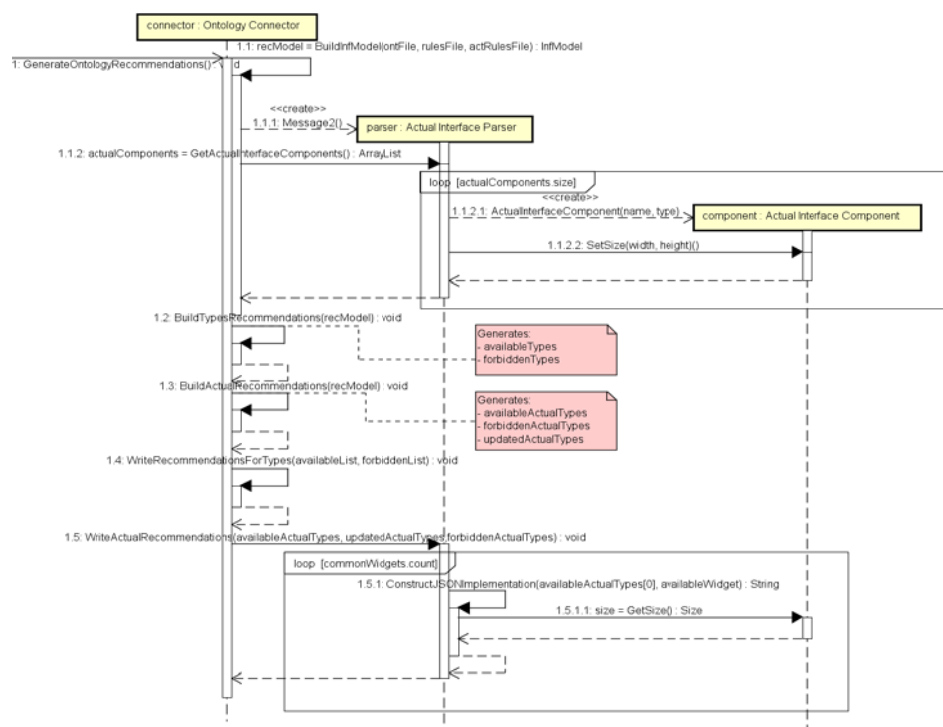


Рис. 4. Процесс получения рекомендаций

Построение модели вызывает создание класса Actual Interface Parser, который считывает из файла все элементы текущего интерфейса и сохраняет их в список, состоящий из элементов типа Actual Interface Component, которые легче записать в онтологию.

Далее с помощью правил и заполненных онтологических моделей класс Ontology Connector запускает процесс получения новых знаний, затем выявленные знания записываются в два файла: для типов элементов интерфейса и для текущих элементов интерфейса.

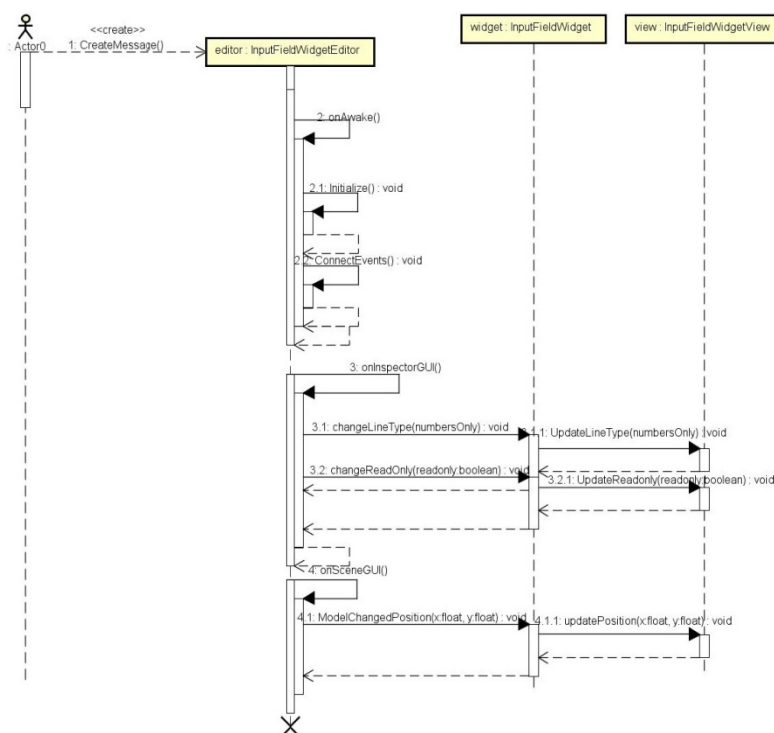


Рис. 5. Редактирование свойств виджета

Самым часто используемым действием с редактором является добавление виджетов на сцену и редактирование свойств добавленных виджетов. Диаграмма последовательности для отображения процесса редактирования виджета представлена на рис. 5.

Сначала разработчик выбирает виджет на сцене, чем инициирует создание объекта editor. Далее editor получает модель и представление и связывает их сигналами, затем генерирует интерфейс для редактирования свойств виджетов с помощью метода OnInspector GUI.

Разработчик редактирует свойство виджета, например, тип текста и флаг только на чтение.

Изменение полей в редакторе инициирует контроллер вызвать события в модели об изменении ее параметров. Модель генерирует события, представление их обрабатывает и меняет параметры у игрового объекта. После того, как разработчик снимает фокус с виджета, объект editor уничтожается, удалив связь между моделью и представлением.

По окончании разработки интерфейса необходимо получить рекомендации относительно правильности выбранных компонентов интерфейса. Процесс получения рекомендаций представлен на рис. 6.

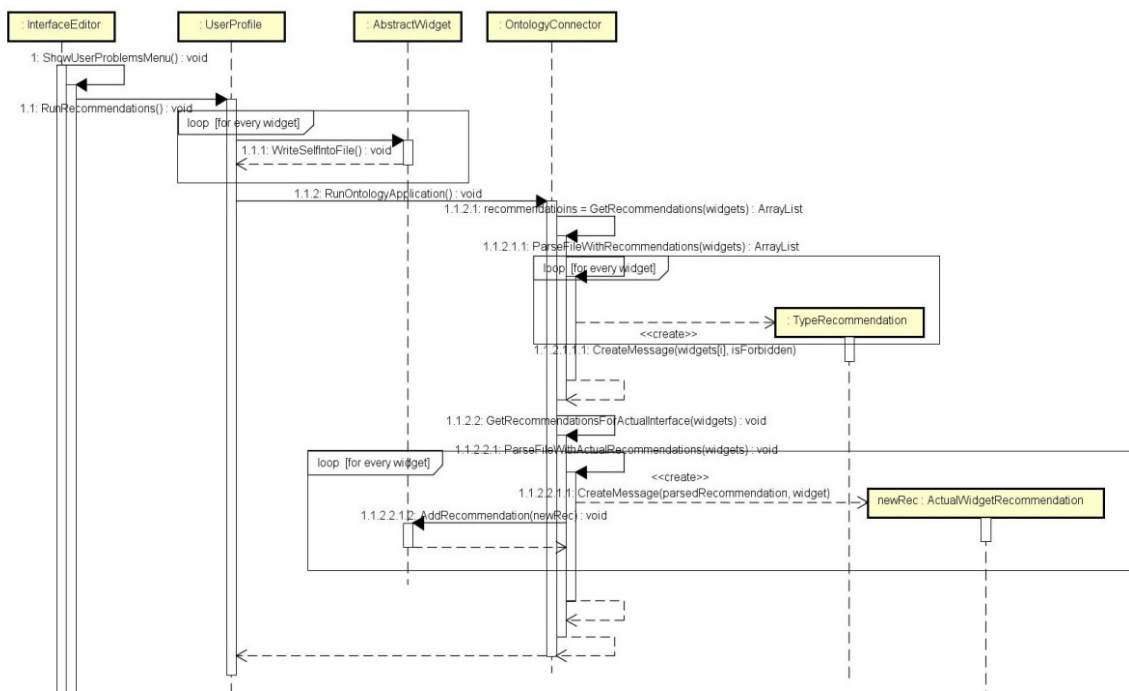


Рис. 6. Получение рекомендаций

На первом этапе разработчик выбирает ограничения пользователя, затем класс User Profile вызывает метод записи в файл для каждого виджета. Далее после формирования входных файлов для приложения класс Ontology Connector вызывает метод Run Ontology, который создает новый процесс для запуска приложения на языке Java. Этот же класс вызывает методы получения рекомендаций, которые вызываются после окончания работы приложения и заключаются в том, чтобы считать файлы с рекомендациями и добавить рекомендации к виджетам, к которым они относятся.

4. Пример работы системы

Чтобы создать новый интерфейс в редакторе, разработчик должен выбрать вкладку Interface Editor на панели инструментов Unity 3D. После этого появится экранная форма, позволяющая выбрать необходимые виджеты, а также изменить их свойства. Экранные формы представлены на рис. 7.

Далее разработчик выбирает ограничения пользователя, для которого создается интерфейс, и получает разрешенные и запрещенные для использования виджеты, а также список рекомендаций для виджетов, которые уже были добавлены к интерфейсу (см. рис. 8).

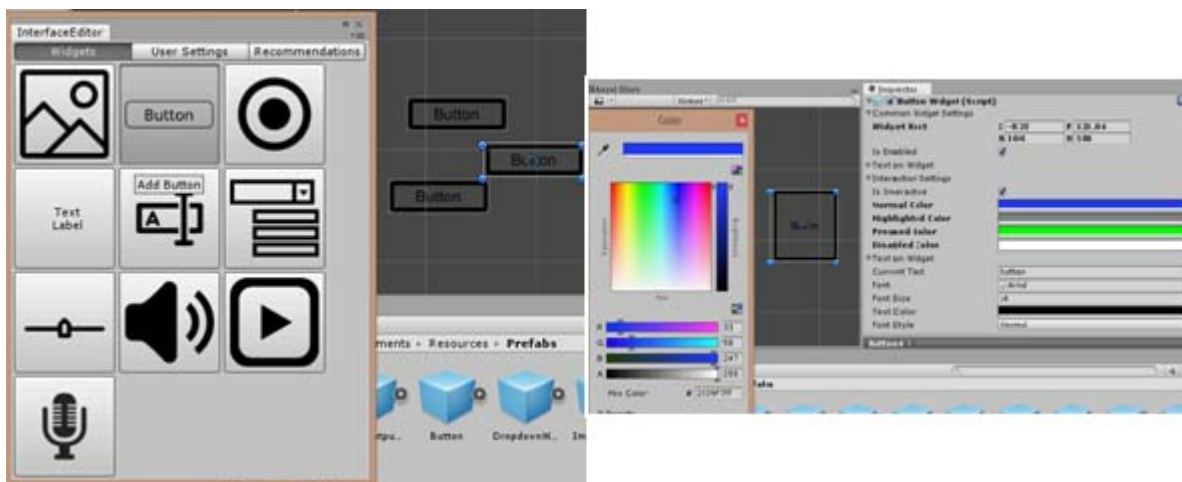


Рис. 7. Формы выбора виджетов и изменения их свойств



Рис. 8. Формы с результатом получения рекомендаций

С помощью кнопок Accept, Reject, Delete разработчик может принять рекомендацию и сменить цвет рамки виджета на зеленый, проигнорировать рекомендацию или удалить виджет. Зеленый цвет соответствует разрешенному виджету; желтый – виджету, требующему изменения свойств, а красный – виджету, который не может быть применен для пользователя с данными заболеваниями.

5. Результаты тестирования программного средства

Тестирование разработанной системы проводилось в три этапа: тестирование с помощью разработки программного кода тестов на пра-

вильность выбора интерфейсных решений, тестирование в группах разработчиков интерфейсов и тестирование разработанных в редакторе интерфейсов в группах пользователей с ограниченными возможностями. Ниже подробно описаны результаты двух последних этапов тестирования.

Для оценки качества программного продукта были использованы следующие критерии:

(K1) – смог ли разработчик спроектировать предоставленный вариант интерфейса;

(K2) – возникали ли трудности в процессе создания интерфейса;

(K3) – возникали ли трудности при выборе нарушений пользователя;

(K4) – возникали ли трудности при изменении свойств компонентов интерфейса;

(K5) – получилось ли сгенерировать рекомендации для спроектированного интерфейса;

(K6) – были ли рекомендации полезными;

(K7) – была ли система удобна в использовании.

Таблица 1

**Результаты тестирования приложения
в группе разработчиков**

Критерий	Выполнен	Не выполнен
K1	8	0
K2	5	3
K3	3	5
K4	8	0
K5	6	2
K6	8	0
K7	4	4

Результаты тестирования представлены в табл. 1; цифры в ячейках таблицы означают соответствующее количество тестируемых разработчиков.

Таким образом, тестирование редактора интерфейсов в группе разработчиков показало, что при некоторой дальнейшей доработке и улучшении системы можно получить полезный плагин для разработки адаптивных интерфейсов.

Последний этап тестирования проводился в Бельгии в MPI Oosterlo, в нем участвовали 5 человек с ограниченными возможностями. Специально для них было разработано приложение для определения профессиональных предпочтений человека (главное окно приложения представлено на рис. 9). Сначала эксперимент проводился с приложением без адаптации интерфейса, а затем с использованием адаптивных интерфейсов, созданных в разработанном редакторе интерфейсов.



Рис. 9. Главное окно приложения для выявления профессиональных предпочтений

Характеристики людей с ментальными нарушениями, принимавших участие в эксперименте, представлены в табл. 2.

Таблица 2

Характеристики людей с ментальными нарушениями

Имя	Описание нарушений	Требования к интерфейсу
Барт	Легкие ментальные нарушения, может читать и использовать планшет без посторонней помощи. Понимает, как использовать приложение и для чего оно нужно, может использовать приложение без посторонней помощи. Имеет небольшие нарушения зрения	Увеличенный шрифт
Фрэнсис	Легкие ментальные нарушения, нарушения физического характера, когнитивные нарушения, не понимает, как работать с приложением, не может сконцентрировать внимание, умеет читать, может использовать приложение без посторонней помощи	Определение времени задержки между нажатиями и определение времени распознавания нажатия на элемент

Окончание табл. 2

Имя	Описание нарушений	Требования к интерфейсу
Димитрий	Тяжелые ментальные нарушения, нарушения моторики рук, не умеет читать и говорить, не понимает назначения приложения, не может использовать приложение без посторонней помощи	Определение времени задержки между нажатиями и определение времени распознавания нажатия на элемент, использование картинок вместо текста
Джелл	Легкие ментальные нарушения, нарушение зрения, не умеет читать, может использовать приложение без посторонней помощи	Использование картинок вместо текста
Таллал	Средние ментальные нарушения, нарушение зрения, не умеет читать, не может использовать приложение без посторонней помощи	Использование картинок вместо текста

Для оценки качества разработанного интерфейса приложения были выделены следующие критерии:

(К1-К2) – выполнил ли пользователь до конца предоставляемое ему задание с помощью приложения-аналога / разработанного приложения;

(К3-К4) – возникали ли трудности при выполнении задания с помощью приложения-аналога / разработанного приложения;

(К5-К6) – понадобилась ли помощь ассистента при выполнении задания с помощью приложения-аналога / разработанного приложения;

(К7-К8) – понравилось ли пользователю выполнять задание с помощью приложения-аналога / разработанного приложения.

