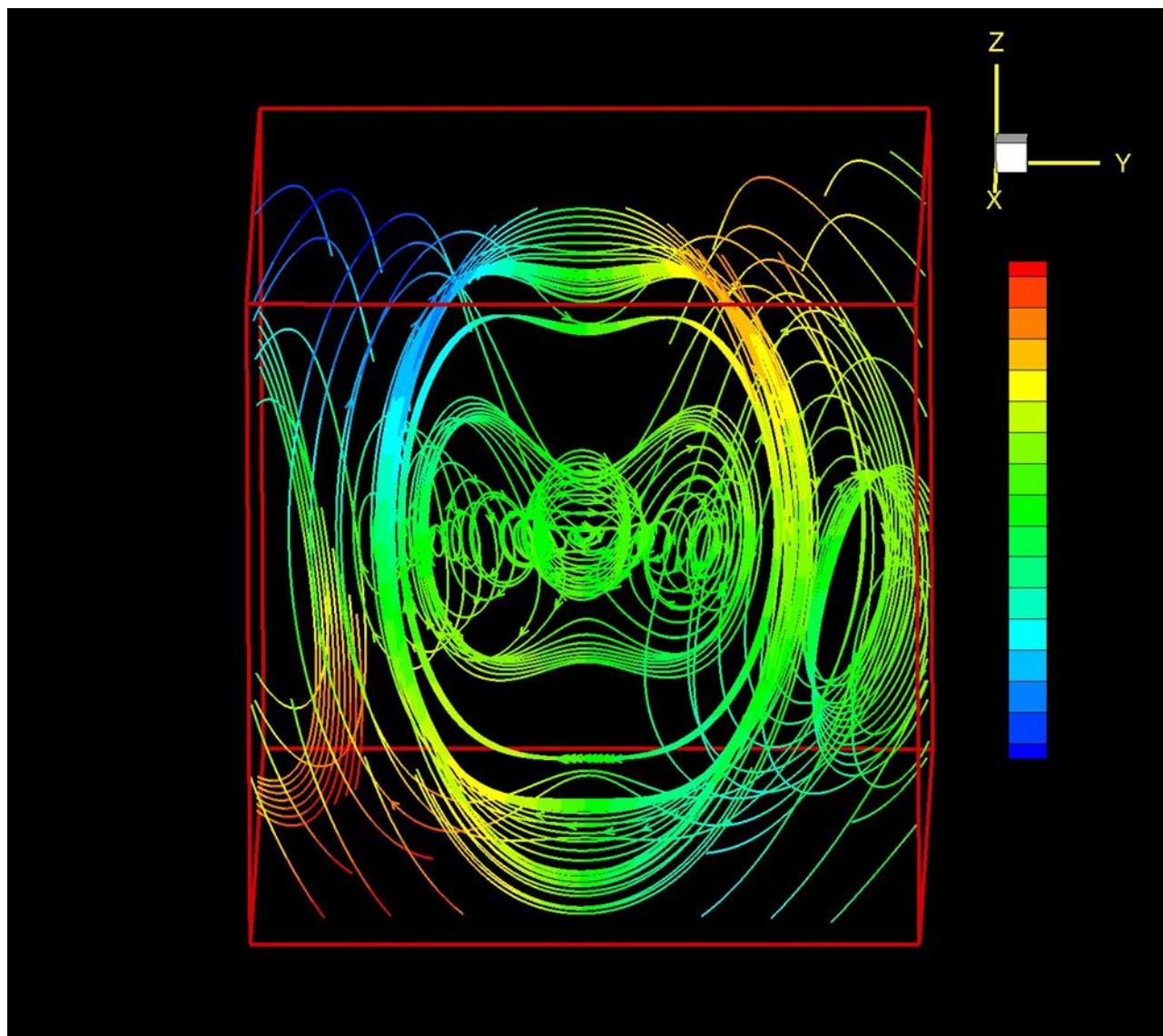




Вестник кибернетики

Proceedings in Cybernetics

№ 4 (48)

12+

2022

**БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
ХАНТЫ-МАНСКИЙСКОГО АВТОНОМНОГО ОКРУГА – ЮГРЫ
«СУРГУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Вестник кибернетики

Proceedings in Cybernetics

№ 4 (48)

Сургут
2022

Учредитель и издатель

Бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры
«Сургутский государственный университет»

Главный редактор

Острейковский Владислав Алексеевич, д. т. н., профессор

Заместитель главного редактора

Тараканов Дмитрий Викторович, к. т. н., доцент

Ответственный секретарь

Брагинский Михаил Яковлевич, к. т. н., доцент

Члены редакционной коллегии:

Советов Б. Я., академик РАН и РАО, д. т. н., профессор
Юсупов Р. М., член-корреспондент РАН, д. т. н., профессор
Абрамов О. В., д. т. н., профессор
Анохин А. Н., д. т. н., профессор
Антонов А. В., д. т. н., профессор
Бурханов Р. А., д. филос. н., профессор
Бушмелева К. И., д. т. н., профессор
Гетман А. Ф., д. т. н., профессор
Григорьев Л. И., д. т. н., профессор
Древс Ю. Г., д. т. н., профессор
Ельников А. В., д. ф.-м. н., профессор
Инютин С. А., д. т. н., профессор
Каштанов В. А., д. ф.-м. н., профессор
Керимов Т. Х., д. филос. н., профессор
Леонов Д. Г., д. т. н., профессор
Малышев Д. С., д. ф.-м. н., профессор
Мельников А. В., д. т. н., профессор
Певзнер Л. Д., д. т. н., профессор
Увайсов С. У., д. т. н., профессор
Цибульский В. Р., д. т. н., профессор
Юрков Н. К., д. т. н., профессор

Выпускающий редактор

Хасанова А. Ш.

Редактор

Манаева Л. И.

Верстка

Мельниковой Е. А.

Переводчик

Петрова А. В.

Рисунок на обложке:

Визуализация магнитогидродинамических расчетов для точного решения внутри сферы
© Галкин В. А., Гореликов А. В.

Решением Высшей аттестационной комиссии с 29 мая 2017 года журнал включен в «Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук».
С 29.03.2022 издание принимает к публикации статьи по следующим научным специальностям: 2.3.1. – системный анализ, управление и обработка информации; 1.2.2. – математическое моделирование, численные методы и комплексы программ; 2.2.2. – электронная компонентная база микро- и нанoeлектроники, квантовых устройств

Журнал зарегистрирован Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций. Свидетельство о регистрации средства массовой информации
Эл № ФС77-83015 от 31.03.2022

Издается с 2002 года. Выпускается 4 раза в год

Адрес редакции:

БУ ВО «Сургутский государственный университет»,
628412, Россия, Ханты-Мансийский автономный округ – Югра, г. Сургут, пр. Ленина, 1
Тел.: (3462) 76-29-88, факс: (3462) 76-29-29, e-mail: proceed.cyber@mail.ru. Сайт: vestcyber.ru

Founder and Publisher
Surgut State University

Chief Editor

Prof. Vladislav A. Ostreikovskiy, Doctor of Sciences (Engineering)

Vice Chief Editor

Dmitry V. Tarakanov, Candidate of Sciences (Engineering), Associate Professor

Executive Editor

Mikhail Ya. Braginsky, Candidate of Sciences (Engineering), Associate Professor

Editorial Board:

Prof. Sovetov B. Ya., Doctor of Sciences (Engineering), member of the Russian Academy of Sciences (RAS), the Russian Academy of Education (RAE)

Prof. Yusupov R. M., Doctor of Sciences (Engineering), member of the Russian Academy of Sciences (RAS)

Prof. Abramov O. V., Doctor of Sciences (Engineering)

Prof. Anokhin A. N., Doctor of Sciences (Engineering)

Prof. Antonov A. V., Doctor of Sciences (Engineering)

Prof. Burkhanov R. A., Doctor of Sciences (Philosophy)

Prof. Bushmeleva K. I., Doctor of Sciences (Engineering)

Prof. Getman A. F., Doctor of Sciences (Engineering)

Prof. Grigoryev L. I., Doctor of Sciences (Engineering)

Prof. Dreys Yu. G., Doctor of Sciences (Engineering)

Prof. Elnikov A. V., Doctor of Sciences (Phys&Math)

Prof. Inyutin S. A., Doctor of Sciences (Engineering)

Prof. Kashtanov V. A., Doctor of Sciences (Phys&Math)

Prof. Kerimov T. Kh., Doctor of Sciences (Philosophy)

Prof. Leonov D. G., Doctor of Sciences (Engineering)

Prof. Malyshev D. S., Doctor of Sciences (Phys&Math)

Prof. Melnikov A. V., Doctor of Sciences (Engineering)

Prof. Pevzner L. D., Doctor of Sciences (Engineering)

Prof. Uvaisov S. U., Doctor of Sciences (Engineering)

Prof. Tsibulsky V. R., Doctor of Sciences (Engineering)

Prof. Yurkov N. K., Doctor of Sciences (Engineering)

Publishing Editor

Khasanova A. Sh.

Editor

Manaeva L. I.

Layout

Melnikova E. A.

Translator

Petrova A. V.

Cover Image:

A visualization of magnetohydrodynamic analysis used to find an exact solution within a sphere

© Galkin V. A., Gorelikov A. B.

Since 29.05.2017 the journal is included in the List of Leading Peer-Reviewed Scientific Journals Published in the RF by the Higher Attestation Commission, which publishes main scientific results of doctor's and candidate's theses.

Since March 29, 2022, the journal is included in the List on the following subject groups:

2.3.1. – System Analysis, Data Management and Processing; 1.2.2. – Mathematical Modeling, Calculus and Program Complexes;

2.2.2. – Electronic Component Base of Micro- and Nanoelectronics, Quantum Devices

The journal is registered with the Federal Service for Supervision of Communications, Information Technology and Mass Media. Mass media registration certificate El No. FS77-83015 dated on 31.03.2022

Published since 2002. 4 issues per year

Editorial Board Address:

Surgut State University, Russia 628412, Khanty-Mansi Autonomous Okrug – Ugra, Surgut, Lenina pr., 1

Tel.: +7 (3462) 76-2988, fax: +7 (3462) 76-2929, e-mail: proceed.cyber@mail.ru. Web: vestcyber.ru

СОДЕРЖАНИЕ

ТЕХНИЧЕСКИЕ НАУКИ

Брыкин В. В., Брагинский М. Я., Тараканова И. О., Тараканов Д. В.	
Анализ состояния растений с применением технологий искусственного интеллекта	6
Васильянов Г. С., Лавров А. А., Лупин А. В., Малышев И. А.	
Оптимизация параметров шагательного цикла робота на основе регрессионной модели	14
Веремчук Н. С.	
Элементы имитационного моделирования в вопросах оптимизации дорожного движения	23
Гончаров А. Р., Лысенкова С. А.	
Анализ английского лексикона интенсивности эмоций и формирование алгоритма построения его русскоязычной версии	29
Громов В. А., Лукьянченко П. П., Бесчастнов Ю. Н., Томащук К. К.	
Анализ структуры временных рядов количества дел в суде	37
Лэ В. Х., Фирсов А. Н.	
Метод регуляризации Тихонова для решения обратной задачи в математической модели кинетики процесса нефтепереработки	49
Музылева И. В., Языкова Л. Н., Мещеряков В. Н.	
Применение метода пространства состояний для обобщенной системы типа «один вход – один выход»	59

ФИЗИКО-МАТЕМАТИЧЕСКИЕ НАУКИ

Базилевский М. П.	
Оценка методом наименьших квадратов простейших неэлементарных линейных регрессий с линейным аргументом в бинарной операции	69
Золотарева Н. С.	
Обзор методов и оценка сложности алгоритмов операций сравнения в модулярной арифметике и перевода из модулярной системы в позиционную систему счисления	77

CONTENTS

ENGINEERING

<i>Brykin V. V., Braginsky M. Ya., Tarakanova I. O., Tarakanov D. V.</i> Analysis of Plants Health Using Artificial Intelligence Technologies	6
<i>Vasilyanov G. S., Lavrov A. A., Lupin A. V., Malyshev I. A.</i> Optimizing Parameters of the Robot Stepping Cycle Based on the Regression Model	14
<i>Veremchuk N. S.</i> Elements of Simulation Modeling in Issues of Traffic Optimization	23
<i>Goncharov A. R., Lysenkova S. A.</i> Analyzing the English Emotion Intensity Lexicon and Generating an Algorithm to Compile Its Russian Version	29
<i>Gromov V. A., Lukyanchenko P. P., Beschastnov Yu. N., Tomashchuk K. K.</i> Time Series Structure Analysis of the Number of Law Cases	37
<i>Le V. H., Firsov A. N.</i> Tikhonov's Regularization Method for Solving the Inverse Problem in a Mathematical Model of the Kinetics of Oil Refining Process	49
<i>Muzyleva I. V., Yazykova L. N., Meshcheryakov V. N.</i> Application of the State Space Method for a Generalized Single Input Single Output System	59

PHYSICS AND MATHEMATICS

<i>Bazilevsky M. P.</i> Ordinary Least Squares Estimation of Simple Non-Elementary Linear Regressions with a Linear Argument in a Binary Operation	69
<i>Zolotareva N. S.</i> Methods Review and Complexity Estimation of the Algorithms for Comparison Operations in Modular Arithmetic and Transfer Operations from a Modular Number System to a Positional Number System	77

ТЕХНИЧЕСКИЕ НАУКИ

Научная статья

УДК 004.032.26:58

doi: 10.34822/1999-7604-2022-4-6-13

АНАЛИЗ СОСТОЯНИЯ РАСТЕНИЙ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

**Валентин Валерьевич Брыкин¹, Михаил Яковлевич Брагинский²,
Ирина Олеговна Тараканова³, Дмитрий Викторович Тараканов^{4✉}**

^{1,2,3,4} Сургутский государственный университет, Сургут, Россия

¹ valentin.brykin@mail.ru, <https://orcid.org/0000-0002-6143-8992>

² braginskiy_mya@surgu.ru, <https://orcid.org/0000-0003-1332-463X>

³ tarakanova_io@surgu.ru, <https://orcid.org/0000-0002-7707-5890>

⁴ sprtdv@mail.ru ✉, <https://orcid.org/0000-0003-1851-1039>

Аннотация. Представлены этапы разработки с помощью языка разметки HTML, каскадных таблиц стилей и языка программирования JavaScript веб-приложения для мобильных устройств, содержащего в качестве основы обученную средствами библиотек машинного обучения Tensorflow и Keras модель искусственной нейронной сети для классификации с максимальной точностью состояния растений.

Все операции, связанные с обработкой изображений, выполнены в цветовом пространстве RGB.

Ключевые слова: сверточные нейронные сети, классификация, веб-приложение

Для цитирования: Брыкин В. В., Брагинский М. Я., Тараканова И. О., Тараканов Д. В. Анализ состояния растений с применением технологий искусственного интеллекта // Вестник кибернетики. 2022. № 4 (48). С. 6–13. DOI 10.34822/1999-7604-2022-4-6-13.

Original article

ANALYSIS OF PLANTS HEALTH USING ARTIFICIAL INTELLIGENCE TECHNOLOGIES

Valentin V. Brykin¹, Mikhail Ya. Braginsky², Irina O. Tarakanova³, Dmitry V. Tarakanov^{4✉}

^{1,2,3,4} Surgut State University, Surgut, Russia

¹ valentin.brykin@mail.ru, <https://orcid.org/0000-0002-6143-8992>

² braginskiy_mya@surgu.ru, <https://orcid.org/0000-0003-1332-463X>

³ tarakanova_io@surgu.ru, <https://orcid.org/0000-0002-7707-5890>

⁴ sprtdv@mail.ru ✉, <https://orcid.org/0000-0003-1851-1039>

Abstract. The article describes the stages of developing a web-application for mobile phones based on an artificial neural network model trained by machine learning libraries (Tensorflow and Keras) to classify plant health with maximum accuracy. The HTML markup language, cascading style sheets, and JavaScript programming language were all used during the development process.

All photo editing manipulations were performed using the RGB color model.

Keywords: convolutional neural networks, classification, web-application

For citation: Brykin V. V., Braginsky M. Ya., Tarakanova I. O., Tarakanov D. V. Analysis of Plants Health Using Artificial Intelligence Technologies // Proceedings in Cybernetics. 2022. No. 4 (48). P. 6–13. DOI 10.34822/1999-7604-2022-4-6-13.

ВВЕДЕНИЕ

В настоящее время находят широкое применение и активно разрабатываются автоматические системы дистанционного контроля состояния растений, позволяющие своевременно определить наличие заболеваний растений, водный статус, получать сведения о биомассе в сыром и сухом состояниях как в полевых условиях, так и в тепличных хозяйствах [1–6].

Распознавание состояния растений предполагает решение задачи классификации изображений, и в качестве классификатора предлагается использовать искусственную нейронную сеть (ИНС) как обучающуюся модель, работа которой практически не требует вмешательства пользователя.

При решении задачи оценки состояния растений используется многоклассовая классификация, т. е. изображение будет относиться к одному из 8 выбранных классов наиболее распространенных заболеваний растительных культур:

- bacterial spot (бактериальная пятнистость);
- fungle spot (грибковая пятнистость);
- healthy (здоровое);
- late blight (фитофтороз);
- leaf mold (листовая плесень);
- leaf scorch (ожог листьев);
- powdery mildew (мучнистая роса);
- yellow curl virus (вирус желтой курчавости).

На выходе классификатора формируется вектор вероятностей отношения входного изображения к каждому классу. Класс с наивысшим значением вероятности и будет являться результатом решения задачи классификации.

Представлены этапы создания и развертывания для общего пользования модели ИНС сверточной архитектуры, решающей задачу многоклассовой классификации состояния растений по цветным изображениям. Все операции, связанные с обработкой изображений, выполнены в цветовом пространстве RGB.

МАТЕРИАЛЫ И МЕТОДЫ

Материалом для исследований послужили цифровые фотографии листьев здоровых и больных растений, опубликованные в открытом доступе платформы Kaggle. Объем входных данных составляет примерно из 18 000 изображений 8 разных классов. Количество

примеров в каждом классе должно быть одинаковым, чтобы избежать несбалансированности классов. Несбалансированность классов создает проблемы при решении задач классификации, поскольку построенные на таких данных модели имеют «перекос» в сторону мажоритарного класса [7, 8]. Выборка изображений делится на 3 компонента в следующем соотношении данных: обучающая – 70 %, валидационная (проверочная) и тестовая – по 15 %. Обучающая выборка используется непосредственно в процессе обучения и является основной, проверочная часть датасета – для оптимизации гиперпараметров сети и исключения переобучения, тестовая – для оценки качества работы сети.

В качестве среды разработки использовалась облачная платформа Google Colab, связанная с учетной записью разработчика на Google Диске, что позволяет существенно упростить исследования в области машинного обучения [8–10]. В качестве классификатора состояния растений используется модель ИНС сверточной архитектуры (CNN), решающая задачу многоклассовой классификации состояния растений по цветным изображениям.

При решении поставленной задачи была использована предобученная сверточная нейронная сеть MobileNet на наборе данных «ImageNet», что позволяет значительно уменьшить время обучения.

Одна из причин переобучения нейронной сети [8] – слишком большое количество в сети нейронов. В данной работе модель CNN MobileNet имеет очень большое число нейронов ввиду своей большой глубины. Таким образом, если наблюдается расхождение в точности выходных значений между обучающей и проверочной выборкой, то процесс обучения следует остановить и уменьшить число нейронов в модели. Но с уменьшением числа нейронов уменьшается и точность выходных значений, то есть ухудшится показатель качества работы ИНС, поэтому при переобучении необходимо сохранение числа нейронов [11].

Это помогает сделать широко распространенный алгоритм Dropout, часто называемый в русском языке методом исключения, целью которого является снижение специализации

каждого отдельного нейрона и его преобразование для решения более общей задачи [11], а именно: на каждой итерации изменения весовых коэффициентов часть нейронов нужно исключать с заданной вероятностью p .

Алгоритм Dropout используется к классификационной части нейронной сети и распо-

лагается перед последним, выходным слоем «Dense» (рис. 1).

У алгоритма Dropout используется один аргумент – вероятность p того, что нейрон не будет исключен из сети.

```
head_model = base_model.output
head_model = GlobalAveragePooling2D()(head_model)
head_model = Dropout(0.2)(head_model)
outputs = Dense(8, activation="softmax")(head_model)
mobilenet_model = Model(base_model.input, outputs, name='pretrained_mobilenet')
for layer in mobilenet_model.layers:
    layer.trainable = True
```

Рис. 1. Применение алгоритма Dropout в Keras для модели на основе MobileNet

Примечание: составлено авторами.

Для исследования влияния алгоритма Dropout на качество работы ИНС проведено несколько обучений CNN-модели с разными

значениями вероятности p . Результаты моделирования приведены в табл. 1.

Таблица 1

Результаты обучения ИНС с использованием алгоритма Dropout

Значение вероятности p	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
Точность (обучающая выборка), %	77	80	86	88	87	85	81	81	79
Точность (проверочная выборка), %	67	72	76	85	83	77	74	70	70

Примечание: составлено авторами.

Из табл. 1 видно, что лучшим значением вероятности является $p = 0,4$, так как при этом значении были продемонстрированы лучшие показатели в точности на обеих выборках, а также наилучший прогресс в точности – на проверочной выборке.

На рис. 2 отображен ход обучения сети MobileNet методом исключения с выбранной лучшей вероятностью $p = 0,4$.

Таким образом, метод вероятностного исключения нейронов Dropout помог избавиться от переобучения в рамках данной конкретной задачи.

На рис. 3 отражены зависимости качества обучения модели нейронной сети MobileNet (точность и ошибка для обучающей и валидационной выборок), улучшенная методами аугментации данных и Dropout.

```
Epoch 1/8
64/64 [=====] - 671s 10s/step - loss: 1.5996 - accuracy: 0.4538 - val_loss: 11.1643 - val_accuracy: 0.2246 - lr: 0.0010
Epoch 2/8
64/64 [=====] - 669s 10s/step - loss: 0.9971 - accuracy: 0.6742 - val_loss: 6.3521 - val_accuracy: 0.4600 - lr: 0.0010
Epoch 3/8
64/64 [=====] - 525s 8s/step - loss: 0.8133 - accuracy: 0.7410 - val_loss: 2.3861 - val_accuracy: 0.6426 - lr: 0.0010
Epoch 4/8
64/64 [=====] - 440s 7s/step - loss: 0.7048 - accuracy: 0.7696 - val_loss: 1.8320 - val_accuracy: 0.7048 - lr: 0.0010
Epoch 5/8
64/64 [=====] - 366s 6s/step - loss: 0.6680 - accuracy: 0.7924 - val_loss: 0.7391 - val_accuracy: 0.8143 - lr: 0.0010
Epoch 6/8
64/64 [=====] - 305s 5s/step - loss: 0.6342 - accuracy: 0.7990 - val_loss: 0.8367 - val_accuracy: 0.7984 - lr: 0.0010
Epoch 7/8
64/64 [=====] - 252s 4s/step - loss: 0.5914 - accuracy: 0.8042 - val_loss: 0.7685 - val_accuracy: 0.8170 - lr: 0.0010
Epoch 8/8
64/64 [=====] - 210s 3s/step - loss: 0.5744 - accuracy: 0.8772 - val_loss: 0.5322 - val_accuracy: 0.8506 - lr: 0.0010
```

Рис. 2. Обучение модели на основе MobileNet при $p = 0,4$

Примечание: составлено авторами.

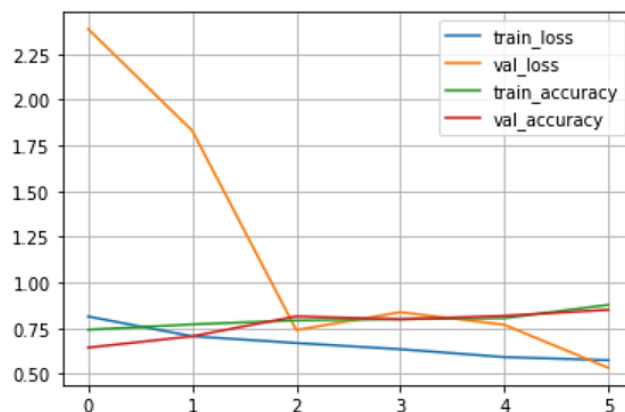


Рис. 3. Графическое представление процесса обучения
Примечание: составлено авторами.

Из рис. 3 видно, что значения точности на обучающей и особенно проверочной выборках повысились и находятся практически на одном уровне 88 и 85 % соответственно (зеленая и красная линии), что подтверждает действенность описанных ранее методов совершенствования модели.

Затем выполнена процедура тестирования обученной модели: случайным образом при помощи библиотек Keras для работы с изображениями выбирается изображение из тестовой выборки (не участвовавшее в обучении) (рис. 4).



Рис. 4. Генерация случайного изображения из тестовой выборки
Примечание: составлено авторами.

Как видно из названия объекта, это растение с болезнью «фитофтороз» (класс «late_blight»).

Результатом является массив вероятностей отношения тестируемого изображения ко всем 8 классам (рис. 5). При этом видно,

что с вероятностью 99,9 % изображение относится к классу с меткой «3». На рис. 6 с метками классов видно, что данной метке соответствует класс «late_blight», следовательно, классификация выполнена успешно.

При создании приложения необходимо выполнить развертывание модели на основе MobileNet для общего пользования и преобразовать модель «mobilenet_model» из формата «H5» в формат «JSON», совместимый с любой

JavaScript средой. Файлы формата «JSON» позволяют осуществлять обмен данными между клиентской и серверной частями приложения.

```
test1 = test1.resize((224,224))
test1_scaled = np.expand_dims(np.asarray(test1), axis = 0) / 255
predictions = mobilenet_model.predict(test1_scaled)
print(predictions)

[[5.6686909e-05 2.8846331e-04 1.5034264e-05 9.9963975e-01 6.2135714e-09
 8.8132729e-10 3.3346109e-10 4.1249504e-10]]
```

Рис. 5. Результат классификации моделью на основе MobileNet
Примечание: составлено авторами.

```
classes_dict = train_set_from_dir.class_indices
classes_dict = { v:k for (k,v) in classes_dict.items() }
classes_dict[np.argmax(predictions)]

'late_blight'
```

Рис. 6. Успешная классификация
Примечание: составлено авторами.

Для запуска моделей машинного обучения в браузере с использованием JavaScript [12] и решения задачи конвертации используется библиотека «Tensorflow.js», имеющая следующие преимущества:

- наличие большого числа инструментов для визуализации происходящих процессов (графики, анимация и др.);
- наличие прямого доступа к сенсорам устройства (камера, GPS и пр.);
- отсутствие необходимости отправлять обрабатываемые данные на сервер, как следствие – безопасность данных пользователя;
- совместимость с Python-моделями.

Tensorflow.js запускается в браузере, поскольку это обеспечивает высокую производительность за счет использования библиотеки WebGL для выполнения различных математических операций с тензорами.

Далее необходимо выполнить преобразование модели в JSON-формат командой «tensorflowjs-converter». По окончании выполнения команды на Google Диске создается архив с выходными данными, который нужно распаковать, а данные из него сохранить в ранее созданном каталоге «tensorflowjs-model» (рис. 7).

```
!tensorflowjs_converter --input_format keras models/mobilenet_model.h5 tensorflowjs-model/

!zip -r tensorflowjs-model.zip tensorflowjs-model

updating: tensorflowjs-model/ (stored 0%)
updating: tensorflowjs-model/group1-shard1of4.bin (deflated 7%)
updating: tensorflowjs-model/group1-shard2of4.bin (deflated 7%)
updating: tensorflowjs-model/group1-shard3of4.bin (deflated 7%)
updating: tensorflowjs-model/group1-shard4of4.bin (deflated 7%)
updating: tensorflowjs-model/model.json (deflated 94%)
```

Рис. 7. Конвертация Keras-модели в формат JSON и сохранение выходных данных в каталоге «tensorflowjs-model»
Примечание: составлено авторами.

Развертывание модели машинного обучения для общего пользования предполагает создание web-приложения – программы, позволяющей пользователю взаимодействовать с этой моделью через браузер с использованием компьютера, в частности сети и веб-сервера.

Пользовательский интерфейс web-приложения создается путем верстки на языке разметки HTML с применением языка формального описания CSS.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Тестирование разработанного приложения осуществляли на локальном веб-сервере – эмуляторе хостинга с использованием XAMPP (Apache + MariaDB + PHP + Perl) – бесплатной кроссплатформенной сборки веб-сервера.

Произведена проверка работы функциональных характеристик системы: загрузки в браузер модели машинного обучения, вывода на экран загруженного изображения и результатов классификации. Для этого нужно нажать левой клавишей мыши на большую светлую область в центре экрана (кнопка со значком стрелки загрузки) и выбрать на устройстве изображение растения (рис. 8).

Сообщение «Model Loaded Successfully» оповещает об успешной загрузке конвертированной модели машинного обучения (рис. 8). Надпись на кнопке загрузки поменялась – после загрузки картинки предлагается заменить изображение.

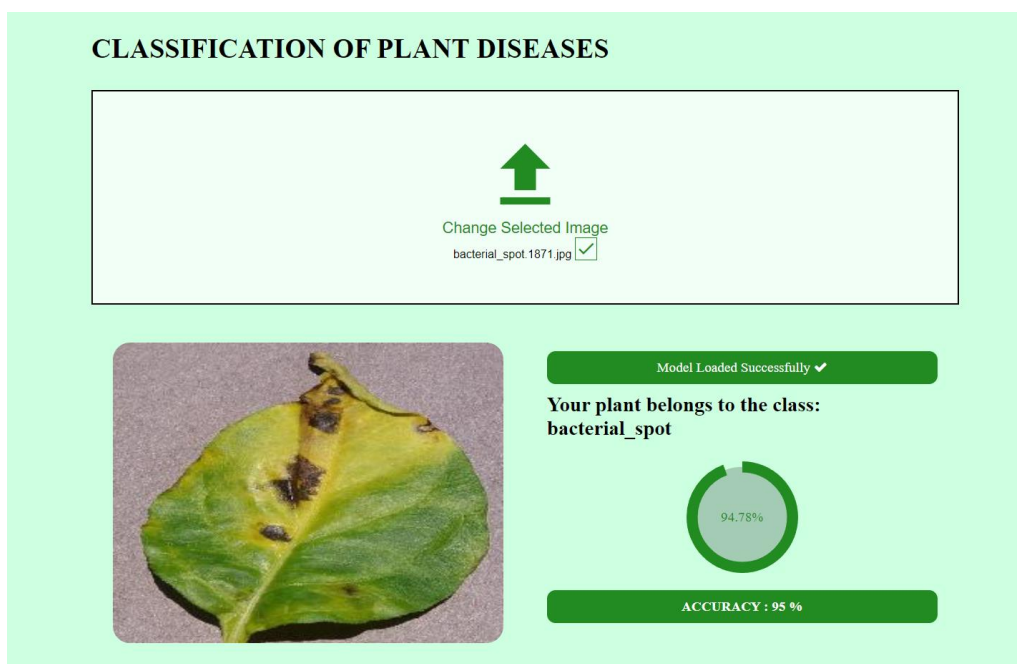


Рис. 8. Результат работы приложения

Примечание: составлено авторами.

Изображение на рис. 8 листа растения, пораженного бактериальной пятнистостью, было получено из тестовой выборки изображений, не участвовавших в обучении загруженной в браузер модели. Как видно из рисунка, картинка была отнесена нейронной сетью к классу «bacterial_spot» (это и есть бактериальная пятнистость) с вероятностью 94,78 %. Этот результат отражен на круглом индикаторе со шкалой, заполненной соответственно значению вероятности, а также на

блоке под индикатором в виде округленного до ближайшего целого значения.

При тестировании приложения выявлены следующие особенности работы системы классификации:

- объект должен занимать максимальное пространство на изображении;
- объект должен быть равномерно освещен.

При невыполнении этих требований точность распознавания падает.

Для выполнения первой классификации требуется немного больше времени, чем для последующих (примерно 7–10 секунд против 3–5 на последующих изображениях). Это связано с затратами времени на инициализацию модели.

При попытке загрузки файла, не являющегося изображением (рис. 9), откроется всплывающее окно с предупреждением о некорректном формате выбранного файла.

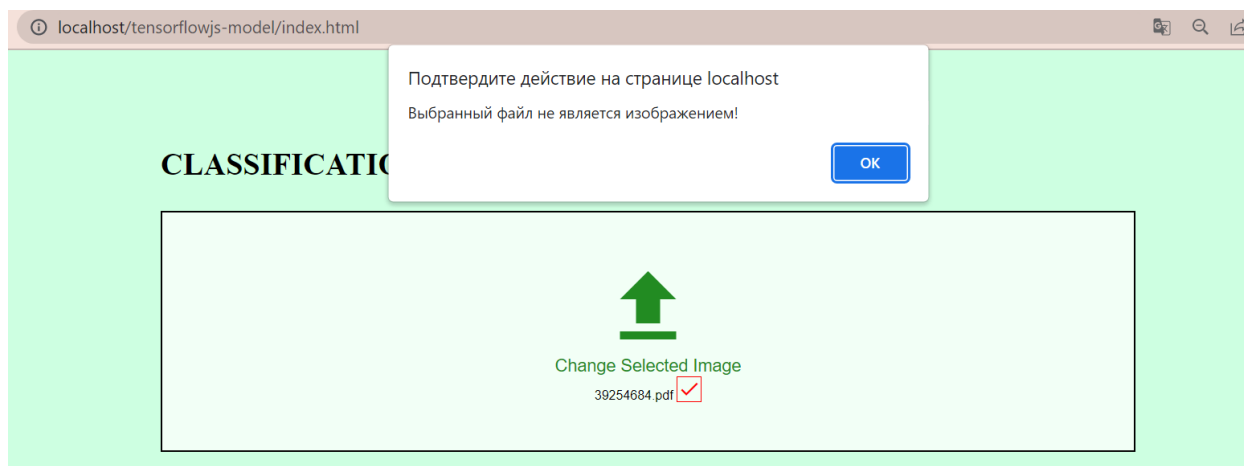


Рис. 9. Предупреждение при загрузке файла некорректного формата
Примечание: составлено авторами.

Таким образом, приложение успешно загрузило модель нейронной сети и корректно классифицировало предоставленное изображение.

ЗАКЛЮЧЕНИЕ

Результатом работы является развернутое для публичного использования и адаптированное под мобильные устройства веб-приложение, функционирующее на основе созданной модели сверточной нейронной сети, решающей задачу классификации состояния растений по анализу фотографий.

Использованы методы улучшения качества работы реализованной модели – аугментация входных данных и алгоритм Dropout, которые позволили повысить точность работы сети на 10 % на обучающей выборке

и на 26 % – на проверочной. Кроме того, с их помощью была решена проблема переобучения модели. Для повышения качества классификации необходимо использовать предварительную обработку регистрируемых изображений: повышение контрастности, масштабирование изображения и т. д.

Улучшенная CNN-модель была развернута для общего пользования в виде веб-приложения.

Проведено тестирование приложения с использованием локального веб-сервера XAMPP для проверки корректности выполнения основных функций: запуска приложения, загрузки модели и файлов, генерации результатов классификации и вывода на экран данных, обработки исключений.

Список источников

1. Rahman C. R., Arko P. S., Ali M. E. et al. Identification and Recognition of Rice Diseases and Pests Using Convolutional Neural Networks // Biosystems Engineering. 2020. Vol. 194. P. 112–120. DOI 10.1016/j.biosystemseng.2020.03.020.
2. Янишевская Н. А., Болодурина И. П. Применение технологий компьютерного зрения для разработки модели распознавания поражений культурных растений // Вестн. Юж.-Урал. гос. ун-та.

References

1. Rahman C. R., Arko P. S., Ali M. E. et al. Identification and Recognition of Rice Diseases and Pests Using Convolutional Neural Networks // Biosystems Engineering. 2020. Vol. 194. P. 112–120. DOI 10.1016/j.biosystemseng.2020.03.020.
2. Yanishevskaya N. A., Bolodurina I. P. Application of Computer Vision Technologies for the Development of a Model for the Recognition of Lesions of Cultivated Plants // Bulletin of the South Ural State

- Серия «Компьютерные технологии, управление, радиоэлектроника». 2021. Т. 21, № 3. С. 5–13.
3. Тараканов Д. В., Брагинский М. Я., Брыкин В. В. Повышение качества идентификации состояния растений искусственной нейронной сетью // Наука и образование в эпоху перемен: перспективы развития, новые парадигмы : материалы X Всерос. науч.-практич. конф., Ростов-на-Дону, 15 июля 2022 г. Ростов-на-Дону : Манускрипт, 2022. С. 51–60.
 4. Брагинский М. Я., Тараканов Д. В. Фенотипирование растений адаптивной системой обработки изображений на базе сверточных нейронных сетей // Вестник кибернетики. 2021. № 2 (42). С. 6–16.
 5. Rahman M. A., Islam M. M., Mahdee G. M. S., Kabir M. W. U. Improved Segmentation Approach for Plant Disease Detection // Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), May 03–05, 2019, Bangladesh. P. 1–5. DOI 10.1109/ICASERT.2019.8934895.
 6. Чешкова А. Ф. Обзор современных методов обнаружения и идентификации болезней растений на основе анализа гиперспектральных изображений // Вавилов. журн. генетики и селекции. 2022. Т. 26, № 2. С. 202–213.
 7. Сэмплинг в условиях несбалансированности классов. URL: <https://loginom.ru/blog/imbalance-class> (дата обращения: 02.11.2022).
 8. Погружение в сверточные нейронные сети: передача обучения (transfer learning). URL: <https://habr.com/ru/post/467967/> (дата обращения: 02.11.2022).
 9. Keras Applications. URL: <https://keras.io/api/applications/> (дата обращения: 02.11.2022).
 10. Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition // ArXiv. 2015. 14 p. URL: <https://arxiv.org/pdf/1409.1556.pdf> (дата обращения: 02.11.2022).
 11. Dropout – метод борьбы с переобучением нейронной сети. URL: https://proproprogs.ru/neural_network/dropout-metod-borby-s-pereobucheniem-neuronnoy-seti (дата обращения: 02.11.2022).
 12. Представляем TensorFlow.js: машинное обучение в JavaScript. URL: <https://datastart.ru/blog/read/predstavlyaem-tensorflowjs-mashinnoe-obuchenie-v-java-script> (дата обращения: 02.11.2022).
 - University. Series “Computer Technologies, Automatic Control, Radioelectronics”. 2021. Vol. 21, No. 3. P. 5–13. (In Russian).
 3. Tarakanov D. V., Braginsky M. Ya., Brykin V. V. Improving the Quality of Identifying the State of Plants by an Artificial Neural Network // Science and Education in an Era of Change: Development Prospects, New Paradigms : Proceedings of the X All-Russian Research-to-Practice Conference, Rostov-on-Don, July 15, 2022. Rostov-on-Don : Manuscript, 2022. P. 51–60. (In Russian).
 4. Braginsky M. Ya., Tarakanov D. V. Plant Phenotyping by an Adaptive Image Processing System Based on Convolutional Neural Networks // Proceedings in Cybernetics. 2021. No. 2 (42). P. 6–16. (In Russian).
 5. Rahman M. A., Islam M. M., Mahdee G. M. S., Kabir M. W. U. Improved Segmentation Approach for Plant Disease Detection // Proceedings of the 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), May 03–05, 2019, Bangladesh. P. 1–5. DOI 10.1109/ICASERT.2019.8934895.
 6. Cheshkova A. F. A Review of Hyperspectral Image Analysis Techniques for Plant Disease Detection and Identification // Vavilov Journal of Genetics and Breeding. 2022. Vol. 26, No. 2. P. 202–213. (In Russian).
 7. Sampling v usloviakh nesbalansirovannosti klassov. URL: <https://loginom.ru/blog/imbalance-class> (accessed: 02.11.2022). (In Russian).
 8. Pogruzhenie v svertochnye neuronnye seti: peredacha obucheniia (transfer learning). URL: <https://habr.com/ru/post/467967/> (accessed: 02.11.2022). (In Russian).
 9. Keras Applications. URL: <https://keras.io/api/applications/> (accessed: 02.11.2022).
 10. Simonyan K., Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition // ArXiv. 2015. 14 p. URL: <https://arxiv.org/pdf/1409.1556.pdf> (accessed: 02.11.2022).
 11. Dropout – metod borby s pereobucheniem. URL: https://proproprogs.ru/neural_network/dropout-metod-borby-s-pereobucheniem-neuronnoy-seti (accessed: 02.11.2022). (In Russian).
 12. Predstavliaem TensorFlow.js: mashinnoe obuchenie v JavaScript. URL: <https://datastart.ru/blog/read/predstavlyaem-tensorflowjs-mashinnoe-obuchenie-v-java-script> (accessed: 02.11.2022). (In Russian).

Информация об авторах

В. В. Брыкин – аспирант.
М. Я. Брагинский – кандидат технических наук, доцент.
И. О. Тараканова – аспирант.
Д. В. Тараканов – кандидат технических наук, доцент.

Information about the authors

V. V. Brykin – Postgraduate.
M. Ya. Braginsky – Candidate of Sciences (Engineering), Associate Professor.
I. O. Tarakanova – Postgraduate.
D. V. Tarakanov – Candidate of Sciences (Engineering), Associate Professor.

Научная статья
УДК 681.5
doi: 10.34822/1999-7604-2022-4-14-22

ОПТИМИЗАЦИЯ ПАРАМЕТРОВ ШАГАТЕЛЬНОГО ЦИКЛА РОБОТА НА ОСНОВЕ РЕГРЕССИОННОЙ МОДЕЛИ

Георгий Сергеевич Васильянов¹, Алексей Александрович Лавров^{2✉},
Анатолий Викторович Лупин³, Игорь Алексеевич Малышев⁴

^{1, 2, 3, 4} Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Россия

¹ g.vasilyanov@mail.ru, <https://orcid.org/0000-0003-2115-6947>

² lavrov_aa@spbstu.ru ✉, <https://orcid.org/0000-0001-5299-4789>

³ lupin_av@spbstu.ru, <https://orcid.org/0000-0002-1920-069X>

⁴ malyshev_ia@spbstu.ru, <https://orcid.org/0000-0002-8035-4401>

Аннотация. Рассмотрен алгоритм ситуационного управления движениями четырехногого робота, основанный на построении и оптимизации регрессионной модели в среде Gazebo. Представлены результаты вычислительных экспериментов влияния длины и высоты шага и времени переноса ноги на снижение до минимально возможного уровня колебаний его корпуса по углам крена и тангажа.

При описании поверхности отклика, предполагаемо имеющей экстремумы, применена регрессионная модель в виде полинома второго порядка, а с целью сокращения числа опытов при ее построении использован ортогональный центральный композиционный план. Описан процесс получения регрессионной модели, оценки значимости ее коэффициентов, оценки адекватности регрессионной модели виртуальной модели робота в среде Gazebo, приведены результаты оптимизации параметров шагательного цикла при помощи симплекса метода Нелдера – Мида в среде Matlab, применение которого возможно при оптимизации параметров систем управления по другим критериям.

Ключевые слова: шагающий четырехногий робот, система управления, симулятор Gazebo, регрессионная модель

Для цитирования: Васильянов Г. С., Лавров А. А., Лупин А. В., Малышев И. А. Оптимизация параметров шагательного цикла робота на основе регрессионной модели // Вестник кибернетики. 2022. № 4 (48). С. 14–22. DOI 10.34822/1999-7604-2022-4-14-22.

Original article

OPTIMIZING PARAMETERS OF THE ROBOT STEPPING CYCLE BASED ON THE REGRESSION MODEL

Georgy S. Vasilyanov¹, Aleksey A. Lavrov^{2✉}, Anatoly V. Lupin³, Igor A. Malyshev⁴

^{1, 2, 3, 4} Peter the Great St. Petersburg Polytechnic University, Saint Petersburg, Russia

¹ g.vasilyanov@mail.ru, <https://orcid.org/0000-0000-0003-2115-6947>

² lavrov_aa@spbstu.ru ✉, <https://orcid.org/0000-0001-5299-4789>

³ lupin_av@spbstu.ru, <https://orcid.org/0000-0002-1920-069X>

⁴ malyshev_ia@spbstu.ru, <https://orcid.org/0000-0002-8035-4401>

Abstract. The article considers an algorithm for situational control of a quadruped dog, based on building and optimizing a regression model in Gazebo. The results of simulation experiments demonstrate the effect of step length and height as well as leg transfer time on the minimum possible decrease in the roll and pitch angles of the robot body.