Результаты тестирования по каждому пользователю представлены в табл. 3.

Таблица 3

Результаты тестирования приложения в группе пользователей с ментальными нарушениями

Критерии	Барт	Френсис	Димитрий	Джелл	Таллал
К1	Да	Да	Нет	Да	Да
К2	Да	Да	Да	Да	Да
К3	Нет	Да, нажимал не те кнопки	Да, не мог нажать кнопки	Нет	Нет
К4	Нет	Нет, хотел пройти задание быстрее	Нет	Нет	Нет
К5	Нет	Да, в полной мере	Да, в полной мере	Частично	Частично
К6	Нет	Да, для перевода	Да, для перевода	Нет	Нет
К7	Да	Частично	Частично	Да	Частично
К8	Да	Частично	Да	Да	Да

Из таблицы видно, что приложение, индивидуально подстроенное под пользователя, разработанное с помощью рекомендаций и дополненное набором паттернов для настроек, решило некоторые проблемы пользователей.

Таким образом, в ходе тестирования было подтверждено, что приложение с применением адаптации является более дружественным к пользователю, чем без ее использования.

Выводы

В результате тестирования разработанной программной системы в группе разработчиков была доказана польза выдачи рекомендаций по

созданию интерфейсов для людей с определенным типом нарушений, так как сами разработчики о таких требованиях не были осведомлены.

Тестирование интерфейса приложения для профориентации в группе людей с ограниченными возможностями показало, что выдача рекомендаций в совокупности с универсальным меню настроек позволили создать интерфейс, взаимодействие с которым вызывает гораздо меньше проблем, чем взаимодействие с интерфейсом, разработанным без рекомендаций и не имеющим возможности персональной настройки для пользователя.

В дальнейшем планируется доработать программное средство, учитывая, что вышла новая версия Unity с поддержкой .NET> 3.5, что позволит заменить отдельное приложение для работы с онтологией на встроенную библиотеку. Кроме того, планируется расширить модель паттерна и реализовать дополнительную адаптацию интерфейса на основе сбора и анализа поведения пользователя в процессе взаимодействия с интерфейсом.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. World Health Statistics [Электронный ресурс]. – Режим доступа : http://www.who.int/gho/publications/world_health_statistics/2016/en/, – 284p. (дата обращения 10.12.2017).
2. Wilbert, O. The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques / O. Wilbert – 3rd edition. – WILEY. – 2007. – 88 p.
3. Ergonomics of Human System Interaction [Текст]: нормативно-технический материал: International Organization for Standardization, 2008. – 28 с.
4. AEGIS Ontology. Официальный сайт. [Электронный ресурс]. – Режим доступа : <http://www.aegis-project.eu>.
5. Jena: A free and open source Java framework for building Semantic Web and Linked Data applications. Официальный сайт [Электронный ресурс]. – Режим доступа : <https://jena.apache.org/>. (дата обращения 10.03.2017).
6. Tutorial: Jena Semantic Web Framework [Электронный ресурс]. – Режим доступа: <http://trime-nlp.blogspot>.

be/2013/06/introduction-to-jena.html (дата обращения 10.03.2017).

7. Поцелуйко, А.С. Метод адаптации графического интерфейса для людей с ограниченными возможностями на основе онтологической модели пользователя и паттернов интерфейса / А. С. Поцелуйко, Р. Р. Романенко, М. Б. Кульцова // Известия Волгоградского государственного технического университета. 2017. № 1(196). С. 89–97.

8. Онтологическая модель пользователя для автоматизированного создания адаптивных интерфейсов для людей с ограниченными возможностями / А. С. Поцелуйко, М. Б. Кульцова, А. В. Аникин, Р. Р. Романенко // Известия Волгоградского государственного технического университета. 2016. № 11 (190). С. 134–141.

9. Duarte, C. GUIDE: Creating Accessible TV Applications / C. Duarte, P. Langdon, C. Jung, J. Coelho, P. Biswas, P. Hamisu // Second W3C Workshop Web & TV. – Series: Assistive Technology Research Series. – 2011. – Vol. 29. – P. 905–912.

10. Software Quality Characteristics : нормативно-технический материал : International Organization for Standardization, 2011. – 34 с.

11. A Two-Phase Method of User Interface Adaptation for People with Special Needs / М.Б. Кульцова, А.С. Поцелуйко, И.Г. Жукова, А. Скориков, Р. Романенко // Creativity in Intelligent Technologies and Data Science. Second Conference, CIT&DS 2017 (Volgograd, Russia, September 12-14, 2017) : Proceedings / ed. by A. Kravets, M. Shcherbakov, M. Kultsova, Peter Groumpos ; Volgograd State Technical University [et al.]. – [Germany] : Springer International Publishing AG, 2017. – P. 805–821. – (Ser. Communications in Computer and Information Science ; Vol. 754).

УДК 623.618

А. В. Старусев, Л. А. Михолап, О. В. Кислов, В. И. Лобейко

МЕТОД ОЦЕНКИ ЗАТРАТ ТРУДА ПРОГРАММИСТА ПРИ РЕШЕНИИ ЗАДАЧИ ИСПЫТАНИЙ

Астраханский государственный университет

starusev-av@yandex.ru

В статье предложен метод, позволяющий определить затраты труда на программирование при комплексной задаче испытания сложной технической системы в конкретные сроки, заданные в техническом задании. Данный метод ориентирован на традиционную технологию программирования, однако для практического использования изложенного метода необходимо располагать знаниями соответствующих нормативов, полученных экспериментально.

Ключевые слова: затраты труда, программирование, комплексная задача, испытание сложной технической системы, трудоемкость.

A. V. Starusev, L. A. Mikholaп, O. V. Kislov, V. I. Lobeyko

METHOD OF THE ASSESSMENT OF EXPENSE OF WORK OF THE PROGRAMMER AT SOLUTION OF THE PROBLEM OF TESTS

Astrakhan State University

In article the method allowing to define work costs of programming at a complex problem of test of difficult technical system in the concrete terms set in the specification is offered. This method is focused on traditional technology of programming, however for practical use of the stated method it is necessary to have knowledge of the corresponding standards received experimentally.

Keywords: work expenses, programming, complex task, test of difficult technical system, labor input.

Введение

Одним из главных факторов, определяющих затраты на разработку и испытания сложных технических систем, является подготовка задач к решению на электронных вычислительных машинах (ЭВМ). Эти затраты составляют около половины всех расходов на создание сложных технических систем. Следует также отметить, что нет достаточно надежных критериев, методик и расчетных зависимостей для оценки трудоемкости программирования решаемых при создании технических систем задач. Используемые на практике методы нормирования в основном носят субъективный характер и основываются на экспертных оценках [1, 2, 4].

Постановка задачи

Как показывает опыт [3, 4, 5], значения нормативов программирования зависят от многих факторов [6, 7, 8]: алгоритмической сложности задачи, знаний и опыта программиста, используемого языка программирования и др. Все это предопределяет необходимость тщательного изучения структуры трудоемкости, а также разработки метода оценки затрат программирования определенных (комплексных) задач при испытаниях сложных технических систем.

Решение

Следует отметить, что программа решения любой задачи состоит из большого числа (до нескольких тысяч) команд, представляемых специальными символами и размещенных в строгой последовательности. Это значит, что каждая задача должна быть проанализирована и подготовлена к решению в виде последовательности операций, реализуемых на электронных вычислительных машинах (ЭВМ) в виде выполнения программы. Такая подготовка задач, а также написание детальной последовательности операций в кодах команд машины определяют трудоемкость программирования. Продолжительность программирования складывается из ряда составляющих, связанных с основными этапами выполнения работы.

В общем случае затраты труда (в человеко-часах) при подготовке задач к решению на ЭВМ (рис. 1) целесообразно определять следующим образом:

$$T_{\Sigma} = T_o + T_u + T_a + T_{\delta c} + T_{\kappa} + T_{om} + T_{\delta}, \quad (1)$$

где T_o – затраты труда на подготовку описания задачи; T_u – затраты труда на изучение описа-

ния задачи; T_a – затраты труда на разработку алгоритма решения задачи; $T_{\delta c}$ – затраты труда на разработку блок-схемы программы; T_{κ} – затраты труда на программирование (кодирование) по готовой блок-схеме вручную или с использованием алгоритмического языка; T_{om} – затраты труда на отладку программы на ЭВМ; T_{δ} – затраты труда на подготовку документации по задаче (инструкции и т. д.) в рукописном виде.

Слагаемое затрат труда определяются через условное количество команд в программе разрабатываемой задачи. Так называется предполагаемое общее число команд, которое требуется написать программисту в процессе работы над задачей с учетом возможных уточнений в ее постановках и в процессе совершенствования программы.

Условное количество команд (W) в программе задачи может быть определено следующим образом:

$$W = \lambda C(1 + \sum_n P_n), \quad (2)$$

где λ – предполагаемое число команд; C – коэффициент сложности программы; P_n – коэффициент корректировки программы при ее разработке; n – количество коррекций программы в ходе ее разработки.

Целесообразно использовать помимо названных коэффициентов и другие: коэффициент квалификации программиста (K); коэффициент затрат на алгоритмизацию (A); коэффициент увеличения затрат труда вследствие недостаточного или некачественного описания задачи (H).

Коэффициенты, которые используются при оценке затрат труда на подготовку задач к решению на ЭВМ, характеризуют:

коэффициент сложности программы C – относительную сложность программы задач по отношению к так называемой типовой задаче, сложность которой принята равной 1. За типовую задачу принимаются задачи, результат решения которых не выдается на алфавитно-цифровое печатающее устройство, а после обновления хранится в памяти ЭВМ. Задачи других классов определяются через коэффициент сложности в зависимости от форм выходных документов следующим образом: задачи управления – $1,25 \div 1,5$; задачи планирования – $1,25 \div 2,0$;

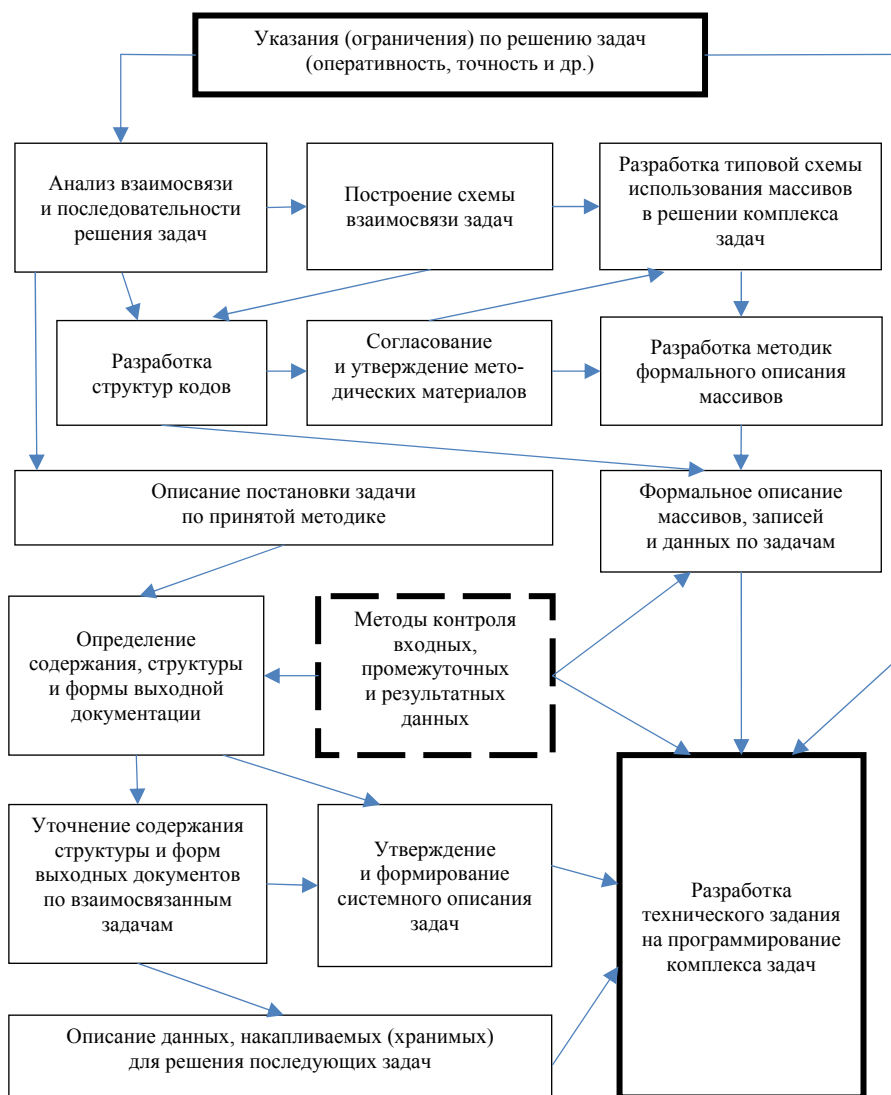


Рис. 1. Схема системного описания задач

коэффициент коррекции программы P – увеличение объема работ за счет внесения изменений в алгоритм или программу решения задачи по результатам уточнения постановок и описаний задач, изменения состава и структуры информации (входной и выходной), а также уточнений, выносимых разработчиком для улучшения качества самой программы без изменения постановки задачи. На практике при разработке программы в среднем вносится $3 \div 5$ коррекций. Каждая из них ведет к переработке от 5 до 10 % готовой программы. Коррекция, вызванная изменением постановки задачи, может привести даже к разработке новой программы, поэтому при определении условного количества команд в программе учитываются лишь те коррекции, которые вызваны стремлением улучшить качество программы без изменения ее постановки;

коэффициент квалификации программиста K – степень подготовленности программиста к порученной ему работе. Этот коэффициент определяется в зависимости от опыта и стажа работы и составляет: для работающих до 2 лет – 0,8; от 2 до 3 лет – 1,0; от 3 до 5 лет $1,1 \div 1,2$; от 5 до 7 лет – $1,3 \div 1,4$; свыше 7 лет – $1,5 \div 1,6$;

коэффициент затрат на алгоритмизацию A – трудоемкость разработки алгоритма по отношению к программированию. Этот коэффициент в общем случае является функцией сложности разрабатываемой задачи. В практике разработки считается, что для задач управления коэффициент затрат на алгоритмизацию равен $0,4 \div 0,5$, а для задач планирования – $0,5 \div 0,7$ от затрат на программирование;

коэффициент увеличения затрат труда вследствие недостаточного или некачественного опи-

сания задачи H – качество описания задачи, выданной для разработки программы. Так как задачи всегда в той или иной степени требуют уточнения и соответствующей доработки, $H \geq 1$. Практика показывает, что в большинстве случаев этот коэффициент в зависимости от сложности задач находится в пределах от 1,2 до 1,5.