A regression model in the form of a second-order polynomial was used to describe the response surface with supposed extremes. When building such a regression model, an orthogonal central composite plan was

applied in order to minimize the number of experiments. The study describes the process of obtaining a regression model, determining its coefficients significance, and evaluating the adequacy of the regression model's virtual robot model in the Gazebo simulator. The results of parameter optimization for a stepping cycle are shown in Matlab using the simplex of the Nelder–Mead method, which can be applied when optimizing parameters of control systems according to other criteria.

Keywords: walking quadruped robot, control system, Gazebo simulator, regression model

For citation: Vasilyanov G. S., Lavrov A. A., Lupin A. V., Malyshev I. A. *Optimizing Parameters of the Robot Stepping Cycle Based on the Regression Model // Proceedings in Cybernetics. 2022. No. 4 (48). P. 14–22. DOI 10.34822/1999-7604-2022-4-14-22.*

ВВЕДЕНИЕ

Шагающие роботы (ШР) имеют более высокую проходимость по сравнению с колесными, поэтому основной сферой применения ШР является исследование участков сильно пересеченной местности и опасных для человека. Учитывая возможные условия передвижения ШР по маршруту с ровными и неровными поверхностями, твердому или мягкому грунту, необходимо использовать ситуационный подход, и для каждой ситуации разработать алгоритм, который позволит выполнить задачу с максимальной эффективностью, т. е. разбить общий алгоритм управления на частные алгоритмы с оптимальными параметрами для конкретных условий [1]. Выбор алгоритма из возможных для конкретной ситуации решает более высокий уровень системы управления ШР.

Рассмотрен алгоритм ситуационного управления движением четырехногого робота по плоской поверхности с обеспечением снижения колебаний корпуса по углам крена и тангажа до минимально возможного уровня, что важно, например, при использовании ШР для видеосъемки. Для проектирования робота и его системы управления использовали моделирование в среде универсального симулятора Gazebo [2, 3] при соответствии параметров модели параметрам опытного образца [4]. При проведении вычислительных экспериментов установлено, что на углы крена и тангажа корпуса робота значительно влияют длина шага, высота на которую робот поднимает ногу, и временной интервал между сменами подшагов. Применение численных методов оптимизации, например, метода градиентного спуска или

метода наискорейшего спуска позволяют найти оптимальное значение целевой функции, но сам процесс поиска не дает полной информации о поверхности отклика. Поэтому в работе исследуется метод, основанный на построении регрессионной модели по результатам моделирования ШР в среде Gazebo и проводимой по этой модели оптимизации [5]. При описании поверхности отклика, предполагаемо имеющей экстремумы, чаще всего применяют регрессионные модели в виде полинома второго порядка, а с целью сокращения числа опытов при построении моделей используют ортогональные центральные композиционные планы. Такой подход применяется при моделировании и оптимизации объектов в самых различных областях, например, горном деле, транспорте, робототехнике [6–10].

В работе описан процесс получения регрессионной модели, оценки значимости ее коэффициентов и адекватности регрессионной модели виртуальной модели робота в среде Gazebo, приведены результаты оптимизации параметров шагательного цикла при помощи симплекс метода Нелдера – Мида в среде Matlab. Произведена оценка соответствия полученных расчетных данных данным, полученным при проведении испытаний опытного образца робота.

МАТЕРИАЛЫ И МЕТОДЫ

Для четырехногих роботов используется один из самых распространенных типов походок животных – шаг кошки или собаки. Шаг является наиболее экономным видом движения при неспешной ходьбе и позволяет сохранять равновесие [11], поэтому за основу была взята именно эта манера движения (рис. 1).

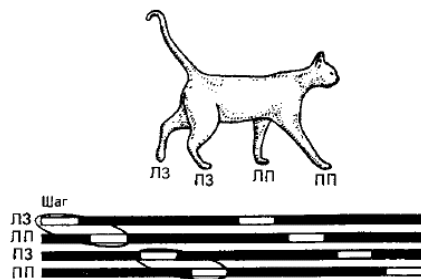


Рис. 1. Движения ног кошки при походке типа шаг

Примечание: составлено по [4].

На рис. 1 светлые горизонтальные полосы – фазы времени, когда стопа в воздухе; темные полосы – фазы, когда стопа опирается на поверхность. Обозначения ног: ЛЗ – левая зад-

няя, ЛП – левая передняя, ПЗ – правая задняя, ПП – правая передняя нога.

Фотография опытного образца четырехногого ШР приведена на рис. 2.

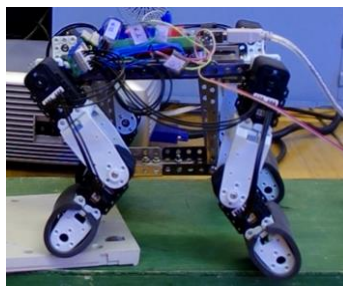


Рис. 2. Фотография опытного образца шагающего робота

Примечание: составлено авторами.

Система управления ШР, выполненная на основе платы Arduino Mega2560 формирует микрокоманды управления восемью сервоприводами (по два на каждую ногу). Используются «интеллектуальные» сервоприводы фирмы Dynamixel AX-12 со встроенным микроконтроллером, позволяющим в пакетном режиме не только получать команды от системы управления (СУ), но и отправлять на СУ данные о состоянии, в частности, текущем угле поворота, текущей нагрузке, моменте, приложенному к выходному валу, скорости и т. д.

Разработан алгоритм формирования циклограммы, обеспечивающей шагательный цикл типа «кошка» [3, 4].

На рис. 3 приведено изображение динамической модели ШР в среде симулятора Gazebo. Параметры ног и туловища модели – массы и длины бедра и голени, масса и размеры туловища, параметры сервоприводов соответствуют параметрам опытного образца робота. Проведенные испытания опытного образца и модели показали удовлетворительное совпадение их перемещений и движений [3].

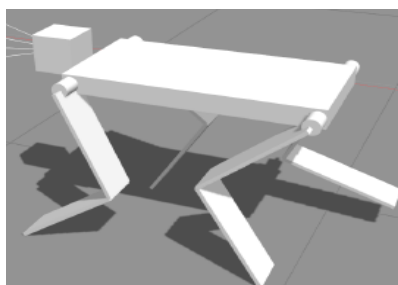


Рис. 3. Модель шагающего робота в среде симулятора Gazebo

Примечание: составлено авторами.

Каждая нога робота поочередно выполняет движение согласно диаграмме на рис. 1. Траектория движения каждой стопы представляет собой прямоугольник. Для каждой точки траектории, заданной с некоторой дискретностью, СУ вычисляет углы поворота сервоприводов

в градусах по результатам решения задачи обратной кинематики.

На рис. 4 приведена кинематическая схема задней ноги робота. Направление движения противоположно оси ОХ [4].

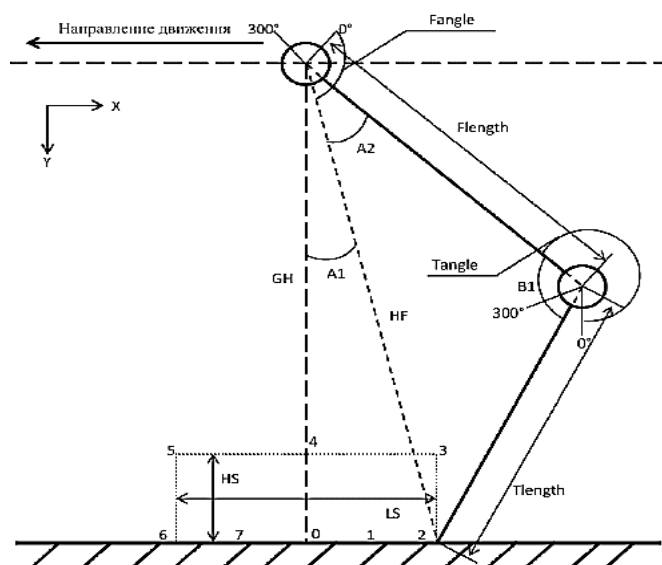


Рис. 4. Формирование траектории перемещения ноги робота:

GH (Ground Height) – расстояние от точки опоры ноги робота до поверхности; Flength (Femur Length) – длина «бедр» ноги робота, ее первого звена (P1); Tlength (Tibia Length) – длина «голени» ноги робота, ее второго звена (P2); HS (Height of Step) – высота шага ноги робота; LS (Length of Step) – длина шага ноги робота; HF (Height of Full leg) – длина ноги робота; A1, A2 – углы, из которых складывается результирующий угол поворота сервопривода, отвечающего за сегмент P1 ноги робота; B1 – угол, из которого складывается результирующий угол поворота сервопривода, отвечающего за сегмент P2 ноги робота.

Примечание: составлено авторами.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Разработанный алгоритм перерасчета циклограмм движения ног использует в качестве входных данных три переменные: расстояние до поверхности (GH – Ground Height, HS – Height of Step) – высота шага ноги робота, и LS (Length of Step) – длина шага ноги робота. При этом алгоритму требуются 2 константных параметра – длина голени Tlength и длина бедра Flength ноги робота. Параметры GH, HS и LS задаются в процессе движения [4]. Изменяя LS можно осуществлять повороты.

В ходе работы с моделью ППР выявлено, что на амплитуду колебаний корпуса робота влияют следующие факторы: длина шага (LS), высота подъема ноги (HS) и время подшага (затраченное на формирование траектории каждой ноги) – TD.

Формула зависимости амплитуды колебаний от этих факторов имеет вид:

$$Y = F(TD, HS, LS).$$

Временные диаграммы углов крена (Roll) и тангажа (Pitch) для одного из возможных сочетаний LS, HS и TD приведены на рис. 5.

Для проведения экспериментов согласно ортогональному центральному композиционному плану (ОЦКП) необходимо привести входные переменные к кодированному виду, для этого выбраны пределы изменений факторов: нижнему пределу присвоено значение «-1», среднему – «0», верхнему – «1» (табл. 1). Пределы изменения факторов были выбраны исходя из допускаемых экспериментальной установкой или моделью робота пределов изменений.

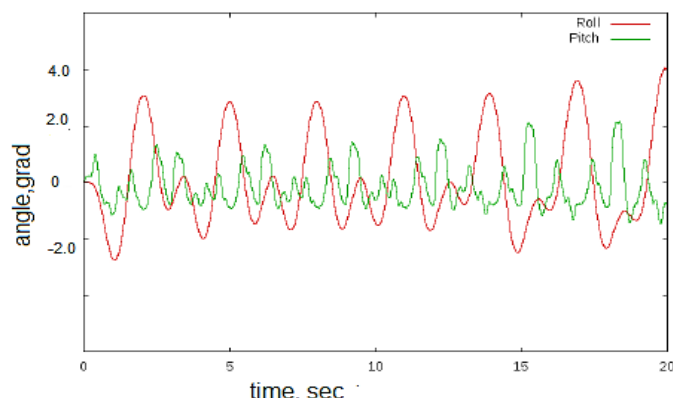


Рис. 5. Временные диаграммы углов крена (Roll) и тангажа (Pitch)

Примечание: составлено авторами.

Таблица 1

Данные для проведения ОЦКП

Факторы	Время подшага, мс	Высота подъема ноги, м	Длина шага, м
Обозначение	TD	HS	LS
Обозначения в ОЦКП	x_1	x_2	x_3
Верхний предел (1)	300	0,06	0,08
Основной уровень (0)	200	0,04	0,06
Нижний предел (-1)	100	0,02	0,04

Примечание: составлено авторами.

После приведения переменных к кодированному виду формула зависимости амплитуды колебаний от факторов будет иметь вид: $y = f(x_1, x_2, x_3)$.

Проведение вычислительных экспериментов проводится согласно обозначенной в табл. 2

матрице ОЦКП [12]. В ходе экспериментов были измерены углы крена и тангажа в градусах средствами среды Gazebo. Значения углов крена приведены в табл. 3 и использованы для расчетов по табл. 2.

Таблица 2

Матрица ОЦКП

Номер опыта	x_0	x_1	x_2	x_3	x_1^2	x_2^2	x_3^2	x_1x_2	x_1x_3	x_2x_3	y
1	+1	-1	-1	-1	0,27	0,27	0,27	+1	+1	+1	y_1
2	+1	+1	-1	-1	0,27	0,27	0,27	-1	-1	+1	y_2
3	+1	-1	+1	-1	0,27	0,27	0,27	-1	+1	-1	y_3
4	+1	+1	+1	-1	0,27	0,27	0,27	+1	-1	-1	y_4
5	+1	-1	-1	+1	0,27	0,27	0,27	+1	-1	-1	y_5
6	+1	+1	-1	+1	0,27	0,27	0,27	+1	+1	-1	y_6
7	+1	-1	+1	+1	0,27	0,27	0,27	-1	-1	+1	y_7
8	+1	+1	+1	+1	0,27	0,27	0,27	+1	+1	+1	y_8
9	+1	-1,215	0	0	0,75	-0,73	-0,73	0	0	0	y_9
10	+1	+1,215	0	0	0,75	-0,73	-0,73	0	0	0	y_{10}
11	+1	0	-1,215	0	-0,73	0,75	-0,73	0	0	0	y_{11}
12	+1	0	+1,215	0	-0,73	0,75	-0,73	0	0	0	y_{12}
13	+1	0	0	-1,215	-0,73	-0,73	0,75	0	0	0	y_{13}
14	+1	0	0	+1,215	-0,73	-0,73	0,75	0	0	0	y_{14}
15	+1	0	0	0	-0,73	-0,73	-0,73	0	0	0	y_{15}

Примечание: составлено авторами.

Значения углов крена

Y1	Y2	Y3	Y4	Y5	Y6	Y7	Y8	Y9	Y10	Y11	Y12	Y13	Y14	Y15
2,74	4,22	2,25	3,08	2,96	3,7	2,38	2,8	2,34	3,1	3,92	2,65	2,82	2,96	3,08

Примечание: составлено авторами.

По результатам проведения эксперимента согласно трехфакторному ОЦКП строится зависимость:

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + b_{12}x_1x_2 + b_{13}x_1x_3 + b_{23}x_2x_3 + b_{11}x_1^2 + b_{22}x_2^2 + b_{33}x_3^2,$$

где $b_1 = \sum_{i=1}^n (x_{1i}y_i)/10,94 = 0,40;$
 $b_2 = \sum_{i=1}^n (x_{2i}y_i)/10,94 = -0,43;$
 $b_3 = \sum_{i=1}^n (x_{3i}y_i)/10,94 = -0,025;$
 $b_{12} = \sum_{i=1}^n (x_{1i}x_{2i}y_i)/8,0 = -0,122;$
 $b_{13} = \sum_{i=1}^n (x_{1i}x_{3i}y_i)/8,0 = -0,0144;$
 $b_{23} = \sum_{i=1}^n (x_{2i}x_{3i}y_i)/8,0 = 0,0193;$
 $b_{11} = \sum_{i=1}^n (x_{1i}^2y_i)/4,34 = -0,154;$
 $b_{22} = \sum_{i=1}^n (x_{2i}^2y_i)/4,34 = 0,231;$
 $b_{33} = \sum_{i=1}^n (x_{3i}^2y_i)/4,34 = -0,038;$
 $b_0 = \sum_{i=1}^n (x_{0i}y_i)/15,0 - 0,73(b_{11} + b_{22} + b_{33}) = 2,97.$

В приведенных формулах i – номер опыта, n – число опытов в эксперименте, $n = 15$.

Чтобы произвести оценку значимости коэффициентов регрессии необходимо определить дисперсию опыта в нулевой точке плана по формуле:

$$S_y^2 = \sum_{i=1}^K (y_j - \bar{y}_j)^2 / (K - 1),$$

где K – число параллельных опытов в нулевой точке плана, $\bar{y}_j$ – среднее значение y_j , вычисленное для параллельных опытов.

В случае выполнения экспериментов на модели результаты в каждом параллельном опыте не будут отличаться друг от друга, и дисперсия опыта S_y^2 будет равна нулю. При реальных экспериментах дисперсия S_y^2 будет отличаться от нуля, поэтому ее значение задают искусственным путем. При этом считают, что значения y , получен-

ные в каждом опыте, определены с погрешностью A % от среднего уровня $\bar{y}_i$, полученного в результате проведенных 15 опытов при заполнении таблицы ОЦКП. Рассматривая $\Delta = (A \bar{y}_i)/100$ как доверительный интервал, соответствующий 95 %-й доверительной вероятности, находят среднеквадратичное отклонение $S_y = \Delta/t_{ct}$, где $t_{ct} = 1,96$ – табличное значение критерия Стьюдента при числе степеней свободы равном бесконечности. Тогда $S_y^2 = (\Delta/t_{ct})^2$.

Дисперсии коэффициентов регрессии при использовании ОЦКП равны между собой и вычисляются по формуле:

$$S^2\{b_i\} = S_y^2 / 15.$$

Оценка значимости коэффициентов b_i производится по формуле:

$$|b_i| > t_{ct} S\{b_i\}.$$

В случае выполнения неравенства коэффициент b_i считается значимым, в случае невыполнения коэффициент b_i – незначимым и отбрасывается [13].

Примем погрешность $A = 5$ %, тогда: $\bar{y}_i = 3,0$; $\Delta = 0,15$; $S_y^2 = 0,0056$; $S\{b_i\} = 0,061$; коэффициенты b_3 , b_{33} , b_{13} , b_{23} можно считать незначимыми.

После оценки значимости коэффициентов b_i необходимо произвести оценку адекватности регрессионной модели. В том случае, когда данные для построения модели получены на основе вычислительных экспериментов, адекватность модели рекомендуется проверять не по стандартной процедуре при помощи критерия Фишера, а при помощи метода допускового контроля [12, 14]. Модель признается адекватной, если абсолютная величина разности между результатами компьютерных экспериментов и результатами, полученными в результате расчетов по модели

во всех опытных точках $i = 1, 2, \dots, n$ не превышает некоторого допуска δ , то есть имеет место соотношение $|y_i - \hat{y}_i| \leq \delta$, где y_i – определенное из опыта значение выходной величины; $\hat{y}_i$ – рассчитанное по полиному значению выходной величины [13, 14].

Проведенные вычисления показали, что $|y_i - \hat{y}_i| \leq 5\%$ ($\delta = 5\%$) в случае если все коэффициенты b_i считаются значимыми,

$|y_i - \hat{y}_i| \leq 8\%$ ($\delta = 8\%$), если $b_3, b_{33}, b_{13}, b_{23}$ считать незначимыми.

Проведем анализ полученной зависимости для случая, когда $b_3, b_{33}, b_{13}, b_{23}$ считаются незначимыми, то есть влияние такого фактора, как длина шага LS на амплитуду колебаний корпуса робота по углу тангажа не учитывается. График для этого случая приведен на рис. 6.

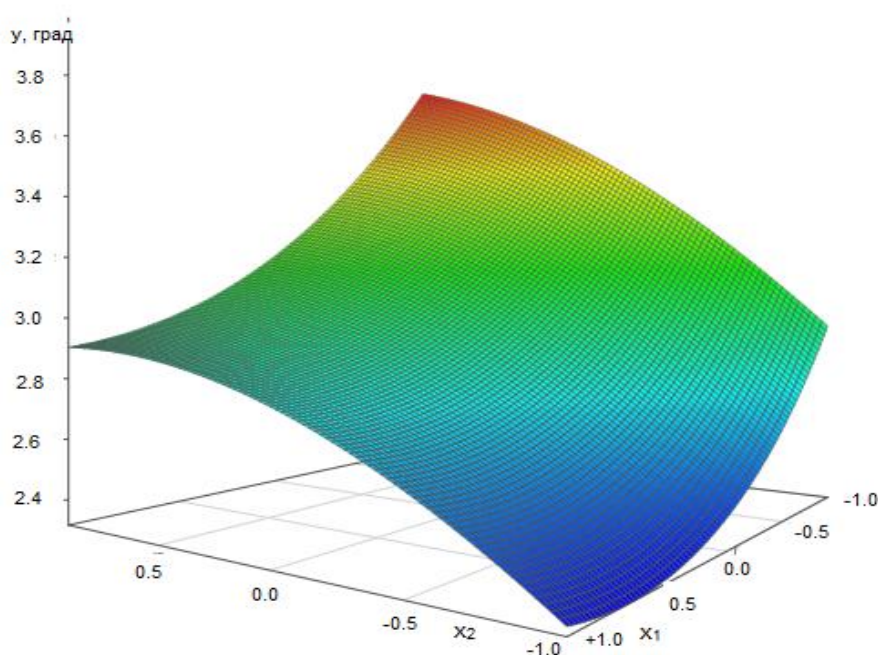


Рис. 6. График функции $y = 2,97 + 0,4 x_1 - 0,43 x_2 - 0,156 x_1^2 + 0,231 x_2^2 - 0,1218 x_1 x_2$
Примечание: составлено авторами.

Амплитуда колебаний по углу крена монотонно убывает при уменьшении высоты шага x_2 и увеличении времени подшага x_1 в установленном диапазоне изменения x_2 и x_1 .

Минимум зависимости найден при помощи функции `fminsearch`, использующей метод прямого поиска Нелдера – Мида среды Matlab. Получены следующие значения: $y_{\min} = 2,3$ град при $x_1 = 0,656$, $x_2 = -1,0$.

После перевода полученных кодированных значений x_1 и x_2 к натуральным $\tilde{x}_1$ и $\tilde{x}_2$ по формулам обратного перехода [12] получены значения: $\tilde{x}_1 = 331$ мс, $\tilde{x}_2 = 0,02$ м.

По приведенной методике получено уравнение регрессионной модели и для амплитуды колебаний по углу тангажа, проведена

оценка значимости коэффициентов и адекватности модели. Минимальное значение амплитуды колебаний составило $y_{\min} = 1,2$ град при времени подшага $\tilde{x}_1 = 352$ мс и высоте подъема ноги $\tilde{x}_2 = 0,02$ м.