Порядок определения затрат труда на различных этапах работы целесообразно применять следующий:

1. Затраты труда на подготовку описания задачи (T_o) связаны с творческим характером работы. Практика показывает, что в зависимости от назначения, сложности задачи на ее описание требуется в среднем от 1 до 3 человеко-месяцев труда специалиста соответствующего профиля при хорошей организации работы у заказчика.

2. Затраты труда на изучение описания задачи (T_{II}) с учетом уточнения описания могут быть определены по следующей формуле (в чел.-час):

$$T_{II} = \frac{W}{(75 \div 85)R_i} H, \quad (3)$$

где R_i – лимит ресурса на разработку i -ой задачи.

На практике описания задач, выдаваемые заказчиком на разработку, как правило, требуют уточнения постановки, состава информации, ее количественных характеристик и форм выходных документов, поэтому с достаточной точностью можно принимать значение коэффициента H в пределах от 1,2 до 1,5.

3. Затраты труда на разработку алгоритма решения задач рассчитываются следующим образом:

$$T_a = \frac{W}{(20 \div 25)K}. \quad (4)$$

4. Затраты труда на разработку блок-схемы программы:

$$T_{BC} = \frac{W}{(10 \div 15)K}. \quad (5)$$

5. Затраты труда на непосредственное кодирование программы по готовой блок-схеме программы:

при ручном программировании:

$$T_p = \frac{W}{(10 \div 15)K}, \quad (6)$$

при использовании алгоритмического языка:

$$T_{Ka} = \frac{W}{(20 \div 25)K}. \quad (7)$$

6. Затраты труда на отладку программы на ЭВМ:

при автономной отладке одной задачи:

$$T_{OT} = \frac{W}{(4 \div 5)K}, \quad (8)$$

при комплексной отладке задачи:

$$T_{OT}^* = 1,5T_{OT}. \quad (9)$$

7. Затраты труда на подготовку документации по задаче составляют:

$$T_d = T_{Dr} + T_{Do}, \quad (10)$$

где $T_{Dr} = \frac{W}{(15 \div 20)K}$ – подготовка материалов

в рукописи; $T_{Do} = 0,75T_{Dr}$ – редактирование, печать и оформление отчета по задаче.

Зная предполагаемое количество команд в программе задачи λ и определив затраты труда на ее разработку T_{Σ} , можно ориентировочно оценить производительность труда программиста (команд в час) при разработке задачи:

$$R = \frac{\lambda}{T_{\Sigma}}. \quad (11)$$

На практике при разработке сложных комплексных задач производительность программиста средней квалификации составляет примерно $0,5 \div 0,75$ команды в час. Исходя из этого, можно сказать, что при годовом ресурсе времени около 2000 часов программист средней квалификации способен разработать задачу, программа решения которой составляет $1000 \div 1500$ команд [5, 6, 7].

Для практического использования вышеизложенного метода необходимо располагать значениями соответствующих нормативов, полученных экспериментально. Точность рассмотренного метода в значительной мере зависит от того, насколько детализированной и полной информацией располагают о задаче, подлежащей программированию.

Заключение

Предложенный метод позволяет определить затраты труда на программирование задач при испытаниях сложных технических систем, что весьма актуально при заданных в техническом задании сроках испытания системы.

Описанный выше метод ориентирован на традиционную технологию программирования, когда каждая программа содержит все этапы машинного решения задачи, начиная от ввода исходных данных и кончая печатью результатов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Бешелев, С.Д. Экспертные оценки / С.Д. Бешелев, Ф.Г. Гурвич. – М.: Наука, 1973. – 79 с.
2. Глушков, В.М. О прогнозировании на основе экспертных оценок / В.М. Глушков // Кибернетика. 1969. № 2. С. 2–4.
3. Старусев, А.В. Системный подход к обеспечению надежности и безопасности моделирования при полунатурных испытаниях автоматизированных систем управления / А.В. Старусев, В.И. Лобейко, С.А. Горемыкин // Известия Волгоградского государственного технического университета. 2017. № 1(196). С. 78–82.

4. Вопросы. Гипотезы. Ответы: наука XXI века / Л.А. Грачева, Ю.А. Извеков, Л.А. Михолап, Л.В. Объедкова, Т.В. Опейник, М.С. Пантелеева, А.И. Поспеловская, Д.В. Поспеловский, А.В. Старусев : кол. моногр. – Краснодар, 2017. Кн. 15. – 143 с.
5. Клейнен, Дж. Статистические методы в имитационном моделировании : пер. с англ. / Дж. Клейнен. – М.: Статистика, 1978. – 221 с.
6. Трахтенгерц, Э.А. Программное обеспечение автоматизированных систем управления. – М.: Статистика, 1974. – 228 с.
7. Лобейко, В.И. Современные подходы к организации испытаний сложных систем / В.И. Лобейко. – Астрахань : Астраханский университет, 2006. – 332 с.

УДК 623.618

*А. В. Старусев, Л. А. Михолап, О. В. Кислов, В. И. Лобейко***МЕТОД ОЦЕНКИ ПРОИЗВОДИТЕЛЬНОСТИ ЭВМ
ПРИ ИСПЫТАНИЯХ СЛОЖНЫХ ТЕХНИЧЕСКИХ СИСТЕМ****Астраханский государственный университет**

starusev-av@yandex.ru

Разработан метод, позволяющий сравнить между собой имеющиеся модели электронно-вычислительных машин и выбрать именно ту модель, которая удовлетворяет заданным требованиям, определенным в техническом задании на проведение испытаний опытного образца сложной технической системы специального назначения. Применять данный метод целесообразно при ограниченных условиях по временным и экономическим параметрам.

Ключевые слова: производительность, пропускная способность электронной вычислительной машины, испытания, сложные технические системы.

*A. V. Starusev, L. A. Mikholaп, O. V. Kislov, V. I. Lobeyko***COMPUTER PRODUCTIVITY ASSESSMENT METHOD AT
TESTS OF DIFFICULT TECHNICAL SYSTEMS****Astrakhan State University**

The method allowing to compare among themselves the available models of electronic computers and to choose that model which meets the set requirements defined in the specification on carrying out tests of a prototype of difficult technical system of a special purpose is developed. It is possible to apply this method under limited conditions on temporary and economic parameters.

Keywords: productivity, capacity of the electronic computer, test, difficult technical systems.

Введение

Производительность электронно-вычислительных машин (ЭВМ) при решении вопросов технического обеспечения испытаний сложных технических систем является расчетным показателем, по которому производится выбор вычислительного оборудования. Производительность ЭВМ определяется в конкретных условиях загрузки и применения с учетом различия и функциональных возможностей машин, набора команд, составе внешних и периферийных устройств ЭВМ и т. д. Рассматриваемые вопросы особенно актуальны для моделей ЭВМ, сравниваемых между собой по целому ряду по-

казателей, обладающих общими средствами математического и программного обеспечения, единой конструктивно-технологической базой, составом команд, форматом информационных слов (и т. д.) но различающихся производительностью базовых процессов [1, 9, 10].

Под производительностью ЭВМ будем понимать ее способность выполнять определенный объем полезной работы в единицу времени. При этом различаются понятия «быстродействие» и «производительность» (номинальное быстродействие) ЭВМ. Быстродействие есть среднее количество фиксированных (эталонных) операций, выполняемых машиной в еди-

ницу времени (секунду) при реализации конкретной программы. В качестве эталонной может быть выбрана любая операция, входящая в систему операций рассматриваемой ЭВМ. Очевидно, что этот показатель фиксирует верхнюю границу производительности ЭВМ, так как не учитывает характера нагрузок машины, условий ее эксплуатации, используемых средств математического обеспечения (в первую очередь операционной системы).

Постановка задачи

Необходимо разработать метод, который позволил бы оперативно сравнивать между собой имеющиеся модели ЭВМ (по наиболее важным показателям) и позволяющий выбрать ту модель, которая удовлетворяла бы заданным в техническом задании на проведение испытаний сложных технических систем требованиям.

Метод решения

Следует отметить, что при испытаниях сложных технических систем производительность определяется средним числом операций, выполняемых ЭВМ в единицу времени при реализации некоторой программы, составленной из фиксированного набора операций и частоты их реализации:

$$P_n = \frac{10^6}{\sum_{i=1}^n P_i \tau_i \alpha_i}, \quad (1)$$

где P_i – частота выполнения i -й операции; τ_i – время (мкс) выполнения операций i -го типа; α_i – удельный вес i -й операции в наборе; n – число типов операций.

Наиболее существенное влияние на производительность моделей ЭВМ оказывает операционная система, которая выполняет ряд функций по организации вычислительного процесса, в частности, управляет данными, заданиями и задачами (см. рисунок, приведенный ниже).

К операционной системе (ОС) от программ поступают задания на выполнение различных операций (например, на ввод или вывод данных). При организации вычислительного процесса операционная система руководствуется, с одной стороны, совокупностью поступивших заданий, с другой стороны, данными о состоянии внешних и периферийных устройств ЭВМ. В результате значительно уменьшается время прохождения задания (или задачи) и тем самым увеличивается производительность ЭВМ.

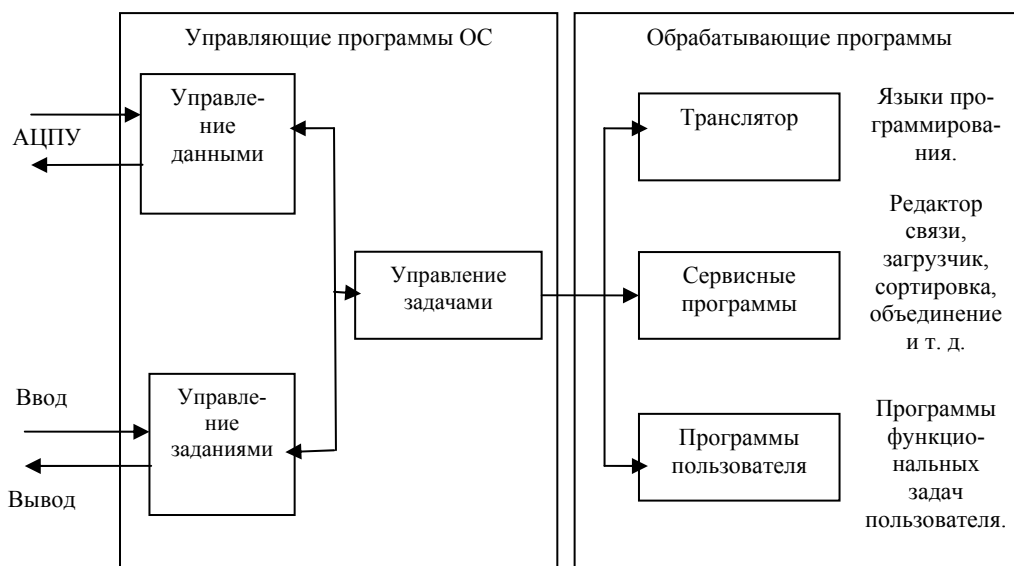


Схема управления задачами в ЭВМ при испытаниях сложных технических систем специального назначения:
ОС – операционная система; АЦПУ – автоматизированный центральный пункт управления

При такой организации вычислительных работ на практике используется понятие «пропускная способность ЭВМ» или ее отдельных устройств [2, 3]. Сущность данного понятия состоит в том, что ОС при фиксированной производительности отдельных устройств определя-

ет потоки заданий (или задач) как между пользователем и ЭВМ, так и между устройствами ЭВМ. Поток заданий (задач), проходящий через устройство и соответствующий его максимальной производительности, определяет пропускную способность устройства.

Например, пропускная способность ЭВМ в режиме пакетной обработки:

$$R = \frac{n}{\sum_{i=1}^n (t_{ip} + t_{ic} + t_{iобс})}, \quad (2)$$

а в случае совмещения работы процессора и внешних устройств равна:

$$R(c) = \frac{n}{\sum_{i=1}^n (t_{ip}^c + t_{ic}) + t_{обс}}, \quad (3)$$

где n – число выполняемых заданий (решаемых задач); t_{ip} – время выполнения i -го задания (задачи); t_{ip}^c – время выполнения i -го задания (задачи) в режиме совмещения; t_{ic} – время работы операционной системы (системное время) для выполнения i -го задания (задачи); $t_{iобс}$ – время обслуживания ЭВМ, приходящееся на период решения задачи. Обычно для правильно спроектированной ЭВМ и вычислительного процессора $\sum t_{ip} \gg t_{обс}$.

При многопрограммной работе ЭВМ сложно точно определить, какая часть времени работы операционной системы (системное время) затрачивается на отдельное задание. Однако современные ЭВМ позволяют определить время работы процессора на реализацию отдельной программы, что дает возможность обоснованно сравнивать производительности ЭВМ на основе различного рода набора операций, характерных для процедур или задач [4, 5].

На практике сравнительная оценка произ-

водительности моделей ЭВМ проводится следующим образом:

1. Определялось быстродействие сравниваемых ЭВМ при выполнении каждого типа операций:

$$P_{ij} = \frac{1}{\tau_{ij}}, \quad (4)$$

где τ_{ij} – время (мкс) выполнения i -го типа операций ($i=1, \dots, m$) для j -й ЭВМ ($j=1, \dots, n$);

2. Рассчитывался усредненный выравнивающий коэффициент (Q_i) для каждого типа операций. Коэффициент определяет среднее соотношение затрат времени на выполнение различных операций для разных ЭВМ:

$$Q_i = \frac{\sum_{j=1}^n P_{ij}}{\sum_{j=1}^n \sum_{i=1}^m P_{ij}}, \quad (5)$$

где m – число типов операций в наборе; n – число сравниваемых ЭВМ. Очевидно, что

$$\sum_{i=1}^m Q_i = 1;$$

3. Вычислялись суммарные затраты времени T_j ($j=1, 2, \dots, n$) на выполнение набора типовых операций для каждой ЭВМ с учетом удельного веса операций в наборе (табл. 1):

$$T_j = \sum_{i=1}^m \alpha_i \tau_{ij} Q_i, \quad (6)$$

где α_i – удельный вес типа операций для рассматриваемых групп решаемых задач.

Таблица 1

Суммарные затраты времени на выполнение набора типовых операций для сравниваемых ЭВМ

Типовая группа задач	Время выполнения набора операций, мкс			
	1-й тип ЭВМ	2-й тип ЭВМ	3-й тип ЭВМ	4-й тип ЭВМ
Обработка данных	15,634	3,4511	0,765	0,6526
Вычислительного типа	40,003	3,752	1,264	0,420
Комбинация задач типовых групп	19,29	3,497	0,83	0,618

Практика использования ЭВМ при испытаниях сложных технических систем показала, что реальная производительность машин зависит не только от производительности их процессора, но во многом определяется составом и характеристиками внешних устройств, объемами внешней и оперативной памяти, средствами математического и программного обеспе-

чения и т. д. В этой связи следует рассматривать производительность ЭВМ с учетом как перечисленных, так и других факторов, имеющих место у конкретного пользователя, а также надежность ЭВМ.

Очевидно, что значения вычисленной в данном случае производительности ЭВМ ограничены сверху быстродействием машины, а снизу

производительностью ЭВМ, вычисленной по фиксированному выше набору операций.

Для того, чтобы получить общую оценку увеличения производительности всех определенных типов моделей ЭВМ по сравнению с базовой 1-го типа, необходимо вычислить значения индексов производительности, взвешенные с учетом объемов информации, обрабатываемой по задачам соответствующих групп (комбинация задач).

Индекс производительности данной ЭВМ (W_j) по отношению к базовой определяется

как отношение времени выполнения набора операций для сравниваемого и базового процессов (табл. 2):

$$W_j = \frac{T_j}{T_{j+1}}. \quad (7)$$

Сопоставление данных табл. 1 и 2 показывает, что совпадение результатов вполне удовлетворительное, что свидетельствует о пригодности описанного метода оценки производительности ЭВМ.

Таблица 2

Индексы производительности сравниваемых ЭВМ по отношению к базовой

Типовая группа задач	Индекс сравнительной производительности			
	1-й тип ЭВМ	2-й тип ЭВМ	3-й тип ЭВМ	4-й тип ЭВМ
Обработка данных	1,0	4,53	20,4	23,8
Вычислительного типа	1,0	10,6	31,7	97,6
Комбинация задач типовых групп	1,0	5,5	23,2	30,6

Автоматизация управления производством, а также единый подход в области технического оснащения и математического обеспечения автоматизированных систем управления производством способствуют внедрению высокопроизводительных вычислительных комплексов [5].

Современный высокопроизводительный вычислительный комплекс невозможно описать с помощью традиционных показателей производительности, так как существенную роль здесь приобретают такие показатели комплекса, как режим обработки данных, сохранность информации, возможность блочного наращивания схемных и программных компонент, эффективность работы с языками высокого уровня и т. д. Влияние перечисленных показателей на общую производительность комплекса с трудом поддается количественной оценке и последующему аналитическому описанию. Решение этих проблем является предметом специального исследования, что позволит учесть взаимное влияние перечисленных факторов и тем самым оценить производительность вычислительных комплексов [5, 6, 7, 8].

Заключение

Метод позволяет оперативно сравнить между собой по важным показателям имеющиеся модели ЭВМ и выбрать ту модель, которая удовлетворяет заданным требованиям. Данный метод особенно актуален при решении вопросов технического обеспечения испытаний сложных технических систем в условиях ограниченных временных и экономических затрат.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Липаев, В.В. Проектирование математического обеспечения АСУ (системотехника, архитектура, технология) / В.В. Липаев. – М.: Сов. радио, 1977. – 400 с.
2. Шенно, Р. Имитационное моделирование – искусство и наука / Р. Шенно. – М.: МИР, 1978. – 420 с.
3. Липаев, В.В. Методы обеспечения качества крупномасштабных программных средств / В.В. Липаев. – М.: СИНТЕГ, 2003. – 520 с., ил.
4. Балыбердин, В.А. Проблемные вопросы создания и внедрения новых информационных технологий в автоматизированных системах военного назначения / В.А. Балыбердин, О.В. Пенкин, А.И. Полунин. – М.: Вооружение. Политика. Конверсия, 2001. – 146 с.
5. Киселев, В.Д. Теоретические основы оптимизации информационно-вычислительного процесса и состава комплексов средств защиты информации в вычислительных сетях / В.Д. Киселев, О.В. Есиков, А.С. Кислицин. – М.: Полиграфсервис XXI век, 2003. – 153 с.
6. Лобейко, В.И. Современные подходы к организации испытаний сложных систем / В.И. Лобейко. – Астрахань: Астраханский университет, 2006. – 332 с.
7. Старусев, А.В. Метод оценки трудоемкости процессов программирования / А.В. Старусев // Прикаспийский журнал: управление и высокие технологии. 2012. № 4(20). С. 51–54.
8. Старусев, А.В. Метод повышения эффективности использования ресурсов ЭВМ / А.В. Старусев // Прикаспийский журнал: управление и высокие технологии. 2013. № 1(21). С. 12–16.
9. Фоменков, С.А. Формирование баз данных по физическим эффектам для концептуального проектирования технических систем / С.А. Фоменков, С.Г. Колесников, Д.М. Коробкин // Сборник научных трудов SWorld, 2014. Т. 5. № 1. С. 54–58.
10. Старусев, А.В. Оптимизация технологии подготовки и проведения экспериментов с использованием имитационного моделирования / А.В. Старусев, В.И. Лобейко, С.А. Горемыкин. Известия Волгоградского технического университета. 2017. № 8(203). С. 40–44.

ПОСТРОЕНИЕ СЕТЕЙ И ЗАЩИТА ИНФОРМАЦИИ

УДК 699.887.32

И. П. Михнев¹, Н. А. Сальникова¹, А. Г. Кравец²

ЗАЩИТА ИНФОРМАЦИИ ОТ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА ПРИ АНАЛИЗЕ РАДИАЦИОННЫХ ХАРАКТЕРИСТИК ПОМЕЩЕНИЙ СПЕКТРОМЕТРИЧЕСКИМ МЕТОДОМ

¹ Волгоградский институт управления – филиал Российской академии
народного хозяйства и государственной службы при Президенте РФ

² Волгоградский государственный технический университет
mkmco@list.ru, ns3112@mail.ru, agk@gde.ru

В статье представлены исследования средств защиты информации от несанкционированного доступа автоматизированной системы радионуклидной спектрометрии. В результате проведенных исследований получены показатели защищенности системы, которые позволяют рассчитать и оптимизировать вероятность ущерба от несанкционированного доступа с учетом времени эксплуатации и применяемых средств защиты информации.

Ключевые слова: естественные радионуклиды, информационная безопасность, несанкционированный доступ, система радионуклидной спектрометрии, угрозы безопасности информации.

I. P. Mikhnev¹, N. A. Salnikova¹, A. G. Kravets²

PROTECTION OF INFORMATION FROM UNAUTHORIZED ACCESS IN THE ANALYSIS OF RADIATION CHARACTERISTICS OF PREMISES BY SPECTROMETRIC METHOD

¹ Volgograd Institute of Management – branch of the Russian Presidential
Academy of National Economy and Public Administration

² Volgograd State Technical University

In the article the research of means of information protection against unauthorized access of the automated system of radionuclide spectrometry is presented. As a result of the conducted researches, the system's security indicators have been obtained, which allow to calculate and optimize the probability of damage from unauthorized access taking into account the operating time and the applied information protection means.

Keywords: natural radionuclides, information security, unauthorized access, system radionuclide spectrometry, threats to information security.

В последние годы при строительстве зданий используются самые разнообразные строительные материалы, как отечественного, так и зарубежного производства. Поскольку дозовые нагрузки облучения населения в помещениях зависят от содержания активности естественных радионуклидов (ЕРН) в строительных материалах, выбора мест застройки и конструкций зданий, возможно ограничить облучение населения от природных источников излучения путем вмешательства в сложившуюся практику строительства [1, 2]. Существующая практика производства строительных материалов складывалась с учетом их стоимости. Поэтому учет дополнительного критерия – степени радиаци-

онного воздействия на население приведет к определенному повышению стоимости производства строительных материалов. Для высокой точности оценки радиационного фона в жилых помещениях требуется автоматизированная система радионуклидной спектрометрии (АСРС), позволяющая измерять удельные активности ЕРН в объектах внешней среды, а также предельно низкие мощности дозы гамма-излучения с разделением вклада в показания приборов, обусловленного космическим и гамма-излучением от строительных материалов [3, 4].

Наиболее приемлемыми приборами для измерения указанных выше радиационных характеристик являются дозиметр ДРГ-01Т1 и радио-

метр СРП-68-01, обладающие высокой чувствительностью, достаточной точностью измерений, удобством и простотой проведения массовых измерений. Для определения удельных активностей ЕРН в строительном сырье, материалах, почве, древесине и др. целесообразно использовать ASRS (Automated Systems Radionuclide Spectrometry) автоматизированную систему радионуклидной спектрометрии с интегрированным универсальным спектрометрическим комплексом (УСК «Гамма Плюс Р») на базе сцинтилляционного гамма-спектрометра, с программным обеспечением «Прогресс-5.1», обеспечивающий установление класса материала [5, 6]. АСРС «Гамма Плюс Р» может использоваться для решения широкого спектра задач радиационного контроля: от измерений в области сертификации соответствия пищевой продукции, питьевой воды, строительных материалов, продукции лесного хозяйства и др. до мониторинга и задач радиационного контроля на предприятиях ядерного цикла, а также для решения целого ряда исследовательских задач, связанных с измерением радиоактивности [7]. Установка может поставляться в различной комплектации в соответствии с требованием заказчика. АСРС «Гамма Плюс Р» состоит из блоков детектирования, защиты от внешнего гамма излучения, электронного устройства и внешнего блока питания. Но этот спектрометрический комплекс не поставляется с защитой программного пакета от вредоносного кода и НСД, что может повлечь за собой сбой в работе ПК и потерю информации при НСД. Обработка спектров, расчет активности и погрешности производится с использованием программного обеспечения [8]. Обнаружение гамма-излучения основано на регистрации эффектов, возникающих при его взаимодействии с веществом. Гамма-кванты, испускаемые атомными ядрами при радиоактивных превращениях, имеют определенные физические характеристики, которые можно использовать для их регистрации. Измеряя энергию и интенсивность испускаемых гамма-квантов, а также оценивая период полураспада их отдельных моноэнергетических групп, можно идентифицировать радионуклиды в измеряемых счетных образцах и достаточно точно определить их абсолютную активность. С целью преобразования аналогового спектрометрического сигнала, поступающего с выхода детектора, в цифровой применяют амплитудно-цифровой преобразователь

(АЦП). Управление работой АЦП осуществляется при помощи специальных программ, входящих в состав программного пакета [3, 5]. Обработку спектров, расчет активности и погрешности производят с использованием программного пакета «Прогресс-5.1».

В настоящее время резко обострились проблемы защиты АСРС и объектов критической информационной инфраструктуры РФ от кибернетического оружия, что позволило сформировать актуальность исследования, которая имеет место в соответствии с основными документами РФ по безопасности – Стратегии национальной безопасности и Доктрины информационной безопасности РФ [9]. По данным Совета Безопасности РФ, в 2016 году было совершено более 50 миллионов кибератак на российские информационные ресурсы, причем 60 % атак велось из-за рубежа [10].

Для обеспечения требуемого уровня защиты информации от несанкционированного доступа при анализе радиационных характеристик помещений спектрометрическим методом, а также проектировании и работе автоматизированных систем радионуклидной спектрометрии, реализуются организационные, технические и организационно-технические меры защиты информации. Организационные меры предназначены для руководящего состава, органов по защите информации, других пользователей и заключаются в организации, упорядочении, контроле деятельности по защите информации в организации. Технические и организационно-технические меры защиты информации предусматривают применение технических средств, которые объединяются в комплексы средств защиты информации (КСЗИ) и являются составной частью АСРС. Степень реализации мер по защите информации оценивается в процессе проектирования и работы АСРС и зависит от оптимального комплектования КСЗИ средствами защиты информации (СЗИ) и эффективностью функционирования КСЗИ в целом [5, 11]. Известно, что на реализацию несанкционированного доступа (НСД) к информации, приводящих к нарушению нормального функционирования автоматизированных систем радионуклидной спектрометрии, конфиденциальности, целостности и доступности информации, нарушитель всегда будет затрачивать время $T_{нсд}$, необходимое для образования канала реализации угрозы безопасности информации, то есть указанное время характеризует временной интервал:

$$T_{нсд} = \sum_{i=1}^4 T_i, \quad (1)$$

где T_1 – выявление уязвимостей программного (аппаратного) обеспечения; T_2 – оценка возможности эксплуатации уязвимости с учетом существующей системы защиты информации предполагаемого объекта воздействия (носителя информации); T_3 – выбор способа реализации несанкционированного доступа; T_4 – осуществление НСД.

Исходя из этого, путем увеличения T_i всегда можно было бы управлять защищенностью информации в АСРС. Таким образом, T_i можно было бы принять в качестве критерия для оценки защищенности информации в АСРС. Тогда путем задания при проектировании АСРС порогового значения $T_{доп\ нсд}$ и обеспечив выполнение условия $T_{нсд} \leq T_{доп\ нсд}$, можно было бы реализовать допустимую защиту информации ограниченного доступа в АСРС. Однако такой подход не будет отражать реальную картину, так как время T_i – это случайная величина, закон распределения которой сложно вычислить, так как он будет меняться в зависимости от возможностей нарушителя. Кроме того, здесь не учитываются основные факторы эксплуатации, такие как: различные угрозы безопасности информации в АСРС, время эксплуатации АСРС, характеристики используемых средств защиты информации (СЗИ), от которых также может зависеть НСД к информации.

Для повышения объективности контроля своевременности, достоверности, полноты и непрерывности защищенности информации, проектируемых АСРС целесообразно разработать математическую модель вероятности НСД к изменяющейся информации с учетом условий эксплуатации и состава комплекса средств защиты информации (КСЗИ). На базе найденной модели сформулировать качественные и количественные критерии повышения защищенности информации при проектировании и работе АСРС. Таким образом, НСД к информации в АСРС, будет зависеть от применяемых СЗИ, от количества угроз безопасности информации, степени защищенности и времени эксплуатации АСРС. В соответствии с ГОСТ Р 50922–2006 «Защита информации. Основные термины и определения» целью защиты информации является предотвращение ущерба обладателю информации в связи с возможным НСД к информации, нарушением нормального функционирования АСРС, хищения, модификации или уничтожения информации [12].