ЗАКЛЮЧЕНИЕ

Достоинством регрессионных моделей является наличие хорошо разработанных экономичных алгоритмов их построения с использованием методов планирования эксперимента и возможность последующего анализа поверхности отклика изучаемой функции и ее оптимизации. При этом важным является вопрос об адекватности модели. В работе регрессионная модель построена по данным, полученным

в результате проведения вычислительных экспериментов с моделью робота в среде симулятора Gazebo, оценка погрешности методом допускового контроля показала, что погрешность регрессионной модели составляет менее 8 %.

Полученные результаты анализа поверхности отклика полученных моделей и их минимизация вполне очевидны. Амплитуда колебаний корпуса робота по углам крена и тангажа минимальна при минимальной высоте шага. Зависимость же амплитуды от времени подшага имеет минимум, который и опре-

делен. Полученные числовые данные позволяют на первых этапах разработки робота получить представление о таком важном параметре, как амплитуда колебаний корпуса робота при передвижении: при подъеме стопы на 2 см и за время 1/3 с амплитуда колебаний по углу тангажа составила менее 1,5 градусов.

По мнению авторов, рассматриваемый метод возможно применить при оптимизации параметров системы управления и по другим критериям, например, критерию минимизации расходуемой энергии.

Список источников

1. Филимонов А. Б., Филимонов Н. Б. Ситуационный подход в задачах автоматизации управления техническими объектами // Мехатроника, автоматизация, управление. 2018. Т. 19, № 9. С. 563–578. DOI 10/15787/mau.19.563-578.
2. Пикалев Я. С. Анализ существующих симуляторов робототехнических систем // Проблемы искусственного интеллекта. 2017. № 1 (4). С. 51–65.
3. Самарин В. А., Лавров А. А., Динамическое моделирование робототехнических систем в программном пакете GAZEBO // Информатика и кибернетика (ComCon-2016). СПб. : Изд-во СПбПУ, 2016. С. 56–58.
4. Васильянов Г. С., Лавров А. А., Разработка и исследование алгоритмов адаптивного управления четырехногими роботизированными платформами // Экстремальная робототехника. 2015. № 1. С. 323–332.
5. Choi H.-Y., Park J., Lee J. Quality Engineering Optimization of Robot Casting Considering Design Robustness: Comparison with Reliability // Int J Precis Eng Manuf. 2013. Vol 14, Is. 12. P. 2157–2163. DOI 10.1007/s12541-013-0292-7.
6. Zare S., Karimi-Nasab S., Jalalifar H. Analysis and Determination of the Behavioral Mechanism of Rock Bridges Using Experimental and Numerical Modeling of Non-Persistent Rock Joints // Int J Rock Mech Min Sci. 2021. Vol. 141. P. 104714.
7. Николаев Н. Н. Основы научных исследований на транспорте, планирование экспериментов и инженерных наблюдений. Зерноград : ФГОУ ВПО АЧГАА, 2007. 37 с.
8. Sun J., Zhang W., Dong X. Natural Frequency Prediction Method for 6R Machining Industrial Robot // Appl Sci. 2020. Vol. 10, Is. 22. P. 8138. DOI 10.3390/app10228138.
9. Завьялов В. М., Шпакова Л. Г., Шпаков М. А. Параметрическая оптимизация автоматизированного электропривода звена робота-манипулятора // Тр. Караганд. технич. ун-та. 2020. № 3 (80). С. 143–147.

References

1. Filimonov A. B., Filimonov N. B. Situational Approach in the Problems of Automation Control by Technical Objects // Mekhatronika, avtomatizatsiia, upravlenie. 2018. Vol. 19, No. 9. P. 563–578. DOI 10/15787/mau.19.563-578. (In Russian).
2. Pikalev Ya. S. Analysis of Existing Simulators for Robotic Systems // Problems of Artificial Intelligence. 2017. No. 1 (4). P. 51–65. (In Russian).
3. Samarin V. A., Lavrov A. A. Dinamicheskoe modelirovanie robototekhnicheskikh sistem v programnom pakete GAZEBO // Informatika i kibernetika (ComCon-2016). St. Petersburg : Publishing House Peter the Great St. Petersburg Polytechnic University, 2016. P. 56–58. (In Russian).
4. Vasilyanov G. S., Lavrov A. A. Development and Research of Algorithms of Adaptive Management of the Quadruped Robotized Platforms // Extreme Robotics. 2015. No. 1. P. 323–332. (In Russian).
5. Choi H.-Y., Park J., Lee J. Quality Engineering Optimization of Robot Casting Considering Design Robustness: Comparison with Reliability // Int J Precis Eng Manuf. 2013. Vol 14, Is. 12. P. 2157–2163. DOI 10.1007/s12541-013-0292-7.
6. Zare S., Karimi-Nasab S., Jalalifar H. Analysis and Determination of the Behavioral Mechanism of Rock Bridges Using Experimental and Numerical Modeling of Non-Persistent Rock Joints // Int J Rock Mech Min Sci. 2021. Vol. 141. P. 104714.
7. Nikolaev N. N. Osnovy nauchnykh issledovaniy na transporte, planirovanie eksperimentov i inzhenernykh nabliudenii. Zernograd : FGOU VPO AChGAA, 2007. 37 p. (In Russian).
8. Sun J., Zhang W., Dong X. Natural Frequency Prediction Method for 6R Machining Industrial Robot // Appl Sci. 2020. Vol. 10, Is. 22. P. 8138. DOI 10.3390/app10228138.
9. Zavyalov V. M., Shpakova L. G., Shpakov M. A. Parametricheskaya optimizatsiia avtomatizirovannogo elektroprivoda zvena robota-manipuliatora // Tr. Karagand. tekhnich. un-ta. 2020. No. 3 (80). P. 143–147. (In Russian).

10. Capelli B., Sabattini L. Human-Friendly Multi-Robot Systems: Legibility Analysis // Proceedings of the Human-Friendly Robotics, October 24–25, 2019, Reggio Emilia, Italy. 2020. P. 15–28.
11. Шеперд Г. М. Нейробиология. М. : Мир, 1987. Т. 2. 368 с.
12. Ивоботенко Б. А., Ильинский Н. Ф., Копылов И. П. Планирование эксперимента в электромеханике. М. : Энергия, 1975. 185 с.
13. Электронная библиотека. Ортогональные центральные композиционные планы. URL: <https://libraryno.ru> (дата обращения: 10.11.2022).
14. Егоров А. Е., Азаров Г. Н., Коваль А. В. Исследование устройств и систем автоматики методом планирования эксперимента / под ред. В. Г. Воронова. Харьков : Вища школа, 1986. 240 с.
10. Capelli B., Sabattini L. Human-Friendly Multi-Robot Systems: Legibility Analysis // Proceedings of the Human-Friendly Robotics, October 24–25, 2019, Reggio Emilia, Italy. 2020. P. 15–28.
11. Shepherd G. M. Neurobiology. Moscow : Mir, 1987. Vol. 2. 368 p. (In Russian).
12. Ivobotenko B. A., Ilyinsky N. F., Kopylov I. P. Planirovaniye eksperimenta v elektromekhanike. Moscow : Energiia, 1975. 185 p. (In Russian).
13. Elektronnaia biblioteka. Ortogonalnye tsentralnye kompozitsionnye plany. URL: <https://libraryno.ru> (accessed: 10.11.2022). (In Russian).
14. Egorov A. E., Azarov G. N., Koval A. V. Issledovanie ustroystv i sistem avtomatiki metodom planirovaniia eksperimenta / Ed. V. G. Voronov. Kharkov : Vishcha shkola, 1986. 240 p. (In Russian).

Информация об авторах

Г. С. Васильянов – ассистент, ведущий инженер научной лаборатории.

А. А. Лавров – кандидат технических наук, доцент.

А. В. Лупин – старший преподаватель.

И. А. Малышев – старший преподаватель.

Information about the authors

G. S. Vasilyanov – Assistant Professor, Leading Engineer of the Scientific Laboratory.

A. A. Lavrov – Candidate of Sciences (Engineering), Associate Professor.

A. V. Lupin – Senior Lecturer.

I. A. Malyshev – Senior Lecturer.

Научная статья

УДК 519.876.5:656+004.94

doi: 10.34822/1999-7604-2022-4-23-28

ЭЛЕМЕНТЫ ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ В ВОПРОСАХ ОПТИМИЗАЦИИ ДОРОЖНОГО ДВИЖЕНИЯ

Наталья Сергеевна Веремчук

Сибирский государственный автомобильно-дорожный университет, Омск, Россия

n-veremchuk@rambler.ru, <https://orcid.org/0000-0002-2709-9755>

Аннотация. Представлена разработка методами агентного и дискретно-событийного моделирования имитационной модели элемента дорожной сети, состоящего из двух смежных перекрестков города Омска, для анализа движения транспортных средств, прогнозирования транспортных потоков и разработки мер по оптимизации дорожного трафика. По результатам вычислительных экспериментов исследования транспортных потоков на выбранном участке дорожной сети с целью улучшения ее функционирования предложена оптимальная схема проезда перекрестков транспортными средствами за счет изменения длительности фаз светофоров.

Ключевые слова: дискретно-событийное моделирование, дорожная сеть, имитационная модель, оптимизационный эксперимент

Для цитирования: Веремчук Н. С. Элементы имитационного моделирования в вопросах оптимизации дорожного движения // Вестник кибернетики. 2022. № 4 (48). С. 23–28. DOI 10.34822/1999-7604-2022-4-23-28.

Original article

ELEMENTS OF SIMULATION MODELING IN ISSUES OF TRAFFIC OPTIMIZATION

Natalya S. Veremchuk

The Siberian State Automobile and Highway University, Omsk, Russia

n-veremchuk@rambler.ru, <https://orcid.org/0000-0002-2709-9755>

Abstract. The study presents the development of a simulation model for an element of the road network that consists of two adjacent intersections in Omsk using agent-based and discrete-event modeling in order to analyze traffic, forecast traffic, and propose traffic optimization measures. Based on the traffic calculations aimed at improving functionality of the selected element of the road network, an efficient scheme for traffic flows is proposed by changing the duration of traffic light phases.

Keywords: discrete-event modeling, road network, simulation model, optimization experiment

For citation: Veremchuk N. S. Elements of Simulation Modeling in Issues of Traffic Optimization // Proceedings in Cybernetics. 2022. No. 4 (48). P. 23–28. DOI 10.34822/1999-7604-2022-4-23-28.

ВВЕДЕНИЕ

В век информационных технологий прогнозирование приобрело всеобщую значимость и находит применение во многих отраслях – в торговле, медицине, инженерии. В последние годы в прогнозировании активно применяется моделирование, которое позволяет проверить эффективность решения

до его реализации без риска потери времени, финансов и угрозы жизни людей [1].

Средствами имитационного моделирования осуществляется воспроизведение работы изучаемой системы или процесса во времени при различных режимах функционирования [2, 3]. Это удобный инструмент для анализа, поскольку обладает такими качествами,

как наглядность, простота для понимания и проверки. Имитационное моделирование помогает найти оптимальные решения в разных областях производства и науки, а также позволяет получить более четкое представление о сложных системах [4]. Имитационные модели широко используются в процессах проектирования сложных многофункциональных систем, и применение имитационного подхода возможно в любой сфере человеческой деятельности [2–8].

Плюсом имитационного моделирования в отличие от других методов является возможность проверки и анализа модели в действии и внесения в создаваемую модель изменений в процессе ее разработки.

Пакет AnyLogic – это профессиональная отечественная система имитационного моделирования, предназначенная для построения и анализа различного типа моделей [1, 2].

В данной работе в среде AnyLogic с применением агентного и дискретно-событийного моделирования построена имитационная модель двух перекрестков города Омска, проведены эксперименты по анализу и прогнозированию транспортных потоков на выбранном участке дорожной сети, предложена оптимизация схемы проезда перекрестков транспортными средствами за счет изменения длительности фаз светофоров.

МАТЕРИАЛЫ И МЕТОДЫ

В литературе выделяют три основных вида имитационного моделирования: агентное, дискретно-событийное и системную динамику. Агентное моделирование применяется для анализа и прогнозирования поведения сложных систем. Агент может взаимодействовать с окружающим миром, принимать решения по определенному набору правил. Главной задачей агентных моделей является получение информации об общем поведении системы, и часто такой подход используется в биологии, социологии, экологии.

Основным способом определения причинно-следственных связей и взаимовлияния элементов системы является системная динамика, используемая в различных процессах производства товаров и услуг. При этом осуществляется абстрагирование от конкретных

объектов и событий и «агрегатный» подход к оценке системы и исследуемым процессам.

В случае необходимости вычленения конкретных действий из общей последовательности событий применяется метод дискретно-событийного моделирования. Часто такой подход используется в производственных циклах, когда важно оценить результат деятельности только определенных участков производства. Дискретно-событийное моделирование используется для построения модели, которая будет отражать развитие системы во времени.

Опишем построение методами агентного и дискретно-событийного моделирования имитационной модели двух смежных перекрестков, расположенных на пересечении улиц города Омска: 7-я Северная – Орджоникидзе, 7-я Северная – Герцена (рис. 1).

Для имитации транспортных потоков и элементов дорожного движения использовались спутниковый снимок и библиотека моделирования дорожного движения в AnyLogic PLE, с помощью которых настраивались дорога, количество полос, разделительные линии, ширина дорожного полотна и другие характеристики, соответствующие реальным перекресткам. В построенной модели откорректированы возможные направления движения транспортных средств с учетом соблюдения правил дорожного движения (рис. 2).

Имитация транспортных потоков и процесса дорожного движения также проводилась с использованием блоков библиотеки [1]: «CarSource» моделировал появление транспортного средства на дороге; «CarMoveTo» – движение транспортных средств; «CarDispose» – окончание проезда транспортными средствами перекрестков. Диаграммы транспортных потоков, полученных на основе дискретно-событийной методологии, задающие возможные направления движения транспортных средств, представлены на рис. 3.

Для регулирования трафика в модель из библиотеки введен блок «TrafficLight», соответствующий светофору, в окне настроек которого указаны перекресток, длительность фаз светофора и другие характеристики режима его работы (рис. 4). Исходные данные для длительности фаз светофора установлены по результатам натурных наблюдений.

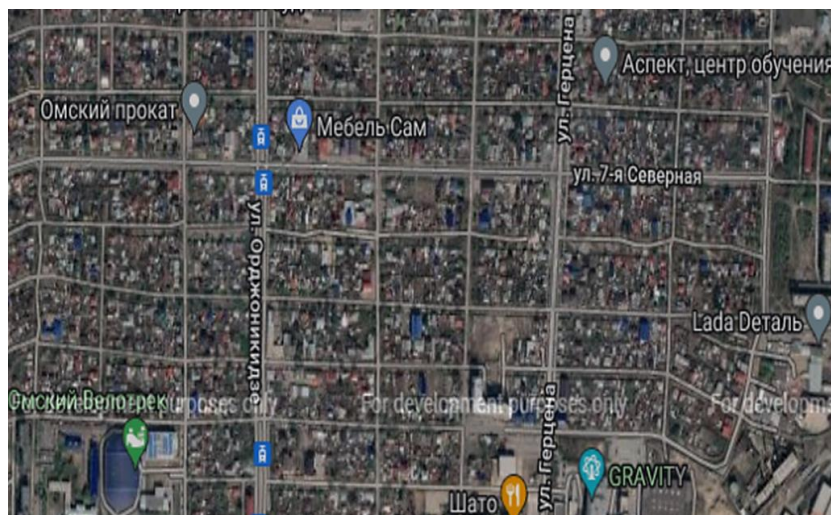


Рис. 1. Спутниковый снимок перекрестков

Примечание: скриншот автора.

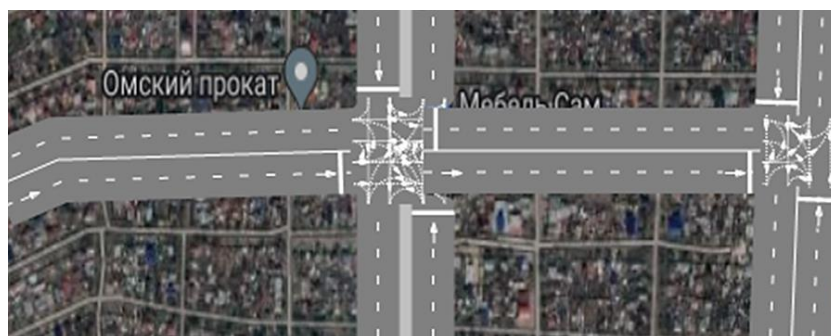


Рис. 2. Модель дорожной сети

Примечание: скриншот автора.

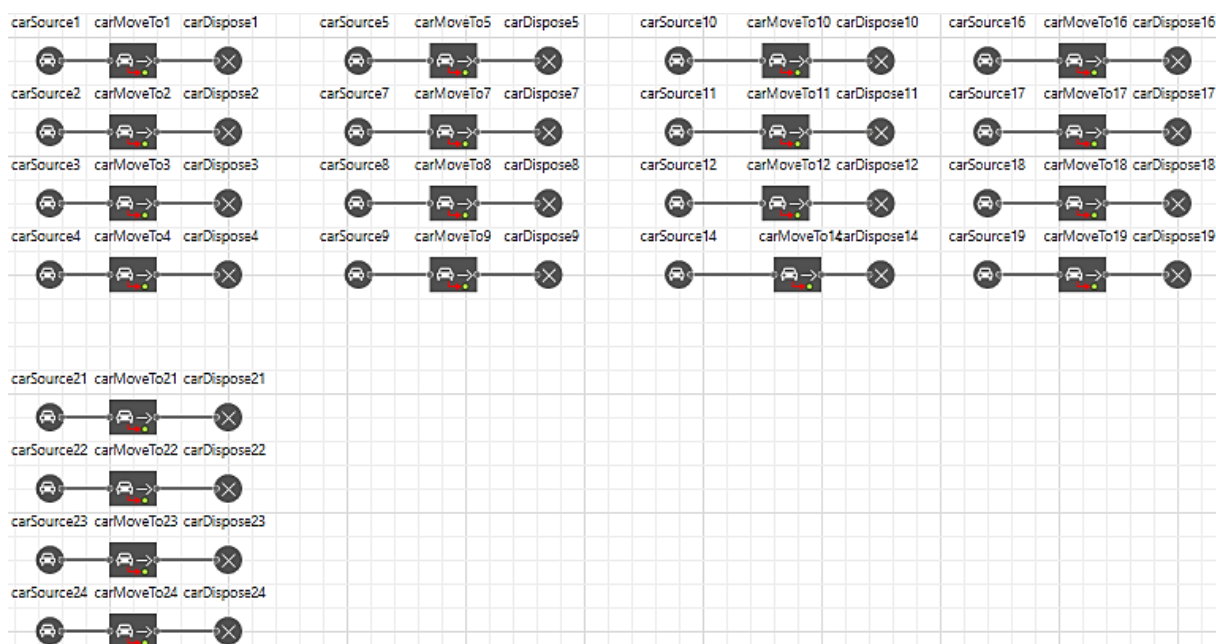


Рис. 3. Диаграммы транспортных потоков

Примечание: составлено автором.

trafficLight - Traffic Light

Имя: ☒ Отображать имя

☐ Исключить

Задаёт режим работы для: ☒ Стоп-линий перекрестка
☐ Соединителей полос перекрестка
☐ Заданных стоп-линий

Перекресток:

Фазы:

Длительности, сек:	50	3	60	3	
Стоп-линии:					
stopLine					
stopLine1					
stopLine2					
stopLine3					

Рис. 4. Окно свойств светофора
 Примечание: скриншот автора.

В модели создана популяция из агентов – автомобилей с именем «Car» с заданным каждому агенту параметром «ВремяПоявления». Популяция указана во всех блоках «CarSource». В поля основных характеристик агентов «Car» введены средние значения, рассчитанные в соответствии с правилами скоростного режима в населенном пункте.

Для сбора статистики в модель добавлена гистограмма с помощью одноименного элемента из библиотеки [1]. На гистограмме отражается среднее время проезда транспортными

средствами двух перекрестков в секундах (рис. 5).

Практика показывает, что транспортные средства на этих перекрестках проводят довольно много времени в пробках, и одна из возможных причин этому – неправильно подобранные временные фазы светофоров. На основе проведенных экспериментов в имитационной модели выполнена оптимизация длительности фаз светофоров с использованием среднего времени проезда транспортными средствами двух перекрестков в качестве целевой функции.

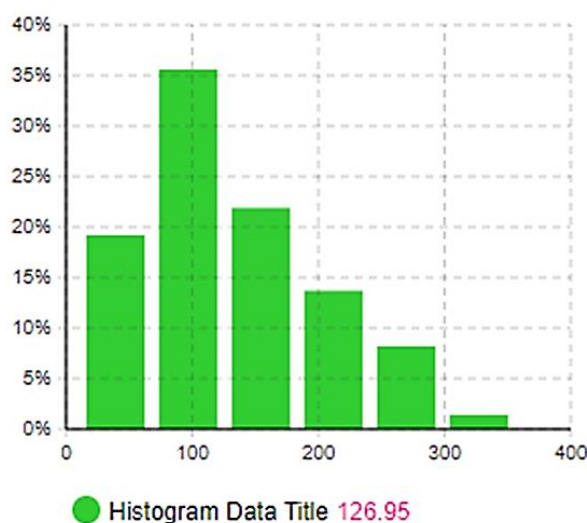


Рис. 5. Гистограмма, отражающая среднее время проезда
 Примечание: скриншот автора.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Для определения оптимальной длительности фаз светофоров в модели добавлены четыре новых параметра (p , $p1$, $p2$, $p3$), отвечающих за настройку их длительности. Создан эксперимент с типом оптимизация и настройка новых свойств. В качестве целевой функции принята минимизация среднего времени проезда двух перекрестков. Изменен тип

всех параметров с фиксированного на дискретный. Установлены границы изменения параметров: минимальное значение для всех параметров – 20 с, максимальное – 80 с, шаг смены параметров – 5 с. Перед запуском эксперимента задано модельное время в 10 минут. Окно настроек эксперимента по оптимизации отражено на рис. 6.

Optimizatia - Оптимизационный эксперимент

Имя: ☐ Исключить

Агент верхнего уровня:

Целевая функция: ☒ минимизировать ☐ максимизировать

☒ Количество итераций:

☐ Автоматическая остановка

Максимальный размер памяти: M6

Параметры

Параметры:

Параметр	Тип	Значение			
		Мин.	Макс.	Шаг	Начальное
p	дискретный	20	80	5	
p1	дискретный	20	80	5	
p2	дискретный	20	80	5	
p3	дискретный	20	80	5	

Рис. 6. Настройка оптимизационного эксперимента

Примечание: скриншот автора.

По результатам эксперимента установлено, что изменение длительности фаз светофоров на оптимальные значения позволяет уменьшить время проезда двух перекрестков с изначальных 135,45 с до 102,71 с (рис. 7). Лучший результат был получен при значениях фаз: первого светофора – $p = 20$ с, $p1 = 65$ с; вто-

рого светофора – $p2 = 75$ с, $p3 = 20$ с. Вычисленная длительность фаз светофоров позволяют увеличить пропускную способность исследуемого элемента дорожной сети и оптимизировать время его проезда транспортными средствами.

Перекресток21 : Optimizatia

	Текущее	Лучшее
Итерация:	500	31
Функционал	135.452	102.71
Параметры	Copy best	
p	25	20
p1	60	65
p2	70	75
p3	20	20

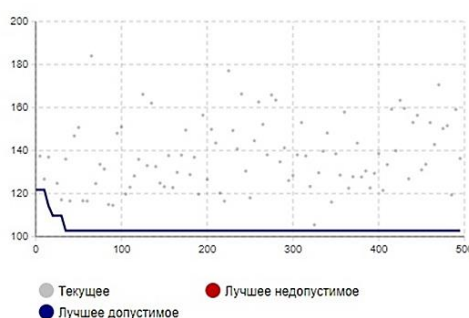


Рис. 7. Результат эксперимента по оптимизации

Примечание: скриншот автора.

ЗАКЛЮЧЕНИЕ

В среде Anylogic PLE с использованием агентного и дискретно-событийного моделирования разработана и реализована имитационная модель двух перекрестков города Омска. Проведен анализ движения транспортных потоков по выбранному элементу дорожной сети. Выполнена оптимизация длительности фаз светофоров. В результате эксперимента

увеличена пропускная способность двух смежных перекрестков и оптимизировано время их проезда транспортными средствами.

Полученные результаты могут быть использованы для анализа и прогнозирования транспортных потоков с целью улучшения функционирования дорожной сети и оптимизации дорожного движения.

Список источников

1. Боев В. Д. Компьютерное моделирование в среде AnyLogic. М. : Юрайт, 2021. 298 с.
2. Веремчук Н. С., Кондрин А. В. О разработке имитационной модели пешеходного перехода // Образование. Транспорт. Инновации. Строительство : сб. материалов IV Национал. науч.-практич. конф. 22–23 апреля 2021 г., Омск. Омск : Сиб. гос. авто-мобил.-дорож. ун-т, 2021. С. 724–727.
3. Веремчук Н. С. Об имитационном моделировании элементов дорожной сети // Перспективы науки. 2021. № 12. С. 45–48.
4. Akopov A. S., Beklaryan L. A., Beklaryan A. L. Cluster-Based Optimization of an Evacuation Process Using a Parallel Bi-Objective Real-Coded Genetic Algorithm // Cybernetics and Information Technologies. 2020. Vol. 20, No 3. P. 45–63.
5. Beklaryan L. A., Khachatryan N. K., Akopov A. S. Model for Organization Cargo Transportation at Resource Restrictions // Int J Appl Math. 2019. Vol. 32, No. 4. P. 627–640.
6. Ordu M., Demir E., Tofallis C., Gunal M. M. A Novel Healthcare Resource Allocation Decision Support Tool: A Forecasting-Simulation-Optimization Approach // J Oper Res Soc. 2021. Vol. 72, Is. 3. P. 485–500.
7. Вегера Д. В., Жиба Г. В., Писаренко В. П. и др. Оценка распространения сигнала LTE на сложной по рельефу загородной трассе с учетом хвойного леса // Вестник кибернетики. 2021. № 3 (43). С. 23–29.
8. Лышов С. М., Иванов И. А., Увайсова А. С. и др. Расчет разбросов резонансных частот печатных узлов электронных средств // Вестник кибернетики. 2018. № 4 (32). С. 129–135.

Информация об авторе

Н. С. Веремчук – кандидат физико-математических наук, доцент.

References

1. Boev V. D. Kompiuternoe modelirovanie v srede AnyLogic. Moscow : Iurait, 2021. 298 p. (In Russian).
2. Veremchuk N. S., Kondrin A. V. On the Development of a Simulation Model of a Pedestrian Crossing // Obrazovanie. Transport. Innovatsii. Stroitelstvo : Proceedings of the IV National Research-to-Practice Conference, April 22–23, 2021, Omsk. Omsk : The Siberian State Automobile and Highway University, 2021. P. 724–727. (In Russian).
3. Veremchuk N. S. Ob imitatsionnom modelirovanii elementov dorozhnoi seti // Science Prospects. 2021. No. 12. P. 45–48. (In Russian).
4. Akopov A. S., Beklaryan L. A., Beklaryan A. L. Cluster-Based Optimization of an Evacuation Process Using a Parallel Bi-Objective Real-Coded Genetic Algorithm // Cybernetics and Information Technologies. 2020. Vol. 20, No 3. P. 45–63.
5. Beklaryan L. A., Khachatryan N. K., Akopov A. S. Model for Organization Cargo Transportation at Resource Restrictions // Int J Appl Math. 2019. Vol. 32, No. 4. P. 627–640.
6. Ordu M., Demir E., Tofallis C., Gunal M. M. A Novel Healthcare Resource Allocation Decision Support Tool: A Forecasting-Simulation-Optimization Approach // J Oper Res Soc. 2021. Vol. 72, Is. 3. P. 485–500.
7. Vegera D. V., Zhiba G. V., Pisarenko V. P. et al. Evaluation of the LTE Signal Propagation along the Highway with a Difficult Terrain in a Coniferous Forest // Proceedings in Cybernetics. 2021. No. 3 (43). P. 23–29. (In Russian).
8. Lyshov S. M., Ivanov I. A., Uvaisova A. S. et al. Calculation of Resonant Frequencies Spread of Printed Circuit Assembly of Electronic Devices // Proceedings in Cybernetics. 2018. No. 4 (32). P. 129–135. (In Russian).

Information about the author

N. S. Veremchuk – Candidate in Sciences (Physics and Mathematics), Associate Professor.

Научная статья
УДК 004.9:517.1+81'37
doi: 10.34822/1999-7604-2022-4-29-36

АНАЛИЗ АНГЛИЙСКОГО ЛЕКСИКОНА ИНТЕНСИВНОСТИ ЭМОЦИЙ И ФОРМИРОВАНИЕ АЛГОРИТМА ПОСТРОЕНИЯ ЕГО РУССКОЯЗЫЧНОЙ ВЕРСИИ

Александр Романович Гончаров^{1✉}, Светлана Александровна Лысенкова²

^{1,2} Сургутский государственный университет, Сургут, Россия

¹ mr.tesanta@ya.ru ✉

² lsa1108@mail.ru

Аннотация. Представлен сравнительный анализ значений основных эмоций двух версий National Research Council Canada Emotion Intensity Lexicon и выдвинуто предположение о различии интенсивности эмоциональных норм в русской культуре. Для создания русскоязычного словаря эмоций предложен алгоритм построения, основанный на использовании синонимичных слов при переводе. Рассмотрены онлайн-сервисы многовариантности перевода слов по их качеству и разнообразию для выбора соответствия задачам исследования и определения количественной разницы коэффициентов интенсивности эмоций в русскоязычном и англоязычном словарях.

Ключевые слова: сравнительный анализ, словарь эмоций, сервис множественного перевода, сентимент-анализ, наилучшее-худшее масштабирование

Для цитирования: Гончаров А. Р., Лысенкова С. А. Анализ английского лексикона интенсивности эмоций и формирование алгоритма построения его русскоязычной версии // Вестник кибернетики. 2022. № 4 (48). С. 29–36. DOI 10.34822/1999-7604-2022-4-29-36.

Original article

ANALYZING THE ENGLISH EMOTION INTENSITY LEXICON AND GENERATING AN ALGORITHM TO COMPILE ITS RUSSIAN VERSION

Aleksandr R. Goncharov^{1✉}, Svetlana A. Lysenkova²

^{1,2} Surgut State University, Surgut, Russia

¹ mr.tesanta@ya.ru ✉

² lsa1108@mail.ru

Abstract. The article presents a comparative analysis of two versions of the National Research Council Canada Emotion Intensity Lexicon and proposes a hypothesis regarding the difference in intensity of Russian emotional norms. Based on the use of synonym words during translation, an algorithm for compiling a Russian-language dictionary of emotions is proposed. Online multiple translation services are investigated according to their quality and variety in order to select the one that will suit the research goal and determine the quantitative difference of emotion intensity coefficients in Russian and English lexicons.

Keywords: comparative analysis, lexicon of emotions, multiple translation service, sentiment analysis, best-worst scaling

For citation: Goncharov A. R., Lysenkova S. A. Analyzing the English Emotion Intensity Lexicon and Generating an Algorithm to Compile Its Russian Version // Proceedings in Cybernetics. 2022. No. 4 (48). P. 29–36. DOI 10.34822/1999-7604-2022-4-29-36.

ВВЕДЕНИЕ

Одним из важных направлений автоматизированной обработки естественного языка и компьютерной лингвистики является изуче-

ние аффективных состояний. Под аффектом подразумевается бессознательная форма проявления эмоции, которая формируется в сознании человека и не может быть отнесена

ни к положительным, ни к отрицательным чувствам [1].

В области сентимент-анализа часто прибегают к исследованиям эмоций, которые позволяют вывести окрашенные лексикой положительные, негативные или нейтральные суждения субъекта о содержании какого-либо документа. Такие сведения позволяют не только систематизировать текстовые источники по оценкам читателей, но также вывести закономерности их предпочтений, поддержки принятых решений и разработки диалоговых систем.

Одним из подходов к распознаванию эмоциональной оценки текста является метод, основанный на применении тональных словарей, где каждому слову в словаре присваивается значение тональности и интенсивной окраски в той или иной группе эмоций. Простым способом, к которому часто прибегают в определении тональности, является подсчет среднего арифметического из суммы значений тональности всех слов в документе [2].

Сравнительный анализ и алгоритм.

The National Research Council Canada Emotion Intensity Lexicon (NRC-EIL) является одним из словарей эмоций [3]. Словарь создавался вручную при помощи краудсорсинга

и к августу 2022 года содержит более 9 000 английских слов, каждое из которых отнесено в одну из восьми категориальных групп эмоций Плутчика [4]. В словарь входят термины N-грамм Google, а также часто встречающиеся слова, собранные из постов пользователей социальной сети «Twitter» по определенным хештегам, например, #angry, #fear и т. д. (табл. 1). Данный состав из распространенных в общении в социальной сети английских слов и лексики позволяет классифицировать мнение пользователей для оценки тенденций продвижения как коммерческих продуктов, так и создаваемого контента.

Словарь NRC-EIL был переведен на более чем 100 других языков при помощи сервиса «GoogleTranslate» с расчетом, что при переводе слово или термин в другой культуре будет иметь такой же коэффициент интенсивности эмоций, как и в исходной англоязычной [6]. Однако известно, что в русском языке совершенно разные слова могут быть одновременно синонимичными и иметь одинаковую эмоциональную ценность, а также менять смысл в зависимости от их использования в общественных, культурных или религиозных диалогах [7].

Таблица 1

Состав словаря NRC-EIL, 2022 г.

Группа эмоций	Количество слов в группе
anger	1 481
anticipation	862
disgust	1 092
fear	1 763
joy	1 264
sadness	1 294
surprise	583
trust	1 490

Примечание: составлено авторами по источнику [5].

Рассмотрим актуальную (август 2022 г.) версию словаря NRC-EIL, его встречающуюся устаревшую версию (2018 г.) и изменение их объема в каждой категории эмоции (табл. 2).

Главным отличительным признаком актуальной версии является повышение качества перевода на другие языки. В ходе сравнитель-

ного анализа было выявлено около 4 000 позиций различий перевода с изменением окончания слов, части речи или новых слов, которые раньше по разным причинам переведены не были, при этом различий в коэффициентах интенсивности эмоций в категориях не выявлено (рис. 1).

Таблица 2

Анализ объема словарей NRC-EIL различных версий

Группа эмоций	NRC-EIL v2018	NRC-EIL v2022	Разница слов	Разница слов (%)	Добавлено новых слов
anger	1 483	1 481	2	0,1349	0
anticipation	864	862	2	0,2315	0
disgust	1 094	1 092	2	0,1828	0
fear	1 765	1 763	2	0,1133	0
joy	1 268	1 264	4	0,3155	0
sadness	1 298	1 294	4	0,3082	0
surprise	585	583	2	0,3419	0
trust	1 564	1 490	74	4,7315	0

Примечание: составлено авторами по источнику [5].

	en	ru-2018	ru-2022	emotion-intensity-score
0	outraged	возмущен	возмущенный	0.964
1	brutality	зверство	жестокость	0.959
2	violently	сильно	яростно	0.938
3	infuriated	разъяренный	взбешенный	0.938
4	furiously	бешено	яростно	0.927
5	screwyou	NO TRANSLATION	пошел вон	0.924
6	fury	неистовство	ярость	0.922
7	execution	выполнение	исполнение	0.917
8	angered	разозлило	рассерженный	0.916

Рис. 1. Пример различия в русскоязычных переводах версий NRC-EIL

Примечание: составлено авторами по источнику [5].

Учитывая языковые особенности, проведена проверка соответствия теории стабильных коэффициентов интенсивности эмоций для русскоязычных переводов словаря NRC-EIL с помощью расчета этих коэффициентов для слов русского языка при формировании нового словаря по следующему алгоритму:

1. Составление тренировочных кортежей с применением синонимичных слов.
2. Выделение «золотых вопросов».
3. Проведение анкетирования способом Best-Worst Scaling (BWS).
4. Подсчет и нормирование веса тональности слов.

МАТЕРИАЛЫ И МЕТОДЫ

Для расчета коэффициентов интенсивности с использованием краудсорсинга и анкетирования по принципу масштабирования

«наилучшее/наихудшее» (best-worst scaling, BWS) каждому участнику предоставляется кортеж из четырех слов для выбора различных вариантов наихудшего и наилучшего элемента по предложенному свойству. Каждое слово должно встречаться в восьми разных кортежах, причем кортежей для анкетирования одной эмоции должно быть в два раза больше, чем слов в данной группе. Например, если в группе 1 483 слова, то количество 4-элементных кортежей составит 2 966.

Такой способ дает пять из шести сравнительных пар элементов, что упрощает саму процедуру анкетирования в отличие от матрицы попарного сравнения. Например, для кортежа из слов (A, B, C, D) при выборе варианта A – как наилучшее и D – как наихудшее получатся сравнительные пары $A > B$,

$A > C$, $A > D$, $B > D$ и $C > D$, где один элемент превосходит другой [8].

Подсчет весов и нормирование коэффициентов ω_w для каждого слова w осуществляется по формуле [3], смысл которой заключается в нахождении разницы между долями выбранных как наихудшие и наилучшие события, предоставленные респондентом:

$$\omega_w = \frac{L_w}{N_w} - \frac{X_w}{N_w} \in [-1; 1], \quad (1)$$

где N_w – количество оценок слова w ;

L_w – количество w наилучшее;

X_w – количество w наихудшее.

Итоговая оценка нормируется в пределах $[0; 1]$.

В основу составления лексикона NRC-EIL легли часто используемые английские слова и термины, распространенные в текстах социальной сети «Twitter», причем все элементы в словаре уникальны, однако при переводе на русский язык встречаются дублирующие элементы (табл. 3).

Таблица 3

Дубликаты русскоязычного перевода в NRC-EIL

Группа эмоций	Количество дубликатов перевода
anger	199
anticipation	57
disgust	159
fear	202
joy	139
sadness	166
surprise	54
trust	122

Примечание: составлено авторами по источнику [5].

Наличие дублирующих записей в русскоязычном переводе словаря и одинаковые элементы в кортежах (рис. 2), предложенных для

оценки участнику анкетирования, приведет к неясности и в дальнейшем к несогласованным результатам.

	en	emotion	emotion-intensity-score	ru
1	happiness	joy	0.984	счастье
40	laugh	joy	0.891	смех
93	laughter	joy	0.824	смех
95	goodness	joy	0.818	доброта
373	felicity	joy	0.636	счастье
419	kindness	joy	0.606	доброта

Рис. 2. Пример повторяющихся слов в русскоязычном переводе NRC-EIL

Примечание: составлено авторами по источнику [5].

Одним из способов решения проблемы является предоставление респонденту модернизированного кортежа, где каждый элемент – строка с множественным переводом оригинального слова, что позволит различать элементы при анкетировании. Пример такого элемента кортежа может иметь следующий вид:

$$\langle A_0, A_1, A_2, \dots, A_N \rangle, \quad (2)$$

где A_0 – оригинальное слово;

A_i – слово множественного перевода, $i = \overline{1, N}$.

Для автоматизации этой задачи были рассмотрены переводы онлайн-сервисов «Woord Hunt.ru», «Linguee» и «Reverso Context», где для каждого оригинального слова из словаря

NRC-EIL был произведен русскоязычный множественный перевод глубиной до четырех слов и рассчитаны неудачные варианты перевода (табл. 4), возникающие в результате

неспособности сервиса распознать лексику, применяемую при общении в социальных сетях, и наибольшее количество таких ошибок показал сервис «Linguee».

Таблица 4

Количество нераспознанных слов онлайн-сервисом для перевода

Группа эмоций	WoordHunt.ru	Linguee	Reverso Context
anger	204	1 478	86
anticipation	21	837	0
disgust	45	1 092	6
fear	183	1 759	62
joy	263	1 217	121
sadness	76	1 293	22
surprise	17	571	5
trust	99	1 482	12

Примечание: составлено авторами по источнику [5].

Дальнейший анализ показал, что первое слово в новом переводе часто совпадает с оригинальным переводом в словаре NRC-EIL. После удаления повторяющихся слов и объединения переводов всех онлайн-сервисов была

составлена таблица дубликатов (табл. 5), где отражается преимущество сервиса «Reverso Context» в задаче формирования кортежей с синонимичными словами множественного перевода.

Таблица 5

Дубликаты в новом русскоязычном переводе

Группа эмоций	WoordHunt.ru	Reverso Context
anger	9	6
anticipation	1	1
disgust	2	1
fear	6	1
joy	20	6
sadness	1	0
surprise	3	1
trust	4	0

Примечание: составлено авторами по источнику [5].

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В ходе исследования был проведен сравнительный анализ версий словаря эмоций NRC-EIL 2018 и 2022 гг., в том числе различия в русскоязычных переводах слов, который показал целесообразность использования его актуальной версии 2022 года. В результате изучения «уникальных» слов, используемых в опреде-

ленной языковой культуре, выдвинуто предположение о различии эмоциональных норм (коэффициентов интенсивности эмоций слов) при составлении аналогичного словаря в русской культуре. Примером этого может быть название тропического циклона «Сэнди», который в американском обществе прозвали ураганом «Франкеншторм» (рис. 3).

en	emotion	emotion-intensity-score	ru
389	frankenstorm	fear	0.672 Франкеншторм

Рис. 3. Пример «уникального» слова в языковой культуре

Примечание: составлено авторами по источнику [5].

Учитывая распространение этого циклона на определенной территории, предполагается, что в русской культуре его обозначение не будет иметь такой же коэффициент интенсивности эмоций, какое оно имеет в оригинале, и для проверки предположения представлена идея составления аналогичного русскоязычного словаря эмоций.

Поскольку в словаре NRC-EIL при переводе часто встречаются повторения значений слов и терминов, разработан алгоритм построения нового словаря, основанный на использовании синонимичных слов (рис. 2) и дополненный вариантами перевода онлайн-сервисом «Reverso Context» (рис. 4).

	en	emotion	emotion-intensity-score	ru	reverso
6	elation	joy	0.944	восторг	эйфория, радость, душевный подъем
10	celebration	joy	0.929	праздник	празднование, торжество, празднество
12	joy	joy	0.924	радость	счастье, удовольствие, веселье
21	gratitude	joy	0.914	благодарность	признательность, спасибо, благодарение
28	ecstasy	joy	0.906	экстаз	экстази, восторг, экзальтация

Рис. 4. Пример дополнения перевода при помощи сервиса «Reverso Context»

Примечание: составлено авторами по источнику [5].

Сервис «Reverso Context» представлен разнообразным набором синонимичных слов (табл. 5), позволяющим создать адекватно идентифицирующие кортежи слов для участников анкетирования и исключить ситуации выбора из двух одинаковых по смыслу слов.

Для удобства участнику анкетирования был представлен ограниченный кортеж слов с набором наиболее часто используемых значений. Пример предоставляемого респонденту 4-элементного кортежа с синонимичными терминами представлен ниже:

$$T_1 = \{e_1, e_2, e_3, e_4\}, \quad (3)$$

где e_1 = <освящение, посвящение, рукоположение, хиротонии>;

e_2 = <учредил, созданный, установленный, основанный>;

e_3 = <благословения, благословение, благословений, блага>;

e_4 = <гедонизм, жизнелюбие, гедонизму, гедонизмом>.

Так как кортежей для анкетирования одной группы эмоции требуется в два раза больше, чем слов в данной группе (по свойству встре-

чаемости каждого слова в восьми разных кортежах), при этом респонденту необходимо выбрать два разных элемента по свойству наихудшее/наилучшее, то всего вопросов респонденту будет в два раза больше, т. е. если в группе 1 483 слова, а количество 4-элементных кортежей – 2 966, то респонденту необходимо предоставить 5 932 заданий.

В процессе анкетирования респонденту предъявляют четыре строки (четыре кортежа) для определения двух строк (элементов), передающих наибольшую и наименьшую интенсивность заданной эмоции. Образец анкеты по определению эмоции «радость» представлен на рис. 5.

Перед анкетированием определены «золотые вопросы» – предварительно аннотированная случайная часть данных из всей выборки, которая служит механизмом предотвращения «шумов». Отнесение анкет к категории шум задается некоторым порогом, например, если точность «золотых вопросов» падает ниже 70 %, то данную анкету следует исключить из анализа.