Очевидно максимальный ущерб может быть нанесен тогда, когда информация в АСРС скомпрометирована полностью. Такая ситуация может возникнуть при следующих условиях: либо при захвате АСРС противником, либо при ситуации, когда суммарные информационно-технические атаки противника позволяют ему обеспечить НСД к защищаемой информации, циркулирующих по всем защищенным каналам АСРС. Если реализована одна угроза безопасности (УБ), то при моделировании будем считать, что это приводит к минимальному ущербу. Найдем выражение для вероятности НСД к информации, циркулирующей в АСРС.

Пусть проектируется АСРС, содержащая k подразделений, в каждом из которых возможна реализация N_i , $i = 1, 2, \dots, k$ угроз безопасности информации. Всего же АСРС содержит S возможных к реализации УБ, причем

$$S = N_1 + N_2 + \dots + N_k = \sum_{i=1}^k N_i,$$

парирование УБ осуществляется СЗИ, включенных в КСЗИ. СЗИ обладают различными функциональными возможностями по обеспечению защиты, в зависимости от характеристик, реализуемым механизмам защиты, техническим требованиям, совместимостью с другими средствами защиты, экономическими и эргономическими характеристиками. Для различения КСЗИ (СЗИ) целесообразно ввести весовые коэффициенты M_i , $i = 1, 2, \dots, k$. Чем выше гриф секретности обрабатываемой информации, жестче требования к защите и выше требования к техническим характеристикам, тем большее значение должно быть присвоено коэффициенту M_i и наоборот.

Далее введем оценку защищенности информации, определяемую как вероятность НСД к информации при реализации всех возможных к реализации УБ одновременно. Максимальный ущерб возникает, как было указано выше, при реализации всех возможных к реализации УБ, то есть

$$P_y = \prod_{i=1}^k P_{iS}^{N_i}. \quad (2)$$

Теоретически в течение времени эксплуатации АСРС таких попыток может быть бесчисленное множество. Количественно оценить число попыток реализации УБ практически невозможно. Однако можно задать априори шаг указанных попыток во времени. При этом интервал времени, в течение которого может осу-

ществляться одна попытка реализации УБ (T_p) может задаваться с учетом реальных условий эксплуатации и социально-политической, военной обстановки. Например, шаг реализации УБ в мирное время можно приравнять одному месяцу, недели, а во время боевых действий несколькими дням, суткам, часам и т. д. Для заданного значения интервала T_p можно определить количество возможных попыток реализации УБ R за время эксплуатации АСРС объекта T :

$$R = \frac{T}{T_p}, \quad (3)$$

где T – время эксплуатации, а T_p – шаг реализации УБ. Зная количество попыток можно оценить вероятность НСД к защищаемой информации при реализации всех или хотя бы одной УБ за время эксплуатации T :

$$P(t) = 1 - (1 - P_k)^R, \quad (4)$$

где значение P_k – это некоторая оценка, которая характеризует вероятность одной успешной попытки реализации УБ, а $t = T$.

Ранее были приведены два метода расчета оценки P_k : P_x и P_y для наилучшего и наихудшего случая соответственно. Следовательно, если необходимо рассчитать вероятность того, что за период времени T будет осуществлен НСД к информации при реализации хотя бы одной УБ, в выражение (4) необходимо подставить значение $P_k = P_x$. С другой стороны, необходимо рассчитать вероятность наихудшего для системы случая, то есть НСД при реализации всех возможных УБ АСРС одновременно. Тогда в выражение (4) в качестве P_k необходимо подставить значение P_y [13].

Установлено, что наименьшее значение $P_x = 0,23$ принимает в АСРС, где весовые коэффициенты КСЗИ по подразделениям имеют наибольшие значения. В других АСРС при меньших весовых коэффициентах КСЗИ величина P_x принимает меньшие значения. Вызывает интерес тот факт, что в случае, когда весовые коэффициенты КСЗИ равномерны во всех подразделениях одной АСРС, вероятность реализации хотя бы одной УБ ниже, чем в другой АСРС с подразделениями, имеющими различные весовые коэффициенты КСЗИ. Причем общий весовой коэффициент КСЗИ всех подразделений обоих АСРС одинаков [13]. Можно сделать вывод, что P_x существенно увеличивается с ростом попыток реализации УБ. Так для АСРС P_x при одной попытке равна 0,25. При осуществлении нарушителем пяти попыток та-

кая вероятность достигнет значения 0,76. Таким образом, разработанные аналитические оценки позволяют на этапах проектирования АСРС рассчитать верхнюю и нижнюю границы вероятности НСД к информации, что имеет исключительно важное значение для проектирования АСРС. Такой подход дает возможность при проектировании оптимизировать вероятность возникновения ущерба относительно времени эксплуатации, количества УБ, применяемых средств защиты информации, заданного грифа секретности информации и количества попыток реализации УБ.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Камаев, В.А., Михнев, И.П., Сальникова, Н.А. Влияние гамма-фона помещений Волгоградской области на индуцирование рака // Известия Волгоградского государственного технического университета. 2015. №14(178). С. 60–63.
2. Михнев, И.П., Михнева, С.В. Природные радионуклиды как источник фонового облучения населения Нижневолжского региона // В сборнике: Образование и наука: современные тренды. Коллективная монография. Сер. «Научно-методическая библиотека» / гл. ред. О. Н. Широков. Чебоксары, 2018. С. 151–166.
3. Kamaev, V.A., Mikhnev, I.P., Salnikova, N.A. Natural Radionuclides as a Source of Background Irradiation Affecting People Inside Buildings. // В сборнике: Procedia Engineering 2. Сер. «2nd International Conference on Industrial Engineering, ICIE 2016» 2016. С. 1663–1672.
4. Mikhnev, I.P., Salnikova, N.A., Lempert, M.B. Research of Activity of Natural Radionuclides in Construction Raw Materials of the Volgograd Region. // Solid State Phenomena. 2017. Т. 265 SSP. С. 27–32.
5. Михнев, И.П., Сальникова, Н.А. Информационная безопасность спектрометрических систем при определении радиационных характеристик в помещениях Волгоградской области // Известия Волгоградского государственного технического университета. 2015. № 13(177). С. 109–113.
6. Михнев, И.П. Информационная безопасность в современном экономическом образовании // Международный журнал прикладных и фундаментальных исследований. 2013. № 4. С. 111–113.
7. Лемперт, М.Б., Михнев, И.П., Сальникова, Н.А. Биологическое воздействие ионизирующих излучений на состояние здоровья населения // В сборнике: Экологические и медицинские проблемы городских экосистем и пути их решения. Материалы региональной научно-практической конференции, посвященной Году экологии в Российской Федерации в 2017 году. 2017. С. 159–164.
8. Сальникова, Н.А., Михнев, И.П. Проведение аттестации знаний студентов с помощью компьютерного тестирования // Известия Волгоградского государственного технического университета. Серия: Новые образовательные системы и технологии обучения в вузе. 2007. Т. 4. № 7 (33). С. 182–184.
9. Доктрина информационной безопасности РФ (утв. Указом Президента РФ от 5 декабря 2016 г. № 646). «Собрание законодательства РФ», 12.12.2016, № 50, ст. 7074.

10. Российская газета. 15.02.2017 г. Совбез: Число кибератак на РФ за 2016 год выросло втрое [Электронный ресурс]. – Режим доступа : <https://rg.ru/2017/02/15/sovbez-chislo-kiberatak-na-rf-za-2016-god-vyroslo-vtroe.html> (дата обращения: 20.04.2018).

11. Сидякин, П.А., Сидельникова, О.П., Козлов, Ю.Д., Михнев, И.П., Малый, В.Т. Материалы для снижения гамма-фона и концентрации радона в помещениях // М.: Строительные материалы. 1998. № 8. С. 26–27.

12. ГОСТ Р 50922-2006. Национальный стандарт Российской Федерации. Защита информации. Основные термины и определения. М.: Стандартинформ. 2008. 8 с.

13. Михнев, И.П., Михнева, С.В., Сальникова, Н.А. Информационная безопасность в Российской Федерации: современность и перспективы развития // В сборнике: Образование и наука: современные тренды. Коллективная монография. Сер. «Научно-методическая библиотека» / гл. ред. О. Н. Широков. Чебоксары, 2018. С. 103–112.

УДК 004.056.5

И. П. Пересыпкин, Н. П. Садовникова, Д. В. Козлов

АНАЛИЗ ПОДХОДОВ И ТРЕБОВАНИЙ К СИСТЕМЕ БЕЗОПАСНОСТИ POS-СЕТИ

Волгоградский государственный технический университет

freeman_33@mail.ru, sadovnikova@vstu.ru, 1mrdiko4@gmail.com

В статье рассмотрены существующие технологии для реализации системы контроля безопасности функционирования POS-сетей и описаны наиболее распространенные системы администрирования сетей POS-устройств. Предложена концепция системы безопасности POS-сети.

Ключевые слова: POS-сети; информационная безопасность, мониторинг; SIEM-система.

I. P. Peresypkin, N. P. Sadovnikova, D. V. Kozlov

ANALYSIS OF APPROACHES AND REQUIREMENTS TO SECURITY SYSTEM OF POS-NETWORK

Volgograd State Technical University

In the article existing technologies for the implementation of the security control system for the functioning of POS networks and describes the most common systems of administration of POS-devices networks. Concept of the POS network security system is offered.

Keywords: POS-networks; information security, monitoring; SIEM-system.

В настоящее время безналичные платежи составляют большую часть платежного оборота. Безналичный денежный оборот в развитых странах составляет 80 % совокупного платежного оборота, и только 20 % приходится на наличные расчеты, а в некоторых странах это соотношение еще больше. В России это соотношение составляет около 60 и 40 % [1]. В повседневной жизни для осуществления безналичных платежей широко применяются POS-устройства. Такие устройства работают с платежной информацией пользователей, которая представляет интерес для злоумышленников. Информация наиболее уязвима при проведении платежных операций. Для своевременной реакции на угрозы со стороны злоумышленников необходима надежная система безопасности, основанная на мониторинге и анализе событий Pos-системы в режиме реального времени.

Системы мониторинга POS-сетей

Согласно стандарту безопасности платежных приложений (PA-DSS) к системам мониторинга предъявляются следующие требования [2]:

1. Сервис мониторинга должен обеспечивать централизованное отображение всех событий в режиме реального времени (основание: PA-DSS 3.2 пункт 4.4).

2. Сервис мониторинга должен регистрировать как минимум следующие параметры для каждого события: идентификатор пользователя; тип события; дата и время; успешным или неуспешным было событие; источник события; идентификатор или название данных, системного компонента или ресурса (основание: PA-DSS 3.2 пункт 4.3).

3. Должно быть обеспечено информирование обо всех ошибках, произошедших на POS-устройствах.

Платежные терминалы являются наиболее уязвимым местом в системе обеспечения безопасности данных клиента, пользующегося услугами POS-сети. Именно через них в систему поступают данные платежных карт в открытом виде. Несмотря на то, что платежные терминалы достаточно серьезно защищены на уровне физической архитектуры и операционной систе-

мы, существует потенциальная возможность несанкционированного доступа к данным через приложения, которые взаимодействуют с терминалом. PA-DSS предъявляет строгие требования к таким приложениям, и система мониторинга должна осуществлять контроль за их исполнением.

Для управления POS-сетями сегодня применяются системы администрирования сетей POS-устройств, которые реализуют следующие функции [3]:

- 1) ведение учета имеющихся POS-устройств и их конфигурации;
- 2) дистанционное обновление ПО на POS-устройствах;
- 3) дистанционное конфигурирование параметров работы POS-устройств;
- 4) дистанционная доставка ключей шифрования по защищенным каналам. При этом отдельно выделяется задача доставки первичного ключа к устройству.

Наиболее популярными на сегодняшний день являются следующие системы: First Data, Networks POS, TransLink.iQ, решения компании INETCO для мониторинга POS и др. Рассмотрим особенности каждой из приведенных систем:

1) First Data (Clover® Apps) – одна из крупнейших систем в области администрирования POS-устройств. Предоставляет обширный инструментарий для администрирования POS-терминалов и периферийных устройств. Поддержка стандарта безопасности инфраструктуры платежных карт PCI [4]. TransArmor от First Data использует комбинацию технологий шифрования и токенизации для защиты и полного удаления данных платежной карты из информационной среды торгового предприятия, то есть системы не содержат фактические номера карт в обрабатываемых транзакциях. Благодаря этому решению, отпадает необходимость хранить данные карт, вместо них используется случайный номер, называемый «токеном»;

2) TransLink.iQ обладает следующими возможностями: управление сетью POS-терминалов; реализация операций с карточками в POS-терминалах; безопасный и оперативный перевод операций по карточкам для банков-приобретателей; интеграция с внешними системами; дистанционное обновление ПО POS-терминалов; мониторинг сети POS-терминалов. Данное ПО было разработано в соответствии с требованиями, предъявляемыми международными компаниями VISA International и MasterCard

International, а также соответствует стандартам EMV. С 2007 года компания сертифицирована согласно требованиям уровня 1 PCI;

3) Networks POS – обеспечивает управление POS-терминалами в режиме реального времени. Является самым быстрорастущим поставщиком аппаратных и программных решений для точек продаж (POS) в индустрии розничной торговли. Производитель предполагает выбор аппаратного и программного обеспечения, осуществляет внедрение и обучение сотрудников;

4) Mercury Processing Services International – предлагает различные и модифицируемые POS-услуги – от выбора POS-терминала и лизинга (услуги по закупкам) и обработки транзакций до разработки и интеграции программного обеспечения, настройки сервисов и обучения. Данная система работает на проверенных и гибких платформах и имеет возможность поддержки: нескольких поставщиков; различных типов связи (IP, Lease Line IP, Dial-UP / GSM); различных протоколов авторизации; интеграции с торговыми кассовыми системами; мультивалютных транзакций между POS-серверами; бесконтактных транзакций (включая решение HCE);

5) INETCO Solutions for POS Monitoring and Retail Application Management. INETCO предоставляет решения для мониторинга POS-сетей и анализа их каналов связи, позволяющие фиксировать данные платежных транзакций в режиме реального времени и обеспечивать централизованное отображение всех POS и других розничных приложений с помощью легко настраиваемых информационных панелей, отчетов по запросу и аналитики взаимодействия с потребителями.

Описанные системы позволяют эффективно управлять инфраструктурой POS-сети. Однако функционал большинства систем не предусматривает мониторинга событий связанных с информационной безопасностью. В таких системах, как TransLink.iQ и система мониторинга INETCO реализованы функции мониторинга POS-сетей. Но весь мониторинг в этих системах сводится к наблюдению за аппаратными ресурсами, нагрузкой на информационную сеть, а также отображению интегральных оценок по осуществленным транзакциям (количество и скорость обработки).

SIEM-системы

Похожими по назначению являются SIEM-системы. SIEM (Security Information and Event Management) – решения, осуществляющие мо-

мониторинг информационных систем, анализ событий безопасности в реальном времени, информацию о которых получают от сетевых устройств, средств защиты информации, ИТ-сервисов, инфраструктуры систем и приложений, и, таким образом, оказывают помощь в обнаружении инцидентов ИБ. SIEM распространяются в виде как аппаратных устройств, так и программного обеспечения или услуг и применяются для сбора и обработки событий безопасности, оповещения операторов, генерации отчетов и визуализации нарушений ИБ.

Наиболее популярными являются следующие SIEM системы [5]:

1. Hewlett Packard Enterprise (HPE) ArcSight – наиболее популярная на российском рынке SIEM-система. Arcsight ESM обладает полным набором возможностей SIEM: обнаруживает аномалии в поведении пользователей; анализирует DNS-трафик и обеспечивает прозрачность и наглядность ИТ-инфраструктуры, что помогает обнаружить сетевые уязвимости еще до того, как ими воспользуются злоумышленники; поддерживает интерактивную базу знаний угроз и позволяет клиентам обмениваться сведениями о способах их обнаружения и ликвидации.

2. McAfee Enterprise Security Manager (ESM) поставляется в качестве физического и виртуального устройств и программного обеспечения. Три основных компонента, входящие в состав SIEM – ESM, Event Receiver и Enterprise Log Manager, которые могут быть развернуты вместе как один экземпляр или отдельно для распределенных или крупномасштабных сред.

Enterprise Security Manager широко применяется в промышленных системах управления (ICS) а также устройствах диспетчерского управления и сбора данных (SCADA); позволяет осуществлять интеграцию со сторонними технологиями без использования специального API. Этот подход дает возможность для использования ESM в качестве платформы SIEM. McAfee Global Threat Intelligence позволяет увеличить возможности SIEM-системы Enterprise Security Manager, путем добавления обновляемого в реальном времени источника информации об угрозах.

3. MaxPatrol SIEM – продукт российской компании Positive Technologies. MaxPatrol SIEM основывается на актив-ориентированном подходе, обеспечивающим устойчивость функционирования системы к изменениям в ИТ-инфраструктуре компании. Правила корреляции на-

значаются не на конкретные IP-адреса или их набор, а на динамическую группу активов (компьютеров, серверов и др.), которая формируется по выбранным признакам и состав которой может изменяться с развитием сети.

4. КОМРАД – SIEM-система, разработанная российской компанией НПО «Эшелон». КОМРАД дает возможность централизованно производить мониторинг событий ИБ, обнаруживать инциденты ИБ, оперативно реагировать на возникающие угрозы, выполнять требования, предъявляемые регуляторами к различным классам информационных систем.

Таким образом, SIEM-системы позволяют выполнять функции мониторинга информационной безопасности системы, регистрируя различные события безопасности, информируя о подозрительной активности для своевременной реакции на возможные угрозы. Оперативное оповещение, полнофункциональная подсистема визуализации, выявление аномалий на основе анализа поведения пользователей, приоритизирование инцидентов в соответствии с важностью актива, анализ трафика и обеспечение полной видимости ИТ-инфраструктуры.

Вместе с тем существует ряд особенностей POS-сети, которые требуют включения дополнительных функций для обеспечения безопасности. Прежде всего, необходим сервис оповещения операторов об инцидентах, зафиксированных в процессе работы POS-сети. Кроме того, необходимы функции для анализа поведения пользователей. Это позволит избежать «атак изнутри» и потерь критически важной информации. Система мониторинга должна позволять добавлять новые типы анализируемых инцидентов в процессе работы системы, без остановки ее работы. Автоматическое построение топологии сети с реализацией полнотекстового поиска по элементам POS-сети и ее событиям позволит оператору наблюдать обобщенное текущее состояние POS-сети в реальном времени, учитывая все имеющиеся взаимодействия между ее структурными элементами. Необходимым требованием также является высокая производительность системы на большом количестве элементов (в районе 5–10 тысяч).

На основе проведенного анализа были определены концепция системы безопасности POS-сетей.

Для эффективного контроля безопасности Pos-сети система администрирования должна быть объединена с сервисом мониторинга.

Необходимо реализовать возможность удаленного обновления и настройки и контроля параметров сети.

В процессе контроля безопасности должны быть реализованы следующие функции:

- получение и обработка информации об объекте наблюдения (POS оборудование, его состояние, конфигурация и выполняемые им действия) в режиме реального времени;

- обработка полученной информации с целью обнаружения инцидентов, свидетельствующих о возможной реализации угрозы безопасности в системе. Для поиска инцидентов используются продукционные модели, методы сопоставления с известными прецедентами и пр. [6];

- ранжирование инцидентов по уровню угрозы;

- выполнение определенного набора действий в соответствии с уровнем угрозы и типом инцидента. В первую очередь это оповещение оператора о произошедшем событии. Также возможны такие действия, как рассылка уведомлений другим ответственным субъектам, полное или частичное блокирование выполнения некоторых функций (как отдельного устройства, так и всей системы) и др.

Интеграция системы администрирования и мониторинга позволит повысить безопасность Pos-сети, за счет своевременного реагирования на подозрительную активность в POS-

сети и предотвращения возможных угроз. Анализ данных мониторинга даст возможность обнаружить потенциальные уязвимости и принять меры по их устранению.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. WWW.Grandars Безналичный денежный оборот [Электронный ресурс]. Режим доступа : <http://www.grandars.ru/student/finansy/elektronnye-platezhi.html> (дата обращения: 12.05.2018)
2. Стандарт безопасности платежных приложений (PA-DSS), версия 3.0.
3. Анализ существующих технологий для реализации облачного мониторинга POS-сетей / О.А. Авдюк, Л.В. Дружинина, Д.В. Козлов, И.А. Тарасова // Материалы X Междунар. конф. «Управление развитием крупномасштабных систем MLSD'20П (2–4 октября 2017 г., Москва, Россия). Т. 2, секция 11. – М.: ИПУ РАН, 2017.
4. The PCI Security Standards [Электронный ресурс]. Режим доступа: https://www.pcisecuritystandards.org/pci_security/ (дата обращения: 30.05.2018).
5. Обзор мирового и российского рынка SIEM-систем 2017. Anti-malware.ru [Электронный ресурс]. Режим доступа : https://www.anti-malware.ru/analytics/Market_Analysis/overview-global-and-russian-market-siem#part1 (дата обращения: 15.06.2018).
6. Предотвращение мошенничества в системах электронных платежей на основе мониторинга и анализа событий в POS-сетях = Fraud prevention in the system of electronic payments on the basis of monitoring and analysis events in POS-networks [Электронный ресурс] / Д.В. Козлов, Н.П. Садовникова, Л.В. Дружинина, Д.В. Петрова // International Journal of Open Information Technologies : электрон. журнал (Москва). – 2017. – Vol. 5, № 11. – P. 15–20.

УДК 004.942

М. Ю. Умницын

ПОДХОД К ПОЛУНАТУРНОМУ АНАЛИЗУ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННОЙ СИСТЕМЫ

Волгоградский государственный университет

umnitsyn@volsu.ru

В статье предлагается подход к полунатурному анализу защищенности информационной системы с использованием многоагентной системы для обеспечения полунатурности. Описана архитектура многоагентной системы, ее компоненты. Приведена классификация агентов по назначению. Разработанная многоагентная система может быть использована при полунатурном тестировании на проникновение на информационную систему.

Ключевые слова: анализ защищенности, информационная система, тестирование на проникновение, многоагентные системы.

М. Y. Umnitsyn

APPROACH TO SEMI-NATURAL SECURITY EVALUATION OF INFORMATION SYSTEM

Volgograd State University

In the article the scaled-down (seminatural) approach to security evaluation is presented. The approach is based on the use of a multi-agent system for the implementation of semi-natural stand. Multi-agent system architecture and components are described. The classification of agents by purpose is given. The developed multi-agent system can be used for semi-natural penetration testing in the information system.

Keywords: security evaluation, information system, penetration testing, multiagent systems.

Одна из основных проблем проведения инструментального анализа защищенности информационных систем (ИС) связана с необходимостью прогнозирования негативных факторов, возникающих при проведении тестирования и влияющих на функционирование ИС.

В большинстве случаев отказываются от исследований, потенциально приводящих к отказу в обслуживании или нарушениям работоспособности организации [1]. Упрощение технологии анализа защищенности приводит к необоснованному завышению показателей защищенности т. к. реальные атаки существенно отличаются по применяемым методам от модельных [2].

Также возникают ограничения, связанные с недостатками используемых методов исследования и выявления уязвимостей [3–5]:

- неполнотой базы выявляемых уязвимостей;
- ограниченными возможностями выбранного сканера уязвимостей;
- отсутствием инструментов анализа полученных привилегий (последствий) при эксплуатации уязвимостей.

В связи с этим, существует необходимость совершенствования технологии тестирования безопасности ИС и поиска новых решений для повышения эффективности данного процесса.

В данной статье предлагается подход для полунатурного инструментального анализа защищенности, позволяющего повысить точность результатов исследования. Также, за счет полунатурности реализуется возможность симуляции потенциально деструктивных воздействий на информационную систему.

Для проведения инструментальных исследования предлагается использовать многоагентную систему, архитектура которой представлена на рис. 1.

В качестве многоагентной среды выбрана BDI-совместимая среда Jason, реализующая исполняющую среду агентов AgentSpeak [6, 7].

Основными компонентами многоагентной системы являются:

Модуль имитации злоумышленных воздействий включает агентов-злоумышленников, реализующих функции проверки информационной системы на уязвимости.

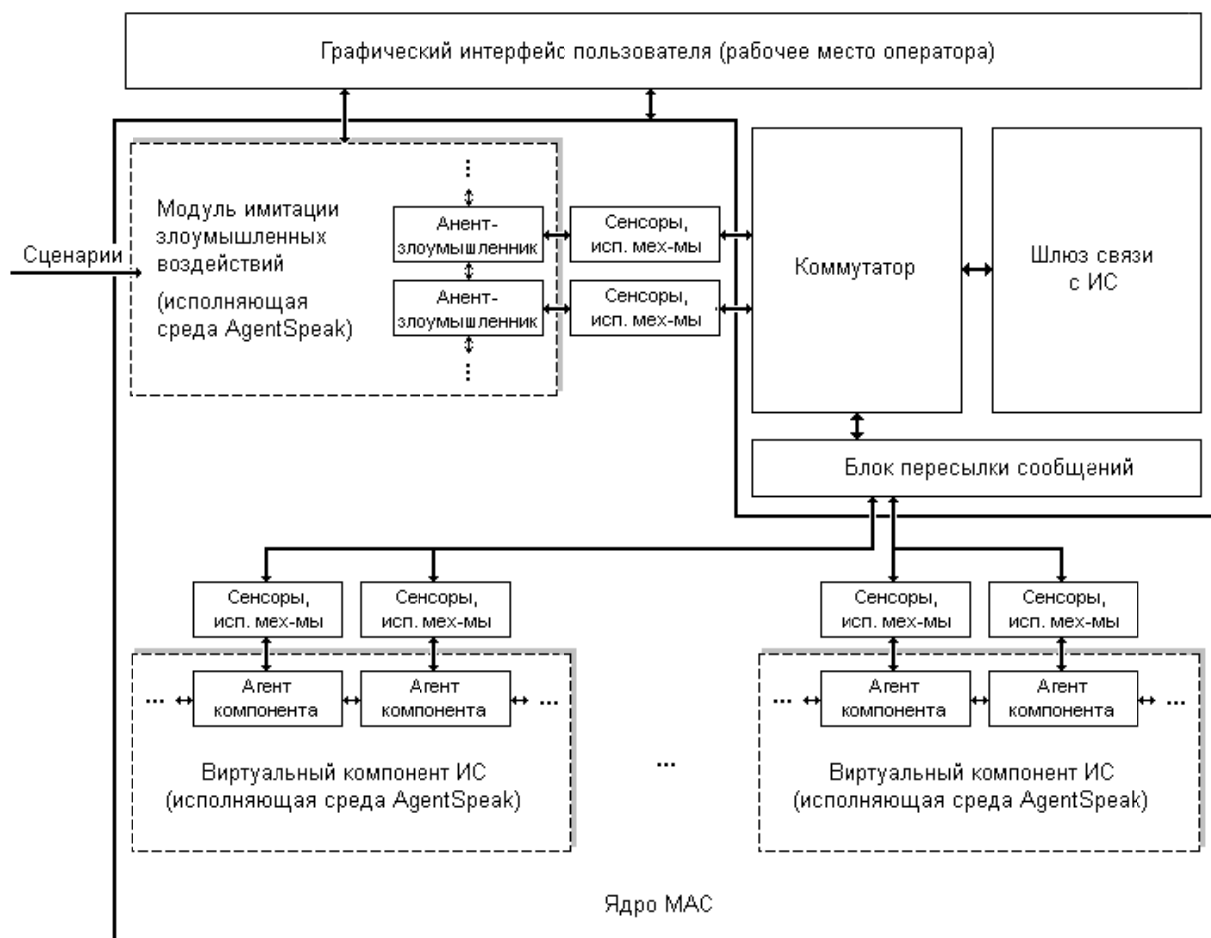


Рис. 1. Архитектура многоагентной системы

Коммутатор определяет, какие компоненты информационной системы являются реальными, а какие имитируются. Если запрос адресуется реальному компоненту системы, то он передается в шлюз связи. Если же запрос предназначен компоненту, наличие которого в системе лишь имитируется, то коммутатор отправляет этот запрос в блок пересылки сообщений.

Шлюз связи с ИС – реализует различные методы взаимодействия с компонентами информационной системы и отвечает за доставку сообщений.

Блок пересылки сообщений управляет пересылкой сообщений между компонентами модуля виртуальных компонентов информационной системы. Данный модуль, совместно с модулем коммутатора позволяет реализовать полунатурный подход.

Модуль виртуальных компонентов информационной системы включает в себя множество моделируемых компонентов информационной системы, которые представлены отдельными агентами.

Модуль имитации злоумышленных воздействий и виртуальные компоненты информационной системы являются условными модулями. Факти-

чески они представлены множеством интеллектуальных агентов, которые выполняют разные задачи, но функционируют в рамках общей исполняющей среды, называемой ядром МАС.

Агенты могут взаимодействовать друг с другом и средой через общую коммуникационную шину, к которой подключен коммутатор. Агент, посылая команду, не знает, какой объект и каким образом обработает ее – его интересует лишь данные, полученные в результате обработки команды. В комплект среды Jason включен суперкласс Environment, что сводит реализацию ядра к простому определению того, какому агенту какие сенсоры (Percepts) и исполнительные механизмы (Executors) соответствуют [7].