Слова могут быть связаны с разной степенью эмоции. Большинство людей согласятся, что слово «смех» ассоциируется с большей степенью радости (или большей радостью), чем «улыбка».

Цель анкеты – определить степени радости, связанные со словами.

Мы предоставим вам анкету с заданиями, в которых располагаются четыре разных строки с синонимичными словами. Мы попросим вас:

- Выбрать строку (элемент), которая ассоциируется с **НАИБОЛЬШЕЙ** (**НАИЛУЧШЕЙ**) степенью радости;
- Выбрать строку (элемент), которая ассоциируется с **НАИМЕНЬШЕЙ** (**НАИХУДШЕЙ**) степенью радости.

Важные замечания:

- Синонимичные слова в строке призваны дополнить смысловое значение первого слова в строке, на основании которого должен производиться выбор;
- Выбор строки как «связанной с **НАИЛУЧШЕЙ** радостью» должен **ОТЛИЧАТЬСЯ** от выбора строки как «связанной с **НАИХУДШЕЙ** радостью» при рассмотрении одинакового набора элементов;
- Некоторые слова, такие как «смех» и «улыбка», явно выражают радость. Другие слова, например, «жёлтый цвет» и «победа» косвенно выражают и ассоциируют заданную эмоцию.

ПРИМЕР:

1. Выберите НАИЛУЧШИЙ элемент.

- победить, выиграть, выигрывать, побеждать
- ностальгия, тоска, ностальгировать, поностальгировать
- дружба, дружеские отношения, дружелюбие, дружеское отношение
- драгоценный камень, жемчужина, самоцвет, драгоценность

Ответ: победить, выиграть, выигрывать, побеждать

2. Выберите НАИХУДШИЙ элемент.

- победить, выиграть, выигрывать, побеждать
- ностальгия, тоска, ностальгировать, поностальгировать
- дружба, дружеские отношения, дружелюбие, дружеское отношение
- драгоценный камень, жемчужина, самоцвет, драгоценность

Ответ: ностальгия, тоска, ностальгировать, поностальгировать

Рис. 5. Вариант анкеты для группы эмоции «радость»

Примечание: составлено авторами по источнику [3].

ЗАКЛЮЧЕНИЕ

При анализе русскоязычных переводов словаря NRC-EIL версии 2022 года сформулировано предположение о расхождении коэффициентов интенсивности эмоций слов в связи с культурными различиями и разработан алгоритм составления аналогичного русскоязычного словаря.

Для получения адекватных результатов и исключения для респондента ситуации выбора из двух одинаковых по смыслу слов

и терминов предложен подход с использованием синонимичных слов.

Дальнейшее исследование значений интенсивности основных эмоций и расчет новых коэффициентов позволит определить количественную разницу при анализе тональности текста по словарям эмоций и значения различий коэффициентов интенсивности при использовании словаря NRC-EIL в русскоязычных текстах.

Список источников

1. Ардашев Р. Г. Аффект через призму общественных настроений // Проблема соотношения естественного и социального в обществе и человеке. 2021. № 12. С. 85–90.
2. Юрганов А. А. Сентимент-анализ как инструмент исследования текстов // Проблемы современной науки и образования. 2017. № 29. С. 39–41.
3. Mohammad S. M. Word Affect Intensities // Proceedings of the 11th International Conference on the Language Resources and Evaluation (LREC-2018),

References

1. Ardashev R. G. Affect through the Prism of Public Sentiment // Problema sootnosheniia estestvennogo i sotsialnogo v obshchestve i cheloveke. 2021. No. 12. P. 85–90. (In Russian).
2. Yurganov A. A. Sentiment-Analysis as a Tool of Investigation of Texts // Modern Problems of Science and Education. 2017. No. 29. P. 39–41. (In Russian).
3. Mohammad S. M. Word Affect Intensities // Proceedings of the 11th International Conference on the Language Resources and Evaluation (LREC-2018),

- May 7–12, 2018, Miyazaki, Japan. URL: <https://arxiv.org/pdf/1704.08798.pdf> (дата обращения: 17.11.2022).
4. Герстнер К. А. Диагностика эмоциональных состояний // Информационное общество и стратегические векторы развития региональных производственных систем : материалы Всерос. науч.-практ. конф., 28–29 ноября 2019 г., г. Ставрополь. Ставрополь : АГРУС, 2019. С. 165–168.
 5. Обработка данных с Pandas: разведывательный анализ данных. URL: <http://prog-life.ru:8085/blog/detail/08-03-2021-obrabotka-dannyih-s-pandas-razvedyivatelnyiy-analiz-dannyih> (дата обращения: 17.11.2022).
 6. The NRC Emotion Intensity Lexicon (NRC-EIL). URL: <http://saifmohammad.com/WebPages/AffectIntensity.htm> (дата обращения: 17.11.2022).
 7. Abbasi M. M., Beltiukov A. P. Analysis of Sentiment and Emotion from Text Written in Russian Language // Information Technologies for Intelligent Decision Making Support : Proceedings of the 5th All-Russian Conference, May 16–19, 2017, Ufa, Russia. 2017. P. 42–47.
 8. Kiritchenko S., Mohammad S. Best-Worst Scaling More Reliable than Rating Scales: A Case Study on Sentiment Intensity Annotation // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL-2017), July 30–August 4, 2017, Vancouver, Canada. 2017. URL: <https://arxiv.org/pdf/1712.01765.pdf> (дата обращения: 17.11.2022).
- May 7–12, 2018, Miyazaki, Japan. URL: <https://arxiv.org/pdf/1704.08798.pdf> (accessed: 17.11.2022). (In Russian).
4. Gerstner K. A. Diagnostika emotsionalnykh sostoianii // Informatsionnoe obshchestvo i strategicheskie vektory razvitiia regionalnykh proizvodstvennykh sistem : Proceedings of the All-Russian Research-to-Practice Conference, November 28–29, 2019, Stavropol. Stavropol : AGRUS, 2019. P. 165–168. (In Russian).
 5. Obrabotka dannykh s Pandas: razvedyivatelnyi analiz dannykh. URL: <http://prog-life.ru:8085/blog/detail/08-03-2021-obrabotka-dannyih-s-pandas-razvedyivatelnyiy-analiz-dannyih> (accessed: 17.11.2022). (In Russian).
 6. The NRC Emotion Intensity Lexicon (NRC-EIL). URL: <http://saifmohammad.com/WebPages/AffectIntensity.htm> (accessed: 17.11.2022). (In Russian).
 7. Abbasi M. M., Beltiukov A. P. Analysis of Sentiment and Emotion from Text Written in Russian Language // Information Technologies for Intelligent Decision Making Support : Proceedings of the 5th All-Russian Conference, May 16–19, 2017, Ufa, Russia. 2017. P. 42–47.
 8. Kiritchenko S., Mohammad S. Best-Worst Scaling More Reliable than Rating Scales: A Case Study on Sentiment Intensity Annotation // Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL-2017), July 30–August 4, 2017, Vancouver, Canada. 2017. URL: <https://arxiv.org/pdf/1712.01765.pdf> (accessed: 17.11.2022).

Информация об авторах

А. Р. Гончаров – магистрант.

С. А. Лысенкова – кандидат физико-математических наук, доцент.

Information about the authors

A. R. Goncharov – Master's Degree Student.

S. A. Lysenkova – Candidate of Sciences (Physics and Mathematics), Associate Professor.

Научная статья

УДК 004.67:519.246.8

doi: 10.34822/1999-7604-2022-4-37-48

АНАЛИЗ СТРУКТУРЫ ВРЕМЕННЫХ РЯДОВ КОЛИЧЕСТВА ДЕЛ В СУДЕ

**Василий Александрович Громов¹, Петр Павлович Лукьянченко²,
Юрий Николаевич Бесчастнов^{3✉}, Корней Кириллович Томащук⁴**

^{1, 2, 3, 4} *Национальный исследовательский университет «Высшая школа экономики»,
Москва, Россия*

¹ *stroller@rambler.ru, <https://orcid.org/0000-0001-5891-6597>*

² *lukianchenko.pierre@gmail.com, <https://orcid.org/0000-0002-1022-6041>*

³ *y.beschastnov@mail.ru ✉*

⁴ *korneytomashchuk@yandex.ru*

Аннотация. Проведен анализ временных рядов количества новых дел в административных судах РФ двумя методами группировки временных рядов с учетом хаотичности, случайности и регулярности их структуры. Первая модель основана на плоскости «энтропия – сложность», вторая – граф «атрибут – объект». Выведено четыре группы временных рядов: регулярные, регулярные-хаотические, строго хаотические и хаотические-стохастические, из которых хаотические-стохастические оказались в большинстве, что свойственно реальным системам. Для каждой группы предложен алгоритм прогнозирования в соответствии со структурой ряда, например, для хаотических рядов – алгоритмы нелинейной динамики, а для сильно стохастических рядов – модели, основанные на случайных процессах.

Ключевые слова: временные ряды, энтропия – сложность, анализ формальных понятий

Финансирование: публикация подготовлена за счет средств гранта на поддержку исследовательских центров в сфере искусственного интеллекта, в том числе в области «сильного» искусственного интеллекта, систем доверенного искусственного интеллекта и этических аспектов применения искусственного интеллекта, предоставленного АНО «Аналитический центр при Правительстве Российской Федерации» в соответствии с соглашением о предоставлении субсидии (идентификатор соглашения о предоставлении субсидии 000000D730321P5Q0002) и договором с ФГАОУ ВО «Национальный исследовательский университет «Высшая школа экономики» от 2 ноября 2021 г. № 70-2021-00139.

Для цитирования: Громов В. А., Лукьянченко П. П., Бесчастнов Ю. Н., Томащук К. К. Анализ структуры временных рядов количества дел в суде // Вестник кибернетики. 2022. № 4 (48). С. 37–48. DOI 10.34822/1999-7604-2022-4-37-48.

Original article

TIME SERIES STRUCTURE ANALYSIS OF THE NUMBER OF LAW CASES

**Vasily A. Gromov¹, Petr P. Lukyanchenko²,
Yury N. Beschastnov^{3✉}, Korney K. Tomashchuk⁴**

^{1, 2, 3, 4} *National Research University Higher School of Economics, Moscow, Russia*

¹ *stroller@rambler.ru, <https://orcid.org/0000-0001-5891-6597>*

² *lukianchenko.pierre@gmail.com, <https://orcid.org/0000-0002-1022-6041>*

³ *y.beschastnov@mail.ru ✉*

⁴ *korneytomashchuk@yandex.ru*

Abstract. The study analyzes the time series of the number of new cases in the administrative courts of the Russian Federation using two methods of time series grouping according to the chaotic, stochastic, and regular structure. The first model is based on the entropy–complexity plane, the second one is presented by the attribute–object graph. As a result, four groups of time series were derived: regular, regular-chaotic, purely chaotic, and chaotic-stochastic. Most of the series turned out to be chaotic-stochastic, which is common for

real systems. Each group of time series is assigned with a suitable prediction algorithm. For example, algorithms of nonlinear dynamics can be used for chaotic series, and models based on stochastic processes can be used for strongly stochastic series.

Keywords: time series, entropy–complexity, formal concept analysis

Funding: the article was supported by the grant for research centers in the field of AI, strong AI, trusted AI systems, and AI ethics provided by the Analytical Center for the Government of the Russian Federation in accordance with the agreement on the provision of subsidies (identifier of the agreement 000000D730321P5Q0002) and the agreement with the National Research University Higher School of Economics No. 70-2021-00139 of November 2, 2021.

For citation: Gromov V. A., Lukyanchenko P. P., Beschastnov Yu. N., Tomashchuk K. K. *Time Series Structure Analysis of the Number of Law Cases* // *Proceedings in Cybernetics*. 2022. No. 4 (48). P. 37–48. DOI 10.34822/1999-7604-2022-4-37-48.

ВВЕДЕНИЕ

Ежегодно арбитражные суды России рассматривают более 2 миллионов дел, их количество ежегодно увеличивается, при этом нагрузка судей по регионам сильно разнится, и даже в одном регионе она может существенно отличаться от месяца к месяцу. В некоторых регионах на судью приходится не более 70 дел в месяц, в других – более 250, а из-за неравномерности возникновения правовых нарушений возникает ситуация, когда один арбитражный суд имеет в распоряжении судей больше чем нужно, в то время как в других регионах судей недостаточно. Таким образом, прогнозирование количества дел в арбитражных судах России чрезвычайно важная задача, математически формулируемая как задача прогнозирования сложных временных рядов.

Мы имеем сильно отличающиеся друг от друга по структуре 3 149 рядов как регулярных, так и хаотических с сильной случайной компонентой (рис. 1). Учитывая, что построение единой для всех рядов прогнозной модели получения адекватного прогноза невозможно, важно определить их сложность и тип, например, регулярные ряды могут быть предсказаны с помощью авторегрессионных моделей, а для хаотических рядов данные методы приводят к экспоненциальному росту ошибки [1].

Проведено исследование сложности временных рядов, описывающих количество новых дел в арбитражных судах. Для разделения рядов на регулярные, хаотические и стохастические использован метод вычисления характеристик энтропии и сложности

для множества ординальных структур рассматриваемого временного ряда и анализ положения соответствующей точки на плоскости «энтропия – сложность», а также метод анализа формальных понятий [2], основанный на построении графа «атрибут – объект», который позволяет не только различить ряды между собой, но также представить структуру ряда. С помощью этих методов удалось разбить ряды на группы по их сложности и выраженности случайной компоненты, что крайне важно при прогнозировании и выборе методов прогнозирования для того или иного ряда.

При проведении исследования учитывались новые способы и методы вычисления, например, метод вычисления наибольшего показателя Ляпунова из временного ряда [3]. Метод опирается на определения старшего показателя Ляпунова, а алгоритм устойчив к изменениям следующих параметров: размер выборки, размерность вложения данных, окно реконструкции и уровень шума. Кроме того, можно использовать алгоритм для одновременного вычисления корреляционной размерности, т. е. одна последовательность вычислений представляет оценку как уровня хаоса, так и сложности системы.

Популярным новым инструментом для обнаружения хаоса во временных рядах является тест 0–1 (G. A. Gottwald, I. Melbourne) [4]. В отличие от методов касательного пространства, знание полного пространства состояний не требуется, так как реконструкция пространства состояний является необязательной процедурой, поэтому метод можно использовать для данных, у которых не все переменные со-

стояния наблюдаемы. Тест 0–1 был представлен в двух вариантах: тест регрессии и тест корреляции, а дальнейшее исследование [5] показало, что корреляционный тест хорошо работает для свободных от шума временных рядов, однако в случае присутствия в данных шума чувствительность оценок к шуму снижает модифицированная версия исходного ре-

грессионного теста. Для решения проблемы шума может быть использован метод корреляции, если уровневые значения даны для временного ряда, о котором известно, что он периодический, однако требуемый уровень, как утверждают ученые в [6], не всегда можно получить в реальных системах.

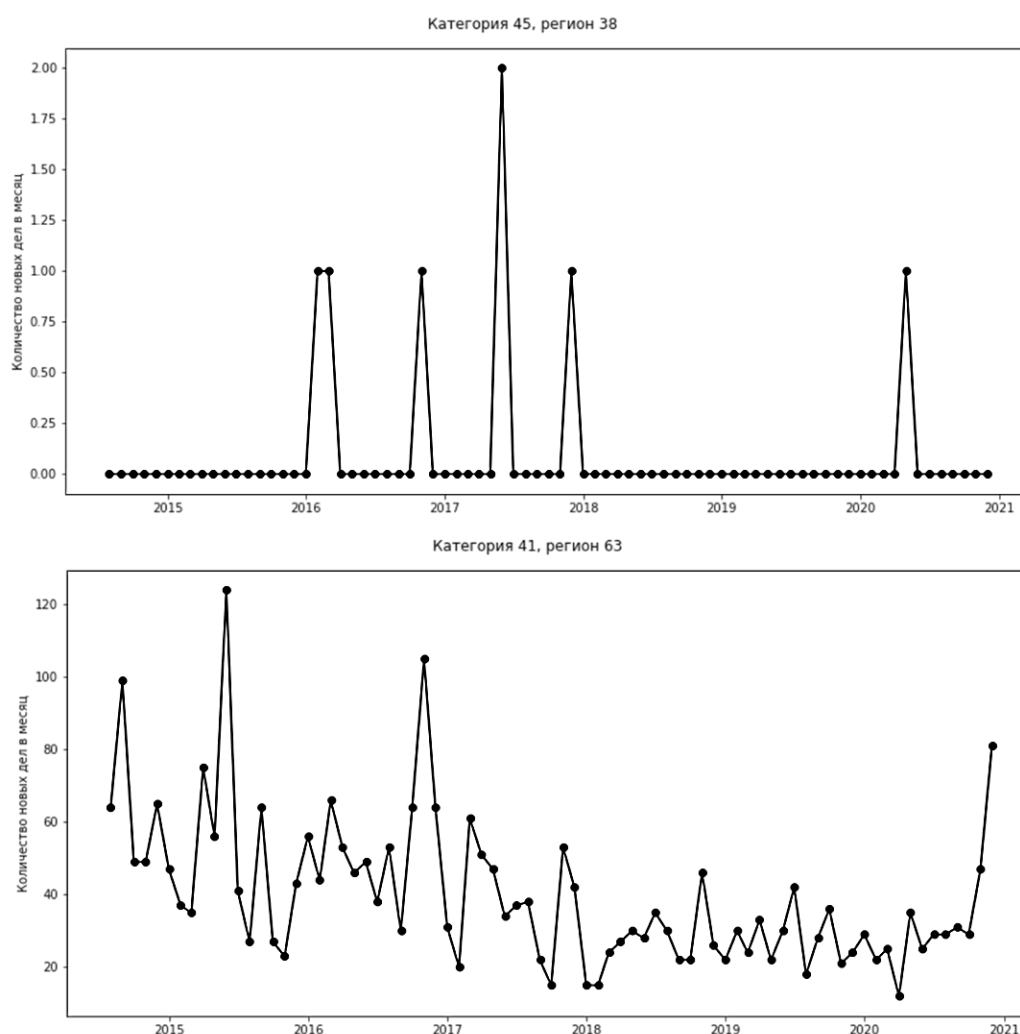


Рис. 1. Пример двух рядов, показывающих количество новых дел
Примечание: составлено авторами на основании данных, полученных в исследовании.

В [7] представлен новый метод обнаружения хаоса, в котором используются инструменты топологического анализа данных. Двумерные оценки плотности случайно спроецированных временных рядов на плоскости « $p-q$ » используются для создания бело-серой картинки, по которой возможно оценить является ли основной временной ряд хаотичным. Алгоритм распознавания использован на ря-

дах Лоренца и Росслера, и этот алгоритм тестирования хаоса сравнивается с тестом корреляции для незагрязненных временных рядов и его способность различать периодическую и хаотическую динамику даже при высоких уровнях шума. Минус этих алгоритмов в том, что они не сходятся для траекторий с частично предсказуемым хаосом (траекторий с нулевым показателем взаимного масштаби-

рования и ненулевой взаимной корреляцией). Авторы используют стабильность устойчивых гомологий в присутствии шума, чтобы представить метод для определения точек сдвига между хаотической и регулярной динамикой в умеренно зашумленных временных рядах.

Иной подход для определения типа временных рядов основывается на понятии ординальной структуры и представлении временного ряда точкой на плоскости «энтропия – сложность» и позволяет различать сложные детерминированные (хаотические) временные ряды, простые детерминированные (хаотические) временные ряды и стохастические временные ряды [8].

В [9] представлен метод, который использовался для выявления хаотических, случайных и регулярных структур во временных медицинских рядах. Алгоритм построен на основе выявления сообществ в графе и анализа формальных понятий, а количеством определенных структур графа описывалось поведение ряда.

МАТЕРИАЛЫ И МЕТОДЫ

Постановка задачи. Дано множество рядов – необходимо различать хаотичную, стохастическую и регулярную структуры рядов. Пусть $x_t = f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, t)$ – значение ряда в момент времени t , где ε_s – стохастический процесс, f – функция перехода. Тогда:

1. Условие регулярности ряда:

$$\text{одновременно } \{\varepsilon_s\}_{s=0}^{t-1} = 0 \\ \text{и } \forall t \parallel \lambda(D_x f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, t)) \parallel_\infty < 1.$$

2. Условия стохастичности ряда:

ε_s – стохастический процесс.

3. Условие хаотичности ряда:

$$\text{одновременно } \{\varepsilon_s\}_{s=0}^{t-1} = 0 \\ \text{и } \forall t \parallel \lambda(D_x f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, t)) \parallel_\infty > 1,$$

где $u_{t+1} = D_x f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, t) = \delta^{-1}(f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, x_0 + \delta u_t, t) - f(\{\varepsilon_s\}_{s=0}^t, \{x_s\}_{s=0}^{t-1}, x_0, t))$ при

$\delta \rightarrow 0$ [10], $\parallel \lambda(A) \parallel_\infty$ – максимальное собственное значение матрицы A .

Для решения задачи выявления различий в структуре временных рядов были использованы энтропийные методы и алгоритмы, основанные на графах. Энтропийный метод отображает рассматриваемый ряд в точку плоскости «энтропия – сложность».

Метод «энтропия – сложность». О. А. Rosso и соавт. [8] был предложен метод, позволяющий отличить хаотические, регулярные и случайные ряды. Алгоритм опирается на вероятностное распределение ординальных структур, построенных по временному ряду $\{y_i\}$. Для данного z -вектора длины n $d_i = (y_{i-(n-1)}, y_{i-(n-2)}, \dots, y_{i-1}, y_i)$, где y_k это k -е наблюдение ряда, строится бинарный вектор размерности $n - 1$ такой, что в i компоненте вектора ставится 1, если i компонента действительного вектора $\leq$ компоненты этого же вектора на $i + 1$ месте.

Указанный вектор носит название ординальной структуры, соответствующей данному действительному вектору. Алгоритм для построения вероятностного распределения порядковых номеров для временного ряда состоит из следующих этапов:

1. Для каждого z -вектора временного ряда вычисляем соответствующую ему ординальную структуру d_i .