Таким образом, сообщения агента могут быть обработаны следующими способами:

- 1) перенаправлены агентам в блок виртуальных компонентов ИС, моделирующим компонентам информационной системы;
- 2) перенаправлены на реальные узлы информационной системы.

Отличие предлагаемого подхода от классического подхода инструментального анализа показано на рис. 2.



Рис. 2. Особенности инструментального и полунатурного подхода

На вход модуля имитации злоумышленных воздействий подается множество сценариев злоумышленника, на основе которых строятся программы поведения интеллектуальных агентов, атакующих исследуемую информационную систему. Сценарии формируются в виде XML-файла.

Данный подход можно описать следующей последовательностью шагов:

1. Задание исходной информации

1.1. Формируется граф сценариев злоумышленника.

1.2. Определяются моделируемые (виртуальные) компоненты информационной системы.

1.3. Генерируется каркас многоагентной системы, согласно графу сценариев.

1.3.1. Генерируется декларативное описание агентов.

1.3.2. Генерируется множество планов и иницирующих событий агентов.

1.3.3. Прототипируются исполнительные механизмы агентов.

1.4. Детализируются планы и исполнительные механизмы агентов с учетом выбранных

виртуальных компонентов и программного обеспечения инструментального анализа защищенности.

2. Проведение имитационных исследований.

2.1. Проведение инструментального анализа защищенности информационной системы с использованием разработанной многоагентной системы.

2.2. Анализ результатов исследований.

2.2.1. Анализ достигнутых злоумышленником целей.

2.2.2. Анализ состояния компонентов информационной системы.

2.3. Корректировка планов проведения эксперимента (виртуальных компонент, перечня и параметров программного обеспечения инструментального анализа защищенности)

2.4. Проведение повторной имитации.

Множество сценариев можно представить в виде графа (рис. 3), в котором:

1) вершины графа – описание состояний информационной системы;

2) дуги графа – изменения состояния ИС, представляющие собой уязвимости ИС и способы их эксплуатации – тактические сценарии агентов-злоумышленников;

3) маршруты $(s_0; s_1; s_3; s_5)$; $(s_0; s_1; s_4; s_5)$; $(s_0; s_2; s_6)$ – стратегические сценарии агентов-злоумышленников.

Соответственно, агенты в модуле имитации злоумышленных воздействия делятся на три типа:

- $\{ag^{ps_{strategy}}\}$ – множество ведущих агентов, которым известны сценарии воздействий на информационную систему. Они выполняют мониторинг состояния информационной системы, определяют какой вершине соответствует текущее состояние и отдают приказы агентам второго множества на эксплуатацию уязвимостей для перевода системы в состояние, соответствующее одной из следующих вершин графа;

- $\{ag^{ps_{tactics}}\}$ – множество ведомых агентов, которым известны способы эксплуатации уязвимостей (т. е. последовательность действий, приводящих к переходу в следующую вершину графа);

- $\{ag^{ps_{lead}}\}$ – агент-координатор. Координирует выполнение сценариев и выполняет их выбор. Обладает информацией о приоритете сценариев и рисках, ассоциированных с каждым сценарием.

Сценарии действий злоумышленников разрабатываются с использованием подхода формальной верификации моделей[8].

Предлагаемый подход обладает следующими преимуществами:

- возможность прозрачного переключения между виртуальными компонентами и реальными объектами информационной системы;

- простота изменения целей злоумышленника в многоагентной системе. Возможность проведения анализа при различных исходных знаниях и расположения злоумышленника;

- модульность многоагентной системы.

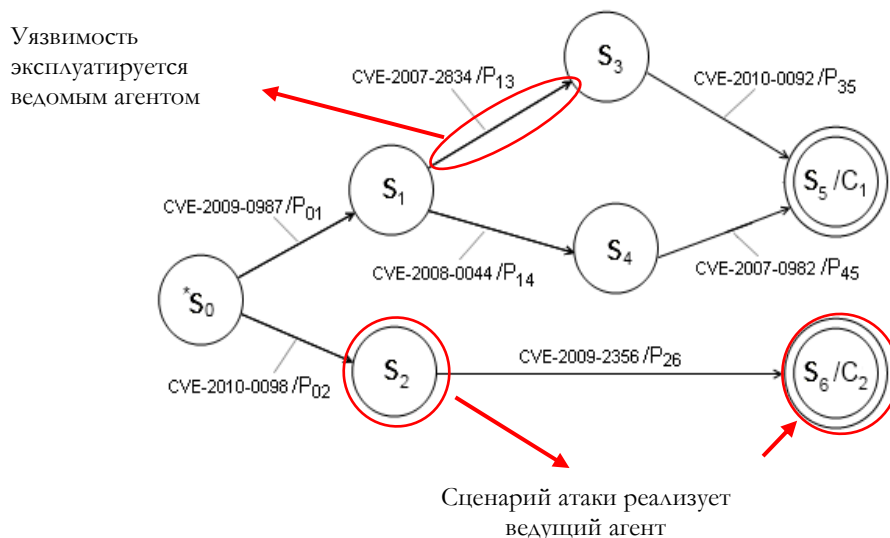


Рис. 3. Граф сценариев злоумышленника

Применение полунатурного подхода при анализе защищенности позволяет учесть усло-

вия функционирования информационной системы, минимизировав риски нарушения ее ра-

боты, при этом повысив эффективность процесса анализа защищенности и точность результатов при сравнимых затратах.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Ермаков, Кирилл. «Сыграем в игру: еще один способ тестирования на проникновение» / ZeroNights 2015. URL: <http://2015.zeronights.ru/assets/files/31-Ermakov.pdf> (дата обращения: 27.02.2017).
2. Янкин, Андрей. «Penetration Test пополючим» / Jet Info, №3-4, 2017. URL: <http://www.jetinfo.ru/stati/penetration-test-po-polochkam> (дата обращения: 27.02.2017).
3. Захаров, А. Пентест как предчувствие [Электронный ресурс] // SecurityLab.ru-2016. – Режим доступа : <http://www.securitylab.ru/analytics/483220.php> (дата обращения: 27.02.2017).
4. Chiem Trieu Phong. A Study of Penetration Testing Tools and Approaches // School of computing and mathematical sciences (MCIS). Auckland University of Technology. – 2014. – 125 с. Режим доступа : <http://aut.researchgateway.ac.nz/bitstream/handle/10292/7801/ChiemTP.pdf?sequence=3> (дата обращения: 27.02.2017).
5. Sven Tuerpe; Jörn Eichler. Testing Production Systems Safely: Common Precautions in Penetration Testing. Testing: Academic and Industrial Conference-Practice and Research Techniques (TAIC PART 2009), 4-6 September 2009, Windsor, United Kingdom. – 2009 – 5 с. DOI: 10.1109/TAICPART.2009.17. Режим доступа : https://testlab.sit.fraunhofer.de/downloads/Publications/tuerpe_eichler_Testing_production_systems_safely_-_Common_precautions_in_penetration_testing_TAIC_PART_2009.pdf (дата обращения: 27.02.2017).
6. Georgeff, M., Pell B., Pollack M. [и др.] The Belief-Desire-Intention model of agency, Materials of Cognitive Modeling and Multi-Agent Interactions Conference. CRC Press, 2005. С. 300–310.
7. Bordini R.H., Hubner F.J., Wooldridge M. Programming multi-agent systems in AgentSpeak using Jason. Wiley, 2007, 294 с.
8. Умницын, М.Ю., Садовникова, Н.П. Формализация построения сценариев злоумышленных воздействий на информационную систему // Известия Волгоградского государственного технического университета. 2016. № 11 (190). С. 72–75.

РОБОТОТЕХНИКА

УДК 629.3

К. Ю. Лепетухин¹, А. В. Малолетов^{1, 2}, Е. С. Брискин¹

О КРИТЕРИЯХ ОПТИМИЗАЦИИ ДВИЖЕНИЯ СЕКЦИЙ ДОЖДЕВАЛЬНОЙ МАШИНЫ КРУГОВОГО ДЕЙСТВИЯ ПРИ ОБРАБОТКЕ ПОЛЕЙ СЛОЖНОЙ ФОРМЫ*

¹ Волгоградский государственный технический университет

² Университет Иннополис

dtm@vstu.ru

Рассмотрены и обоснованы критерии оптимизации законов движения подвижных опор дождевальной машины кругового действия при обработке полей сложной формы. Приведены примеры моделирования движения дождевальной машины и расчеты различных критериев оптимальности.

Ключевые слова: дождевальная машина кругового действия, управление движением, критерии оптимальности, обработка некруглых полей, модели складывания секций.

K. Y. Lepetukhin¹, A. V. Maloletov^{1, 2}, E. S. Briskin¹

ON THE CRITERIA OF OPTIMIZATION OF CENTER-PIVOT IRRIGATION MACHINE SECTIONS MOVEMENT AT PROCESSING THE FIELDS OF A COMPLEX FORM

¹ Volgograd State Technical University

² Innopolis University

The criteria of optimization of movement control of center-pivot irrigation machine towers for processing fields of complex form are considered and justified. The examples of modelling the movement of irrigation machine and calculation of various criteria of optimization are given.

Keywords: center-pivot irrigation machine, movement control, criteria of optimization, processing non-circular fields, models of center-pivot irrigation machine folding.

Введение

Дождевание является одним из перспективных способов полива. Дождевальные машины кругового действия используются по всему миру. В их состав входят неподвижная центральная опора, секции трубопровода, шарнирно соединенные между собой, установленные на

подвижные самоходные опоры. Например, на рис. 1 показана машина дождевальная электрифицированная кругового действия (МДЭК) «Водолей-А» производства ООО «АГИС Инжиниринг», разработанного в сотрудничестве с ФГБНУ ВНИИ «Радуга».



Рис. 1. Дождевальная машина кругового действия

Система управления дождевальной машины поддерживает постоянный развернутый угол между секциями, что обуславливает траектории движения опор в виде концентрических окружностей или их частей, а форму обрабатываемого поля, соответственно в виде круга или сектора круга.

На рис. 2 представлена фотография из космоса (полученная в сервисе GoogleMaps) группы круглых полей, на которых видно, что между полями находятся достаточно большие участки необрабатываемой земли. Актуальность проблемы, например, подтверждается [1], где она вынесена отдельным пунктом в недостатки.



Рис. 2. Космическая фотография круглых полей. Участки круглой формы – орошаемые с помощью дождевальных машин поля, участки между кругами – необрабатываемые

Для уменьшения потерь площади используются различные технические решения. Среди которых можно отметить ипподромные дождевальные машины, представляющие собой комбинацию фронтальных и круговых дождевальных машин [2], системы полива углов [3, 4, 5].

В работах [6, 7] показано, что изменением угла между секциями можно обрабатывать поля, форма которых отличается от круга. Для минимизации потерь полезной площади, непосредственно связанных с работой дождевальной машины (колеи), целесообразно, чтобы траектории движения опор секций оставались неизменными при каждом обходе поля, а поиск законов управления их движением вести на основе различных критериев качества.

В качестве критериев оценки качества полученных законов могут рассматриваться:

- критерий минимума энергозатрат;
- критерий максимума коэффициента использования земли;
- критерий минимума относительного угла поворота секций;
- критерий минимума суммарной длины следовых дорожек;

– критерий минимума коэффициента перекрытий территорий;

– критерий минимума количества звеньев.

В работе [7] представлено решение задачи оптимизации по критерию минимума энергозатрат, понимаемых как энергозатраты на преодоление сил сопротивления, для модели дождевальной машины в виде непрерывно деформируемой кривой. В работе [6] показан вариант траекторий при движении дождевальной машины при решении задачи оптимизации по критерию минимума энергозатрат для модели дождевальной машины в виде шарнирно сочлененных недеформируемых секций при обработке квадратного поля при минимальном возможном угле между секциями 0° . А в работе [8] рассматривается управление движением секций машины при равномерном складывании.

Описание критериев качества

Независимые критерия качества, как и решения, полученные на их основе, находятся в противоречии друг с другом. Поэтому на практике необходимо искать компромиссные решения, учитывающие различные критерии в зависимости от требований к обработке того или иного поля [9, 10].

1. Критерий минимума энергозатрат

Критерий I_1 требует нахождения такого закона движения самоходных подвижных опор дождевальной машины, при котором энергия, затрачиваемая на перемещение дождевальной машины (сумма энергий A_i для каждой i -й из n тележек), будет минимальной:

$$I_1 = \sum_{i=1}^n A_i \rightarrow \min. \quad (1)$$

Расчет энергозатрат может вестись разными способами. В самом простом случае под энергозатратами понимается работа на преодоление сил сопротивления движению. Например, полагая, что в процессе движения на каждую опору действует одинаковая сила [7], решение задачи оптимизации сводится к обеспечению минимума критерия, формально совпадающего с критерием минимума суммарной длины следовых дорожек.

2. Критерий максимума коэффициента использования земли

Критерий I_2 требует нахождения такого закона движения самоходных подвижных опор секций дождевальной машины, при котором отношение площадей обрабатываемого участка S_p и сельскохозяйственного угодья S_w , предназначенного для обработки данной дождевальной машиной, было максимальным.

$$I_2 = \frac{S_P}{S_W} \rightarrow \max. \quad (2)$$

В связи с тем, что одним из недостатков традиционных дождевальных машин кругового действия, который должна устранить модернизированная дождевальная машина, является потеря полезной площади между обрабатываемыми полями, критерий коэффициента использования земли является очень важным.

Согласно определению максимальным значением критерия I_2 является 1, что означает равенство обрабатываемой и полной площади (без учета потерь на следовые дорожки). Поля некруглой формы характеризуются среди прочего минимальным и максимальным расстоянием от центральной точки до кромки поля. Для обработки всей площади поля необходимо, чтобы суммарная длина секций была больше или равна максимальному радиусу обрабатываемого поля, однако достаточность будет определяться законами управления движением опор дождевальной машины и ее характеристиками.

3. *Критерий минимума относительного угла поворота секций*

Критерий I_3 требует нахождения такого закона движения самоходных подвижных опор секций дождевальной машины, при котором среднее квадратичное отклонение угла между секциями от развернутого было минимальным.

$$I_3 = \frac{1}{2\pi(n-1)} \sum_{i=1}^{n-1} \int_0^{2\pi} \Delta \alpha_i^2 d\varphi \rightarrow \min, \quad (3)$$

где $\Delta \alpha_i$ – угол между i и $i+1$ секциями; n – количество секций; φ – угол поворота текущего радиус-вектора периферийной тележки.

При движении дождевальной машины изгибы в шарнирах создают дополнительное гидравлическое сопротивление. Для уменьшения гидравлических потерь часть соединений между секциями может быть заменена на традиционные, поддерживающие только минимальные отклонения от соосности секций трубопровода.

4. *Критерий минимума суммарной длины следовых дорожек*

Критерий I_4 требует нахождения такого закона движения самоходных подвижных опор секций дождевальной машины, при котором суммарная длина траекторий движения всех i из n опор дождевальной машины была минимальной:

$$I_4 = \sum_{i=1}^n L_i \rightarrow \min. \quad (4)$$

При движении любой дождевальной машины на почве образуется колея, что уменьшает полезную площадь обрабатываемого поля. В колеях происходит накопление воды, что приводит к ускоренной эрозии почвы, умень-

шению сцепных свойств почвы, что повышает энергозатраты на перемещение дождевальной машины.

5. *Критерий минимума коэффициента перекрытий территорий*

Данный критерий требует нахождения такого закона движения самоходных подвижных опор секций дождевальной машины, при котором отношение суммарной площади пашни, на которой будет невозможно получить урожай по причине движения дождевальной машины S_L к общей площади пашни S_W был бы минимальным:

$$I_5 = \frac{S_L}{S_W} \rightarrow \min. \quad (5)$$

Для обработки некруглых полей дождевальной машиной кругового действия необходимо чтобы текущий радиус-вектор точки на границе поля в каждый момент времени был меньше или равен суммарной длине всех звеньев дождевальной машины. Таким образом, для обработки полей форма которых отличается от круга, необходимо менять текущий радиус обработки, тем самым уплотняя траектории движения опорных тележек. Такое сближение траекторий может привести к тому, что часть поля между ними будет непригодна для использования по назначению.

6. *Критерий минимума количества дополнительных секций*

Для оценки минимума количества секций необходимо за базовое значение взять количество секций в традиционной дождевальной машине, которую можно использовать для обработки круга, вписанного в границы заданного поля. Использование дополнительных секций ведет к увеличению стоимости как самой машины, так и ее обслуживания. Поэтому, важной задачей является минимизация количества дополнительных секций.

В качестве критерия вводится отношение количества дополнительных секций в модифицированной дождевальной машине кругового действия $(n-n_0)$ к количеству звеньев n_0 базовой машины.

$$I_6 = \frac{(n-n_0)}{n_0} \rightarrow \min. \quad (6)$$

Примеры расчета критериев для различных вариантов складывания

В рамках исследования были разработаны некоторые алгоритмы управления движением опор дождевальной машины, положенные в основу для программы, рассчитывающей их траекторию. На рис. 3 представлены результаты моделирования движения дождевальной маши-

ны для различных законов управления, отличающихся алгоритмами складывания секций. Для обработки круглого поля (рис. 3, а) рассматривается дождевальная машина, содержащая семь секций, постоянно расположенных вдоль текущего радиус-вектора.

Для обработки квадратного поля, описанного вокруг базового круглого, требуется не ме-

нее чем в $\sqrt{2}$ раз больше секций той же длины, то есть в рассматриваемом случае – 10. В диагональном положении эти секции должны быть почти выпрямлены, а в прочих положениях – складываться. В исследовании рассмотрены различные алгоритмы складывания секций дождевальной машины.

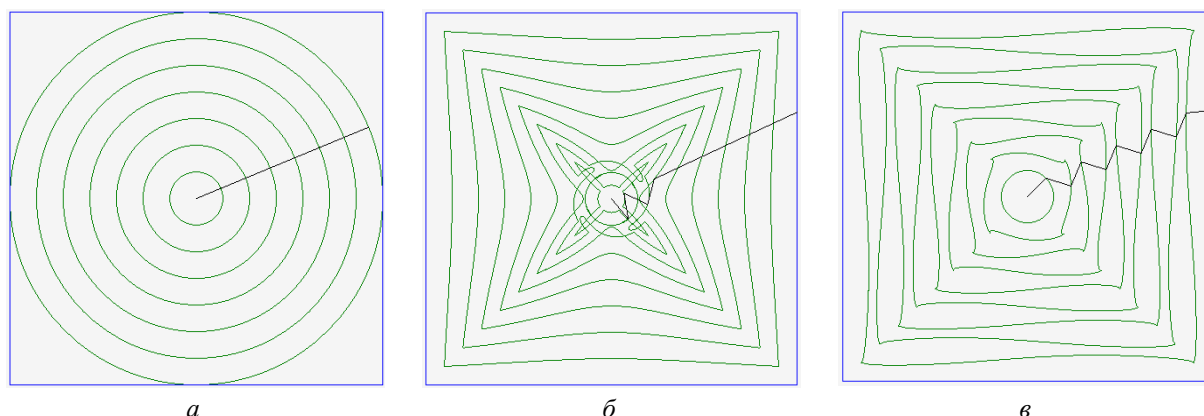


Рис. 3. Траектория движения опор дождевальной машины при обработке круглого поля (а); квадратного поля при алгоритме складывания парами (б) и при равномерном складывании (в)

Алгоритм складывания для закона управления, представленного на рис. 3, б, состоит в последовательном складывании секций дождевальной машины попарно до минимально допустимого угла между ними. Секции, не участвующие в складывании, выстраиваются вдоль текущего радиуса обработки поля. Складывание последующей пары секций начинается после завершения сближения предыдущей.

Алгоритм складывания для закона управления, представленного на рис. 3, в, состоит в равномерном изменении угла между секциями независимо от их положения вдоль текущего радиуса [8].

В приведенной ниже таблице показаны значения рассчитанных критериев для различных вариантов управления дождевальной машиной.

Результаты моделирования

Варианты управления дождевальной машиной	Знаки рассчитанных критериев					
	I_1	I_2	I_3	I_4	I_5	I_6
Круглое поле без складывания	5277	0,79	0	5277	0,038	0
Квадратное поле – складывание парами	9988	1	361	9988	0,056	0,43
Квадратное поле – равномерное складывание	9314	1	476	9314	0,053	0,43

При складывании парами опоры крайних секций дождевальной движутся по направлению перпендикулярному текущему радиусу обрабатываемого поля. Подвижные опоры начальных секций обеспечивают уменьшение текущего радиуса за счет реализации сложных законов управления, включающих реверсирование направления движения и прохождение по некоторым участкам по несколько раз. Кроме того, происходит пересечение траекторий, что увеличивает эрозию почвы и уменьшает полезную площадь поля.

При сравнении траекторий на рис. 3, б и 3, в видно, что подвижные опоры секций дождевальной машины при равномерном складывании движутся по линиям более близким к прямым, чем при складывании парами. При этом, опоры движутся более равномерно и без пересечений.

Из теоретических соображений известно [7], что энергозатраты на движение машины при прочих равных условиях тем меньше, чем ближе к центру находятся в своем движении опорные тележки машины. Однако при рассмотрен-

ном алгоритме складывания вблизи центра па-рами (вариант б) энергозатраты оказываются выше, чем в случае равномерного складывания (вариант в). Это объясняется тем, что складывание вблизи центра требует больших перемещений тележек в радиальном направлении.

Заключение

Разработанные критерии оптимальности позволяют синтезировать законы управления дождевальной машиной кругового действия для обработки полей сложной формы. Также эти критерии можно использовать при оценке существующих систем управления для обработки полей круглой формы.

Увеличение количества секций и внедрение более сложной системы управления позволяет обрабатывать участки, которые до этого были неиспользованными. Законы управления синтезируются на основе минимизации или максимизации критерия качества. Реализация полученных законов приводит к ухудшению значений остальных критериев. Это позволяет ставить задачи многокритериальной оптимизации при рассмотрении задач орошения полей дождевальными машинами рассматриваемого типа.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Berne, D.* Agricultural Irrigation Initiative: Overview of Center Pivot Irrigation Systems [Электронный ресурс]. NorthwestEnergyEfficiencyAlliance. – 2015. – Режим доступа : <https://neea.org/img/uploads/overview-of-center-pivot-irrigation-systems.pdf>, свободный (дата обращения: 22.05.2018).
2. *Фокин, Б.П., Носов, А.К.* Современные проблемы применения многоопорных дождевальных машин. – Ставрополь : Научное издание, 2011. – 80 с.
3. *Кузнецов, П.И.* К вопросу модернизации многоопорных дождевальных машин кругового действия / П.И. Кузнецов // Природообустройство. – М., 2011. № 4. С. 72–76.
4. Многоопорные дождевальные машины / С.Х. Гусейн-заде, Л.А. Перевезцев, В.И. Коваленко, В.Г. Луцкий ; под ред. С.Х. Гусейн-заде. – М.: Колос, 1984. 191 с., ил.]
5. Патент US6085999A, МПК B05B 3/12 Corner irrigation system / Jerry D. Gerdes, Omaha; Dale A. Christensen, Arlington, both of Nebr, Valmont industries, Inc., Valley, Nebr – 1998.
6. *Малолетов, А.В.* Управление движением опор дождевальной машины кругового действия при движении по полям сложной формы / А.В. Малолетов, К.Ю. Лепетухин // Известия Волгоградского государственного технического университета. 2015. № 14(178). С. 143–147.
7. *Лепетухин, К.Ю.* Об оптимальном управлении взаимным расположением секций дождевальной машины кругового действия при обработке некруглых полей / К.Ю. Лепетухин, А.В. Малолетов, Е.С. Брискин // Мехатроника, автоматизация, управление. – 2017. – Т. 18, № 2. – С. 103–107 + 3-я стр. обложки.
8. *Лепетухин, К.Ю.* Об управлении роботизированной многоопорной дождевальной машиной кругового типа действия / К.Ю. Лепетухин, А.В. Малолетов // В книге: 10-я Всероссийская мультиконференция по проблемам управления (МКПУ-2017) // Материалы 10-й Всероссийской мультиконференции. В 3 т. – 2017. – С. 93–95.
9. Опыт разработки и испытаний шагающих опор дождевальной машины / Е.С. Брискин, В.В. Чернышев, В. В. Жогга, А.В. Малолетов // Тракторы и сельхозмашины. – 2011. – № 9. – С. 27–31.
10. *Малолетов, А.В.* Оптимизация структуры, параметров и режимов движения шагающих машин со сдвоенными движителями : моногр. / А.В. Малолетов, Е.С. Брискин. – Волгоград. – 2015. – 175 с.

Редактор РИО, ответственный за выпуск
Л. Н. Рыжих

Компьютерная верстка:
Е. В. Макарова

Свидетельство о регистрации ПИ № ФС77–61113 от 19 марта 2015 г.
Федеральной службы по надзору в сфере связи, информационных технологий
и массовых коммуникаций (Роскомнадзор)

Учредитель:
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Волгоградский государственный технический университет» (ВолГТУ)

Адрес редакции и издателя: 400005, г. Волгоград, просп. В. И. Ленина, 28.
Волгоградский государственный технический университет (ВолГТУ)
Телефон: гл. редактор – (8442) 24-80-00
E-mail: president@vstu.ru

Темплан 2018 г. Поз. № 10 ж. Подписано в печать 06.09.2018 г. Формат 60 x 84 1/8.
Бумага офсетная. Гарнитура Times. Печать офсетная. Усл. печ. л. 14,41. Уч.-изд. л. 14,79.
Тираж 80 экз. Свободная цена. Заказ №

Оригинал-макет и электронная версия подготовлены РИО ИУНЛ ВолГТУ
400005, г. Волгоград, просп. В. И. Ленина, 28, корп. 7.

Отпечатано в типографии ИУНЛ ВолГТУ
400005, г. Волгоград, просп. В. И. Ленина, 28, корп. 7.

К СВЕДЕНИЮ АВТОРОВ

В журнале «Известия высших учебных заведений», серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах», публикуются статьи, которые содержат результаты теоретических и экспериментальных исследований, направленных на совершенствование наземных транспортных и тяговых систем и их элементов, а также на повышение эффективности транспортных операций.

Вопрос об опубликовании статьи или ее отклонении решает редакционная коллегия журнала, которая утверждается ректором университета, ее решение является окончательным. Редколлегия направляет представленный для издания материал на рецензирование.

Рукопись должна быть набрана и сверстана в текстовом редакторе Word и распечатана на лазерном принтере в режиме полной загрузки тонера. Формат бумаги А4 (210×297 мм).

Для ускорения подготовки издания необходимо представлять файлы статей на компакт-дисках (CD) в полном соответствии с распечатанным оригиналом. Диск должен быть вложен в отдельный конверт, на этикетке диска указываются фамилии авторов статьи.

При наборе текста следует соблюдать следующие требования: поля – верхнее – 2,0 см, нижнее – 3,0 см, левое – 2,5 см, правое – 2,5 см; шрифт Times, кегль 14, интервал полупетельный.

Текст набирается с применением автоматического переноса слов, перед знаками препинания (в том числе внутри скобок) пробелы не допускаются, после них ставится один пробел. Разрядка слов не допускается, следует избегать перегрузки статей большим количеством формул, рисунков, таблиц. Для набора символов в формульном редакторе MS Equation (MS Word) использовать установки (Стиль/Размеры) только по умолчанию; рисунки должны быть выполнены в редакторах векторной графики, таких как CorelDRAW или в любом приложении к Word. Допускается сканирование рисунков в программе Microsoft Photo Editor.

Инициалы и фамилия автора (авторов) пишутся над заглавием статьи. Ниже заглавия, перед основным текстом, указывается организация или предприятие, на котором работает автор статьи. В конце статьи ставится подпись автора (на бумажном варианте).

Литературные ссылки должны быть оформлены в соответствии с ГОСТ 7.1–2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления». Библиографический список использованной литературы, составленный в порядке упоминания в тексте, дается в конце статьи; связь с основным текстом осуществляется цифровым порядковым номером в квадратных скобках в строке. Подстрочные ссылки не допускаются.

Иностранные фамилии и термины в тексте следует приводить в русском переводе. В библиографическом списке фамилии авторов, полное название книг и журналов приводится на языке оригинала.

Ссылки на неопубликованные работы не допускаются.

При обозначении единиц физических величин должна применяться Международная система единиц (СИ).

Объем статьи не должен превышать 8 страниц бумаги формата А4, включая таблицы и библиографический список; число рисунков – не более четырех, включая рисунки, помеченные буквами *a*, *b* и т. д. Рекомендуется включать в сборник статьи с авторским коллективом не более четырех человек с участием каждого автора в одной-двух статьях.

Статьи должны представлять сжатое четкое изложение результатов, полученных автором без повторов приводимых данных в тексте статьи, таблицах и рисунках. К статье должны быть приложены: сведения об авторах (полное имя, отчество, фамилия, ученая степень, звание, домашний адрес, номера служебного и домашнего телефонов, E-mail); документация, подтверждающая возможность ее открытого опубликования.