2. Строим эмпирическое распределение вероятностей появления ординальных структур в ряде P_i .

3. Для распределения вычисляется энтропия (1) и сложность (2):

$$H[P] = \frac{S[P]}{S_{\max}}, \quad (1)$$

$$C[P] = Q[P, P_e] \cdot H[P], \quad (2)$$

где $S[P] = -\sum_{i=1}^n P_i \cdot \ln(P_i)$, $S_{\max} = \ln(n!) = S[P_e]$, P_e – равномерное распределение; $Q[P, P_e] = Q_0 \cdot \left(S\left[\frac{P + P_e}{2}\right] - S\left[\frac{P}{2}\right] - S\left[\frac{P_e}{2}\right] \right)$, Q – дивергенция Дженсен – Шеннона между данным и равномерным распределениями, а Q_0 – коэффициент нормализации.

Описанная выше процедура отображает временной ряд в точку на плоскости «энтропия – сложность». Положение точки относительно нижней и верхней теоретических границ указывает на тип рассматриваемого ряда [9].

Простые детерминированные процессы дадут точку, расположенную в левом нижнем углу фигуры, ограниченной верхней и нижней теоретическими границами (рис. 2).

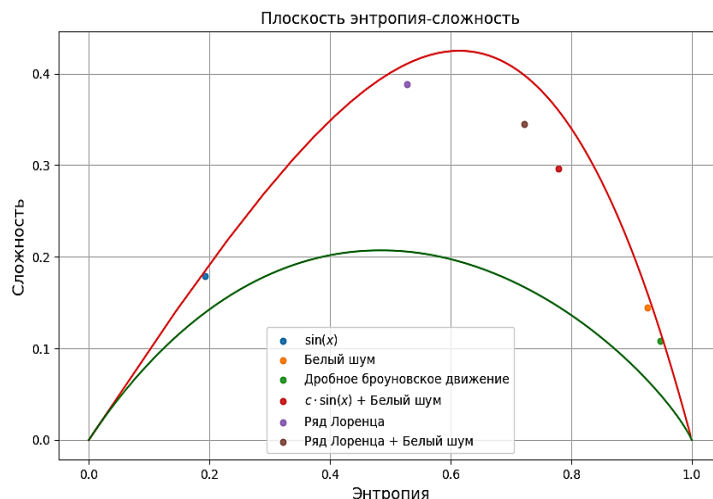


Рис. 2. Стохастические (зеленый, оранжевый), хаотический (фиолетовый), регулярный (синий), регулярный с шумом (красный), хаотический с шумом (коричневый)

Примечание: составлено авторами на основании данных, полученных в исследовании.

С помощью алгоритмов кластеризации можно осуществить кластеризацию точек на плоскости «энтропия – сложность» и выявить группы временных рядов различных типов.

В рамках анализа формальных понятий был предложен другой подход к различению регулярных рядов от хаотических – построение графа «атрибут – объект» [9]. В методе предполагается построение выборки по нормализо-

ванному временному ряду, бинаризация атрибутов, характеризующих элементы выборки, построение графа «атрибут – объект», анализ характеристик указанного графа с целью различения графов, соответствующих регулярным и хаотическим временным рядам. Пример характерного графа «атрибут – объект» представлен на рис. 3.

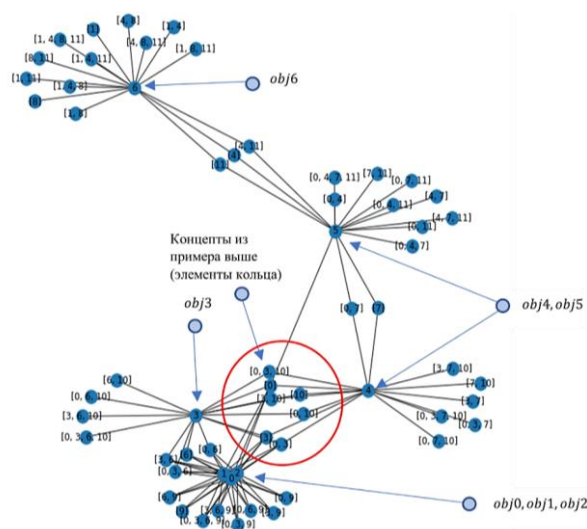


Рис. 3. Граф «объект – атрибут»

Примечание: составлено авторами на основании данных, полученных в исследовании.

Допустим, есть временной ряд $\{x_t\}_{t=0}^T$. Сначала, приведем нормированный ряд в удобный для кластеризации вид. Запишем

ряд как матрицу $\begin{bmatrix} x_0 & \cdots & x_L \\ \vdots & \ddots & \vdots \\ x_{T-L} & \cdots & x_T \end{bmatrix}$, где L – пара-

метр размерности матрицы. Для того, чтобы воспользоваться методами формальных понятий бинаризуем эту матрицу. Для бинаризации установим количество разбиения фазового пространства n и найдем интервалы разбиения. Определим $k_{\min} = \min_{0 \leq t \leq T} x_t$, $k_{\max} = \max_{0 \leq t \leq T} x_t$, тогда I_i (3) – интервал итого разбиения:

$$I_i = \left[k_{\min} + (i-1) \frac{k_{\max} - k_{\min}}{n}, k_{\min} + i \frac{k_{\max} - k_{\min}}{n} \right]. \quad (3)$$

Бинаризация строчки матрицы является булевой операцией. Для каждой компоненты записывается булевой вектор ее попадания во все данные интервалы. Для удобства запишем такую операцию как $F(x) = [\text{bool}(x \in I_1), \dots, \text{bool}(x \in I_n)]$. Тогда строчка булевой матрицы имеет вид $[F(x_j), F(x_{j+1}), \dots, F(x_{j+L})]$. Пример для $n = 3$, $L = 2$, $k_{\min} = -0,0048$, $k_{\max} = 0,004$: $[-0,00024336, 0,00021588, 0,00096167] \rightarrow [0, 1, 0, 0, 1, 0, 0, 1, 0]$. Примером такой матрицы служит рис. 4.

	atr0	atr1	atr2	atr3	atr4	atr5	atr6	atr7	atr8	atr9	atr10	atr11
obj0	1	0	0	1	0	0	1	0	0	1	0	0
obj1	1	0	0	1	0	0	1	0	0	1	0	0
obj2	1	0	0	1	0	0	1	0	0	1	0	0
obj3	1	0	0	1	0	0	1	0	0	0	1	0
obj4	1	0	0	1	0	0	0	1	0	0	1	0
obj5	1	0	0	0	1	0	0	1	0	0	0	1
obj6	0	1	0	0	1	0	0	0	1	0	0	1

Рис. 4. Бинарная матрица

Примечание: составлено авторами на основании данных, полученных в исследовании.

После реконструкции матрицы построим граф «атрибут – объект». Для каждой пары строчек бинарной матрицы найдем все множества кольца ее совпадений значения 1.

Например, строчки *obj3* и *obj4*: $\{atr0, atr3, atr10\}$, $\{atr0, atr3\}$, $\{atr10, atr3\}$, $\{atr0, atr10\}$, $\{atr0\}$, $\{atr3\}$, $\{atr10\}$ (рис. 5).

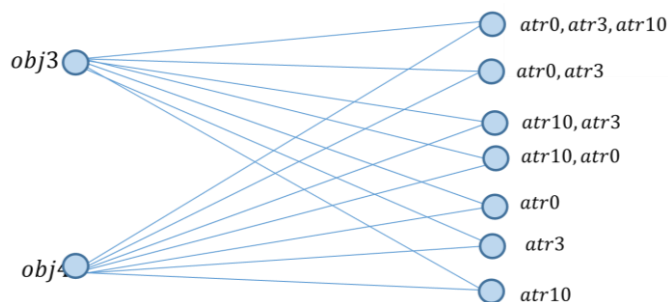


Рис. 5. Граф «объект – атрибут»

Примечание: составлено авторами на основании данных, полученных в исследовании.

Таким образом, получаем полный алгоритм построения графа «атрибут – объект»:

x_k – вектор, который состоит из бинарных компонент;

L – размерность векторов;

$E = \{\emptyset\}$ – множество ребер;

$V = \{x_1, \dots, x_n\}$, где $1, \dots, n$ индексы векторов, которые будут кластеризованы

```

for  $v_i$  in  $V$ :
for  $v_j$  in  $V$ :
if  $i \neq j$ :
 $A = \{\}$  – порождающее множество кольца
for  $k$  from 1 to  $L$ :
if  $v_i[k] = 1$  and  $v_j[k] = 1$ :
 $A \leftarrow k$ 
for  $m$  in  $\text{Ring}(A)$ :
 $m = \text{string}(\text{sorted}(m))$ 
 $E \leftarrow (v_i, m)$  добавляем ребро
Получаем  $G(E)$  граф «атрибут – объект»

```

Для нахождения классов используем алгоритм распространения меток (рис. 6). Однако проблема в том, что подграфы в графе, которые являются двудольными или почти двудольными по структуре, приводят к колебаниям меток. Это особенно верно в случаях, когда подграфы имеют форму звездного графа. Следовательно, нужно использовать асинхронное обновление, когда некоторые соседи уже были обновлены в текущей итерации, а некоторые еще не обновлены в текущей итерации. Тогда алгоритм кластеризации графа будет состоять из следующих шагов:

Пусть:

G – граф;
 $V = [x_1, \dots, x_n]$ – массив вершин графа;
 $D_V = [\varepsilon(x_1), \dots, \varepsilon(x_n)]$ – массив меток вершин V (на 0 итерации, $\varepsilon(x_i) = i$);
 $w(k, S)$ – количество элементов k в массиве S ;
 $N(x)$ – массив соседей вершины x .

Тогда:

```

while  $\forall x \in V \forall u \in N(x) \max_{\varepsilon} \{w(\varepsilon(y), D_{N(x)}): \forall y \in N(x)\} < w(\varepsilon(u), D_{N(x)})$ :

```

$\bar{V} = [x_{\sigma(1)}, x_{\sigma(2)}, \dots, x_{\sigma(N)}]$ – массив перестановок вершин в случайном порядке

for $u \in \bar{V}$:

$\varepsilon^* = \text{argmax}_{\varepsilon} \{w(\varepsilon(y), D_{N(u)}): \forall y \in N(u)\}$

if $\nexists! \varepsilon^*$: выбираем наугад ε^* из множества $\{\varepsilon^*\}$

$D\{u\} = \varepsilon^*$

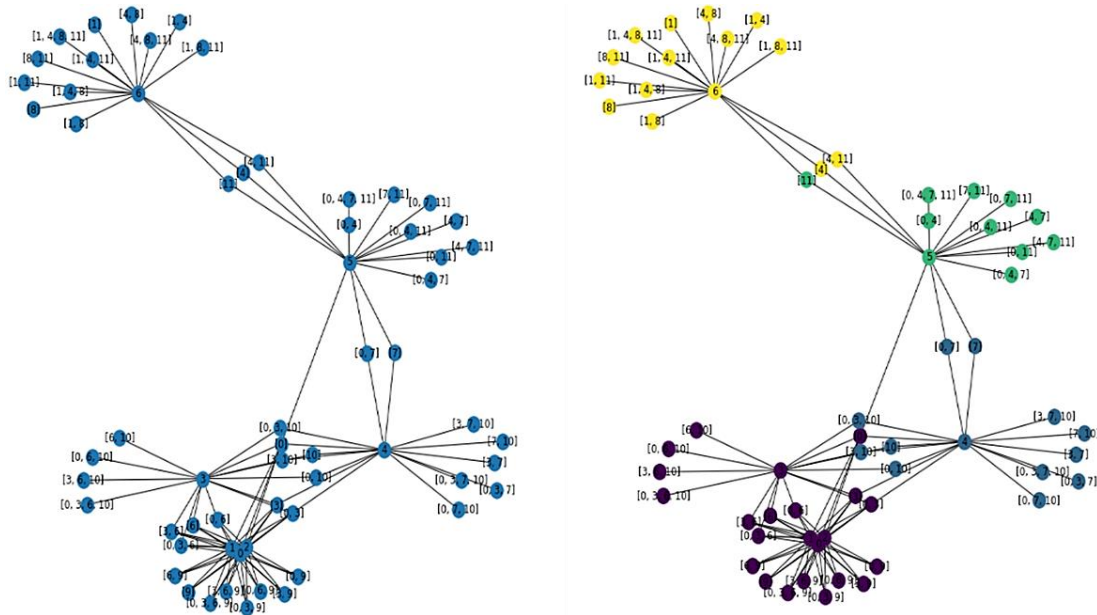


Рис. 6. Кластеризация графа «атрибут – объект»

Видно, что некоторые объекты группируются в один кластер (фиолетовая метка), а другие образуют отличные друг от друга группы (желтая, зеленая, синяя метки)

Примечание: составлено авторами на основании данных, полученных в исследовании.

Характеристики хаотичности, случайности и регулярности в ряде. Определим шумовые кластеры как кластеры с мощностью 1. Если доля вне кластерного шума определяет большинство объектных вершин графа и присутствует много малых кластеров, тогда ряд

можно считать стохастическим. В случае, если пропорция количества кластеров к количеству объектных вершин приближается к 0 и доля шума почти 0, то ряд регулярный. Если количество кластеров большое, а доля шума маленькая (приближается к 0), то ряд

хаотический. Малое количество кластеров и большое количество шума дает нам хаотический-стохастический ряд. Примеры класте-

ризации графа «атрибут – объект» хаотического временного ряда и регулярного изображены на рис. 7.



Рис. 7. Примеры графов хаотического и регулярного ряда

Примечание: составлено авторами на основании данных, полученных в исследовании.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В численном эксперименте были использованы временные ряды, характеризующие количество новых судебных дел по месяцам в разных регионах России и по разным судебным категориям с 2014-08-01 по 2020-12-01 [11]. Общее число рядов – 3 149. Перед применением вышеописанных алгоритмов все ряды были трансформированы в стационарные; проверка на стационарность осуществлялась с помощью расширенного теста Дики – Фул-

лера [12]. Далее каждый ряд был нормализован для удобства подбора параметров в методах.

Так как многие ряды имеют много константных участков, такие ряды отнесены в отдельную «константную» группу: критерием отнесения является условие $\frac{|[x_i] - \text{mod } a\{x_i\} = 0|}{|\{x_i\}|} > \varepsilon$, где ε – мало. Пример

такого ряда показан на рис. 8. Для предсказания таких рядов использовали среднее по ряду.

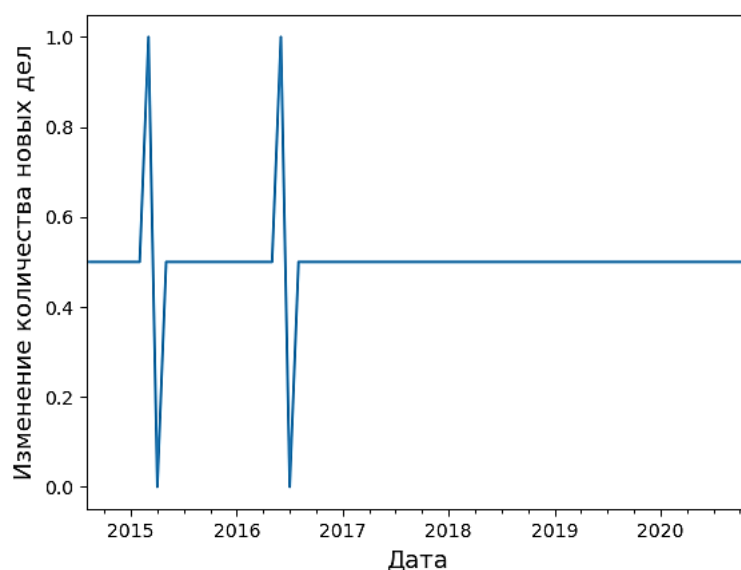


Рис. 8. Пример околоконстантного ряда

Примечание: составлено авторами на основании данных, полученных в исследовании.

Остальные ряды были сгруппированы путем (1) вычисления значений энтропии и сложности с последующей кластеризацией и (2) алгоритма, основанного на атрибут-объектном графе.

Фиолетовые точки на плоскости занимают 74,4 % (рис. 9а). «Неконстантных» рядов $\left(\frac{|\{x_i\} - \text{mod } a\{x_i\} = 0|}{|\{x_i\}|} \leq \varepsilon, \text{ для } \varepsilon = 0,33 \right)$ было обнаружено порядка 76,3 % (синие точки на рис. 9б). Кластеризация (методом K-means [13])

разбивает плоскость «энтропия – сложность» на четыре основных области (фиолетовая, желтая, зеленая, синяя), где фиолетовая область – стохастические-хаотические ряды 74,4 %, желтая – хаотические 14,5 %, зеленая – хаотические-регулярные 8,1 %, и синие – регулярные 3 % (рис. 9а). Из первого и второго графика (рис. 9) можно предположить, что фиолетовый кластер состоит из неконстантных рядов, которые имеют хаотическую природу с ярко выраженным случайным компонентом.

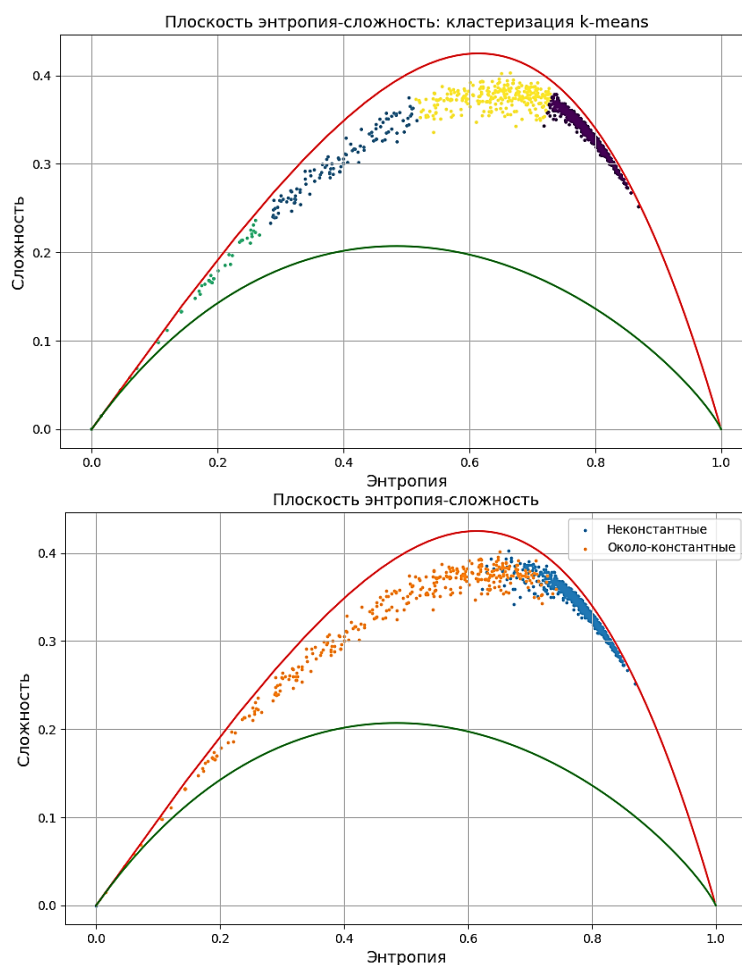


Рис. 9. «Энтропия – сложность»: 9а (верхний) – кластеризация точек на плоскости «энтропия – сложность» с помощью k-means; 9б (нижний) – отображение неконстантных и околоконстантных рядов на плоскости «энтропия – сложность»

Примечание: составлено авторами на основании данных, полученных в исследовании.

Применение метода, основанного на атрибут-объектном графе, дало другое разбиение рассматриваемого множества рядов (рис. 10). Здесь были выделены шумовые и не шумовые кластеры. Распределение процента шумовых

кластеров показало, что у 78 % рядов более 15 % частей последовательности непохожи друг на друга (рис. 10а). К тому же 66 % рядов имеют отношение менее 0,1 количества кластеров на все объекты (рис. 10б).

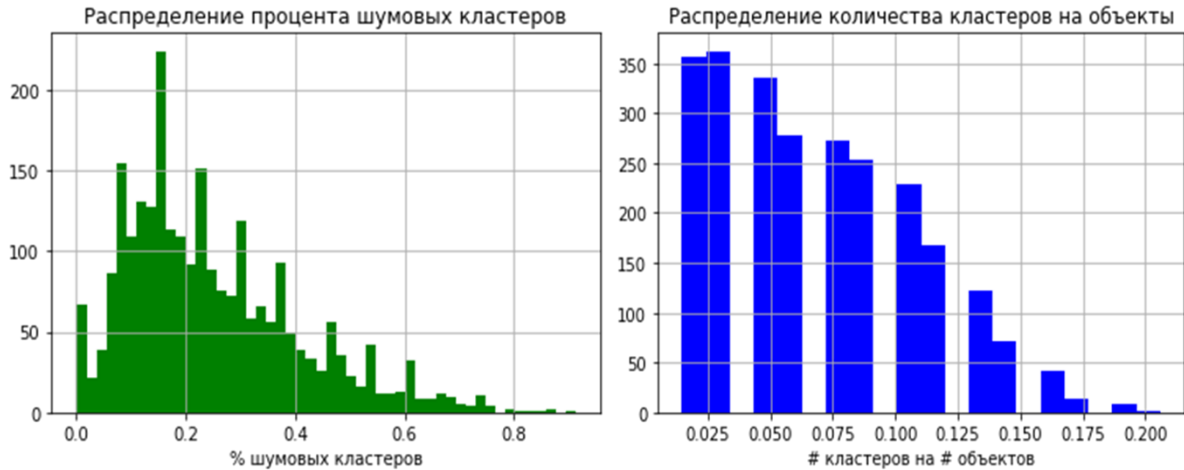


Рис. 10. Распределение метрик: 10а (левый) – процент шумовых кластеров; 10б (правый) – количество кластеров на объекты

Примечание: составлено авторами на основании данных, полученных в исследовании.

Если рассмотреть ряд как прообраз этих двух подмножеств метрик, такой ряд имеет стохастическую смешанную с хаотической структуру, то есть существуют похожие участки рядов, но есть и части последовательности, непохожие друг на друга. Пересечение

прообразов этих двух подмножеств метрик составляет 73,3 % (рис. 11). Такие ряды относятся к хаотическим-стохастическим рядам. Регулярные ряды составляют 5 % от всех рядов, а строго хаотические, или хаотические-регулярные – 21,7 % (рис. 11).

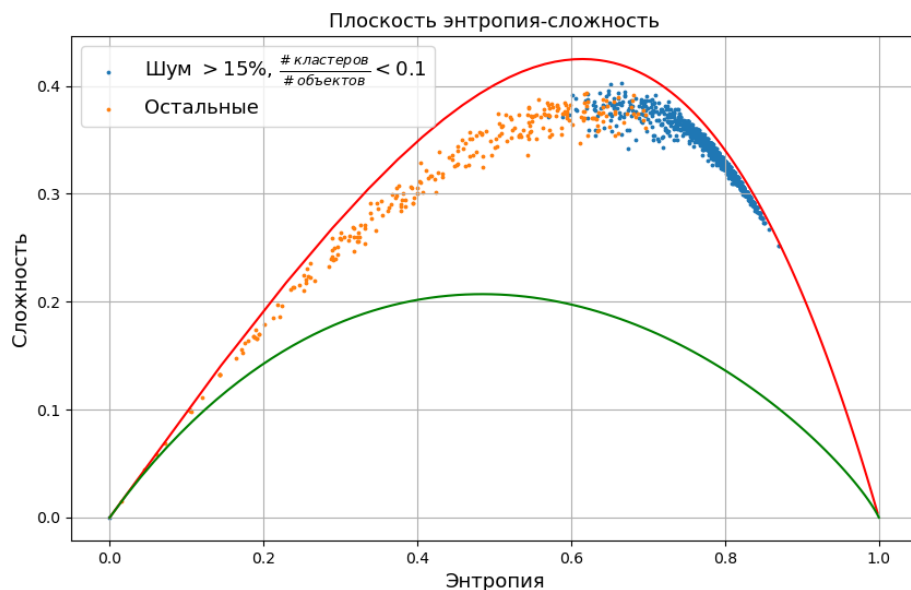


Рис. 11. Распределение около-константных рядов на плоскости «энтропия – сложность»

Примечание: составлено авторами на основании данных, полученных в исследовании.

Приведены примеры рядов, принадлежащих разным структурным группам (рис. 12). На левом верхнем графике доля вне кластерного шума – 1,4 %, а отношение количества кластеров к объектам графа – 0,058, что говорит о регулярности ряда. Правый верхний ряд

имеет 17 % шума и 0,014 пропорцию кластеров к объектам графа, такой ряд является хаотическим-стохастическим. На нижнем – 76 % и 0,073, можно предположить, что ряд чисто стохастический.

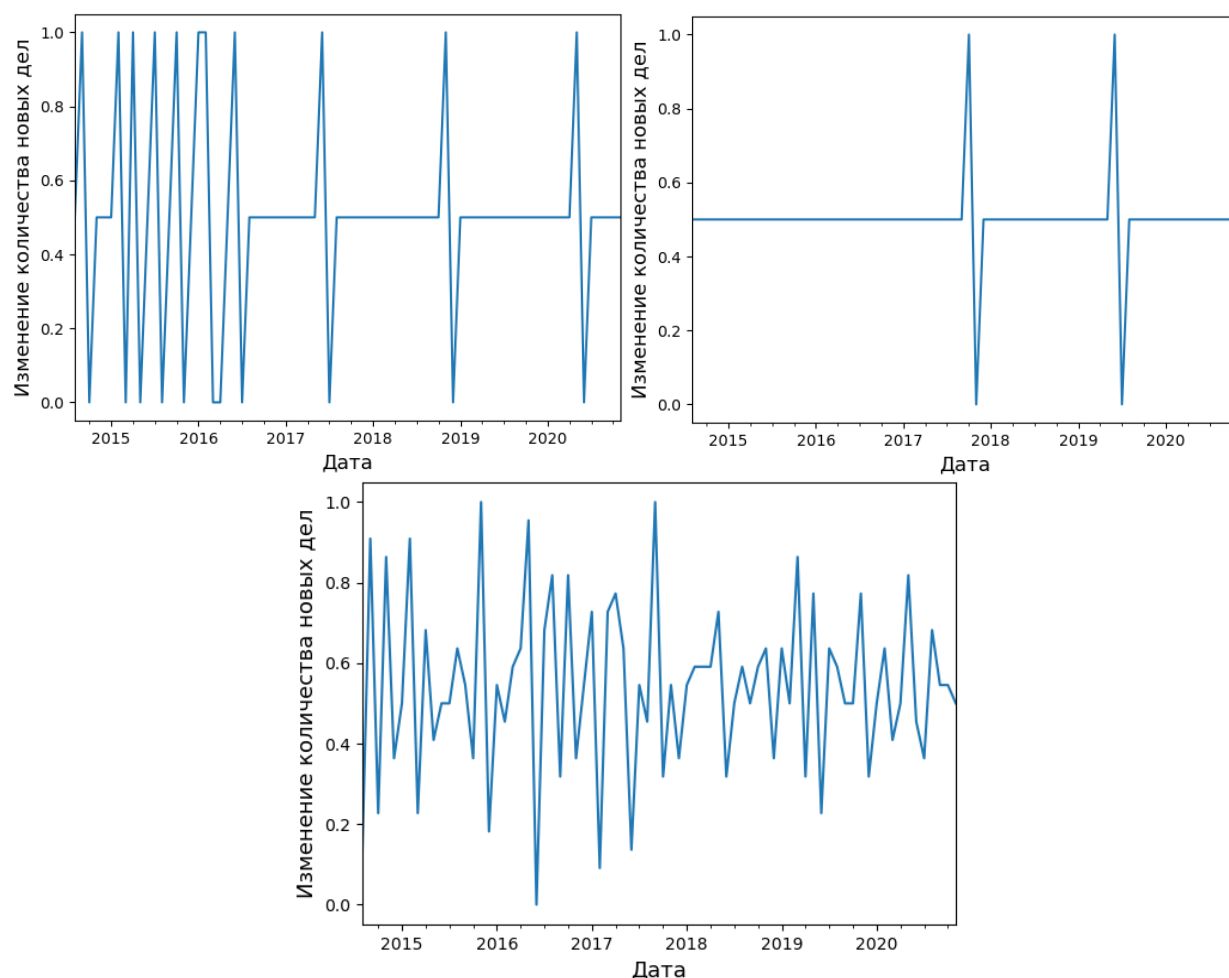


Рис. 12. Примеры регулярного, хаотического и хаотического-стохастического рядов
Примечание: составлено авторами на основании данных, полученных в исследовании.

ЗАКЛЮЧЕНИЕ

Проведено исследование характеристик временных рядов, описывающих количество новых дел в суде в разных регионах России и в разных категориях.

Предложено два способа группировки временных рядов: плоскость «энтропия – сложность», граф «атрибут – объект». Установлено, что используя первый метод, процент хаотических-стохастических рядов в выборке составляет 74,4 %, хаотических и хаотических-регулярных – 22,6 %, регулярных – 3 %. Второй метод классифицировал 73,3 % рядов как хаотические-стохастические ряды, 21,7 % –

строгие хаотические, или хаотические-регулярные, 5 % – регулярные.

В дальнейшем этот эксперимент поможет решить задачу прогнозирования этих рядов (число дел) на много шагов вперед, так как для разных структур ряда должны быть использованы разные методы прогнозирования. Например, ряды с сильной стохастической разностью разумнее прогнозировать моделями случайных процессов, хаотические ряды – моделями нелинейной динамики, а регулярные – в зависимости от константности ряда (средним или OLS моделями).

Список источников

1. Gromov V. A. Chaotic Time Series Prediction: Run for the Horizon // Proceedings of the 5th International Conference, TMPA 2019, Tbilisi, Georgia, November 7–9, 2019. 2021. P. 29–43.

References

1. Gromov V. A. Chaotic Time Series Prediction: Run for the Horizon // Proceedings of the 5th International Conference, TMPA 2019, Tbilisi, Georgia, November 7–9, 2019. 2021. P. 29–43.

2. Ferré S., Huchard M., Kayoue M., Kuznetsov S. O., Napoli A. Formal Concept Analysis: From Knowledge Discovery to Knowledge Processing // A Guided Tour of Artificial Intelligence Research / Eds. Marquis P., Papini O., Prade H. 2020. P. 411–445.
3. Mehdizadeh S. A Robust Method to Estimate the Largest Lyapunov Exponent of Noisy Signals: A Revision to the Rosenstein's Algorithm // J Biomech. 2019. Vol. 85. P. 84–91.
4. Gottwald G. A., Melbourne I. A New Test for Chaos in Deterministic Systems // Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences. 2004. Vol. 460, No. 2042. P. 603–611.
5. Gottwald G. A., Melbourne I. On the Implementation of the 0–1 Test for Chaos // SIAM Journal on Applied Dynamical Systems. 2009. Vol. 8, Is. 1. P. 129–145.
6. Gottwald G. A., Melbourne I. The 0-1 Test for Chaos: A Review // Chaos Detection and Predictability / Eds. Skokos C., Gottwald G., Laskar J. 2016. P. 221–247.
7. Tempelman J. R., Khasawneh F. A. A Look into Chaos Detection through Topological Data Analysis // Physica D: Nonlinear Phenomena. 2020. Vol. 406. P. 132446.
8. Rosso O. A., Carpi L. C., Saco P. M. et al. Causality and the Entropy–Complexity Plane: Robustness and Missing Ordinal Patterns // Physica A: Statistical Mechanics and its Applications. 2012. Vol. 391, Is. 1–2. P. 42–55.
9. Громов В. А., Мазайшвили К. В., Заикин П. В. и др. Различение хаотических и регулярных временных рядов для идентификации состояния артериовенозной фистулы // Вестник кибернетики. 2022. № 1 (45). С. 72–82.
10. Малинецкий Г. Г., Потапов А. Б., Подлазов А. В. Нелинейная динамика: подходы, результаты, надежды // Синергетика: от прошлого к будущему. 2016. Т. 9, № 28. С. 30–121.
11. Электронное правосудие. URL: <https://kad.arbitr.ru/> (дата обращения: 01.11.2022).
12. Mushtaq R. Augmented Dickey Fuller Test // SSRN Electronic Journal. 2011. 19 p.
13. Liu Y., Ma S., Du X. An Improved K-Means Algorithm Based on a New Cluster Center Selection Method // IEEE Access. URL: https://www.researchgate.net/publication/347610712_An_improved_k-means_algorithm_based_on_a_new_cluster_center_selection_method (дата обращения: 01.11.2022).
2. Ferré S., Huchard M., Kayoue M., Kuznetsov S. O., Napoli A. Formal Concept Analysis: From Knowledge Discovery to Knowledge Processing // A Guided Tour of Artificial Intelligence Research / Eds. Marquis P., Papini O., Prade H. 2020. P. 411–445.
3. Mehdizadeh S. A Robust Method to Estimate the Largest Lyapunov Exponent of Noisy Signals: A Revision to the Rosenstein's Algorithm // J Biomech. 2019. Vol. 85. P. 84–91.
4. Gottwald G. A., Melbourne I. A New Test for Chaos in Deterministic Systems // Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences. 2004. Vol. 460, No. 2042. P. 603–611.
5. Gottwald G. A., Melbourne I. On the Implementation of the 0–1 Test for Chaos // SIAM Journal on Applied Dynamical Systems. 2009. Vol. 8, Is. 1. P. 129–145.
6. Gottwald G. A., Melbourne I. The 0-1 Test for Chaos: A Review // Chaos Detection and Predictability / Eds. Skokos C., Gottwald G., Laskar J. 2016. P. 221–247.
7. Tempelman J. R., Khasawneh F. A. A Look into Chaos Detection through Topological Data Analysis // Physica D: Nonlinear Phenomena. 2020. Vol. 406. P. 132446.
8. Rosso O. A., Carpi L. C., Saco P. M. et al. Causality and the Entropy–Complexity Plane: Robustness and Missing Ordinal Patterns // Physica A: Statistical Mechanics and its Applications. 2012. Vol. 391, Is. 1–2. P. 42–55.
9. Gromov V. A., Mazayshvili K. V., Zaikin P. V. et al. Differentiating Chaotic and Regular Time Series for Identification of Arteriovenous Fistula State // Proceedings in Cybernetics. 2022. No. 1 (45). P. 72–82. (In Russian).
10. Malinetsky G. G., Potapov A. B., Podlazov A. V. Nelineinaia dinamika: podkhody, rezultaty, nadezhdy // Sinergetika: ot proshlogo k budushchemu. 2016. Vol. 9, No. 28. P. 30–121. (In Russian).
11. Elektronnoe pravosudie. URL: <https://kad.arbitr.ru/> (accessed: 01.11.2022). (In Russian).
12. Mushtaq R. Augmented Dickey Fuller Test // SSRN Electronic Journal. 2011. 19 p.
13. Liu Y., Ma S., Du X. An Improved K-Means Algorithm Based on a New Cluster Center Selection Method // IEEE Access. URL: https://www.researchgate.net/publication/347610712_An_improved_k-means_algorithm_based_on_a_new_cluster_center_selection_method (accessed: 01.11.2022).

Информация об авторах

В. А. Громов – доктор физико-математических наук, профессор.
П. П. Лукьянченко – старший преподаватель.
Ю. Н. Бесчастнов – стажер-исследователь.
К. К. Томашук – стажер-исследователь.

Information about the authors

V. A. Gromov – Doctor of Sciences (Physics and Mathematics), Professor.
P. P. Lukyanchenko – Senior Lecturer.
Yu. N. Beschastnov – Research Assistant.
K. K. Tomashchuk – Research Assistant.

Научная статья
УДК 517.983:519.6:544.4
doi: 10.34822/1999-7604-2022-4-49-58

МЕТОД РЕГУЛЯРИЗАЦИИ ТИХОНОВА ДЛЯ РЕШЕНИЯ ОБРАТНОЙ ЗАДАЧИ В МАТЕМАТИЧЕСКОЙ МОДЕЛИ КИНЕТИКИ ПРОЦЕССА НЕФТЕПЕРЕРАБОТКИ

Ван Хуен Лэ^{1✉}, Андрей Николаевич Фирсов²

^{1,2} Санкт-Петербургский политехнический университет Петра Великого,
Санкт-Петербург, Россия

¹ huyenlevan120193@gmail.com ✉, <https://orcid.org/0000-0002-8645-821X>

² anfirs@yandex.ru, <https://orcid.org/0000-0003-4483-3381>

Аннотация. Представлено решение обратной задачи приближенного определения констант скорости реакций по заданным концентрациям исходного вещества и продуктов в рамках математической модели кинетики процесса нефтепереработки с использованием методов конечных разностей, интерполировании данных (кубического сплайна) и метода регуляризации Тихонова. Найдены приближительные константы скорости реакций, которые являются единственными и непрерывно зависят от исходных данных, а также необходимы для прогнозирования концентрации исходного вещества и продуктов в любой момент времени и определения показателей в реакциях процесса нефтепереработки – энергии активации, температуры и других внешних условий.

Ключевые слова: обратная задача, некорректно поставленная задача, математическая модель, химическая кинетика, процесс нефтепереработки, метод конечных разностей, метод кубического сплайна, метод регуляризации Тихонова

Для цитирования: Лэ В. Х., Фирсов А. Н. Метод регуляризации Тихонова для решения обратной задачи в математической модели кинетики процесса нефтепереработки // Вестник кибернетики. 2022. № 4 (48). С. 49–58. DOI 10.34822/1999-7604-2022-4-49-58.

Original article

TIKHONOV'S REGULARIZATION METHOD FOR SOLVING THE INVERSE PROBLEM IN A MATHEMATICAL MODEL OF THE KINETICS OF OIL REFINING PROCESS

Van Huyen Le^{1✉}, Andrey N. Firsov²

^{1,2} Peter the Great St. Petersburg Polytechnic University, Saint Petersburg, Russia

¹ huyenlevan120193@gmail.com ✉, <https://orcid.org/0000-0002-8645-821X>

² anfirs@yandex.ru, <https://orcid.org/0000-0003-4483-3381>

Abstract. The study presents a solution to the inverse problem of approximate constants determination of response rate according to the given concentrations of initial substance and products in the framework of mathematical model of kinetics of oil refining process. In the course of the study, the finite difference method, the data interpolation method (the cubic spline), and the Tikhonov regularization method were used. Response rate approximate constants are identified. They are unique and depend on the initial data. Response rate approximate constants aid in forecasting concentrations of the initial substance and products at any moment and determining indicators in oil refining process reactions, such as energy activation, temperature, and other external conditions.

Keywords: inverse problem, ill-posed problem, mathematical modeling, chemical kinetics, oil refining process, finite difference method, cubic spline method, Tikhonov's regularization method

For citation: Le V. H., Firsov A. N. Tikhonov's Regularization Method for Solving the Inverse Problem in a Mathematical Model of the Kinetics of Oil Refining Process // Proceedings in Cybernetics. 2022. No. 4 (48). P. 49–58. DOI 10.34822/1999-7604-2022-4-49-58.

ВВЕДЕНИЕ

В химической науке в целом и в физической химии, в частности, есть особая область – химическая кинетика, изучающая механизмы и закономерности протекания химических процессов во времени. Химическая кинетика рассматривает и устанавливает зависимости скорости химических реакций от концентраций реагентов, температуры и других внешних условий. С момента появления химическая кинетика всегда была краеугольным камнем, на котором стоит химическая промышленность – нефтехимия, нефтепереработка и производство полимеров [1].

Современный этап развития областей нефтехимии и нефтепереработки характеризуется внедрением методов математического моделирования для описания процессов подготовки и переработки нефтяного сырья с целью улучшений технологий. Надежная информация о параметрах моделей и закономерности их изменения является основой правильного математического описания нефтяных систем и процессов и гарантией верных технологических решений [2].

Процесс гидрокрекинга используется с целью увеличения количества вырабатываемых светлых продуктов нефти. В процессе гидрокрекинга могут перерабатываться легкие и тяжелые дистилляты первичной переработки в смеси с водородсодержащим газом на комплексных катализаторах. В современной промышленности гидрокрекинг является процессом получения ценного топлива – керосинового и дизельного [3].

Для процессов нефтепереработки и нефтехимии характерен сложный состав реагирующей смеси, большое количество одновременно протекающих реакций и взаимных превращений реагентов. Структура кинетических уравнений реакций зависит от физико-химических представлений состава реагирующей смеси и получаемых продуктов. На основе реакции гидрокрекинга были построены математические модели одного из процессов нефтепереработки с использованием системы обыкновенных дифференциальных уравнений для описания математической модели кинетики [4–6]. Для этого взята гипотетическая гомогенная химическая реакция превращения

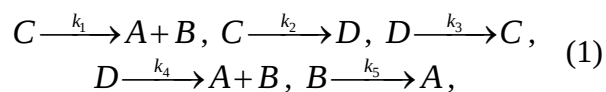
углеводородов парафинового ряда в процессе каталитического крекинга [4].

Решение данной системы уравнений позволило бы прогнозировать и управлять превращением концентраций исходного вещества и продуктов и решать проблемы масштабного перехода от лабораторных исследований к работе промышленных агрегатов [5].

МАТЕРИАЛЫ И МЕТОДЫ

Математическая модель кинетики процесса нефтепереработки. Задача математического моделирования каталитического крекинга рассмотрена в [2, 4–7]. В рамках данной работы будем рассматривать математическую модель кинетики процесса нефтепереработки, которая представляет систему однородных дифференциальных уравнений первого порядка с постоянными коэффициентами [4]. Пусть исходная смесь состоит из одного тяжелого углеводорода C . Под действием температуры и соударений углеводород C распадается на углеводороды A , B , а также превращается в изомер D с тем же количеством атомов углерода, что и в исходной молекуле. Вещество D также распадается на A и B , либо обратно превращается в C . Пусть продукты реакции A , B – это более легкие углеводороды, и с ними никаких превращений далее не происходит [4].

Схема химических превращений в описанном выше процессе каталитического крекинга будет иметь следующий вид:



где k_1, k_2, k_3, k_4, k_5 – константы скорости реакций, c^{-1} [4].

Скорость реакций в схеме превращений (1) определяется следующими формулами:

$$\begin{aligned} v_1 &= -\frac{dC}{dt} = \frac{dA}{dt} = \frac{dB}{dt} = k_1[C], \\ v_2 &= -\frac{dC}{dt} = \frac{dD}{dt} = k_2[C], \\ v_3 &= -\frac{dD}{dt} = \frac{dC}{dt} = k_3[D], \end{aligned} \quad (2)$$

$$v_4 = -\frac{dD}{dt} = \frac{dA}{dt} = \frac{dB}{dt} = k_4[B],$$

$$v_5 = -\frac{dB}{dt} = \frac{dA}{dt} = k_5[B],$$

где v_1, v_2, v_3, v_4, v_5 – скорость реакций, $\frac{\text{моль}}{\text{л.с}}$;

$\frac{dA}{dt}, \frac{dB}{dt}, \frac{dC}{dt}, \frac{dD}{dt}$ – изменение концентрации веществ A, B, C, D за малое время, $\frac{\text{моль}}{\text{л.с}}$; $[B], [C], [D]$ – концентрация ве-

ществ $B, C, D, \frac{\text{моль}}{\text{л}}$. Из (2) следует, что изменение концентрации веществ A, B, C, D за малое время определяются по следующим формулам:

$$\begin{aligned} \frac{dA}{dt} &= k_1[C] + k_4[D] + k_5[B], \\ \frac{dB}{dt} &= k_1[C] + k_4[D] - k_5[B], \\ \frac{dC}{dt} &= -k_1[C] - k_2[C] + k_3[D], \\ \frac{dD}{dt} &= k_2[C] - k_3[D] - k_4[D]. \end{aligned} \quad (3)$$

Математическая модель кинетики реакции каталитического крекинга, описанная схемой превращений (1) и (3), представляет собой систему дифференциальных уравнений [5]:

$$\begin{cases} \frac{dy_1(t)}{dt} = k_5y_2(t) + k_1y_3(t) + k_4y_4(t); \\ \frac{dy_2(t)}{dt} = -k_5y_2(t) + k_1y_3(t) + k_4y_4(t); \\ \frac{dy_3(t)}{dt} = -(k_1 + k_2)y_3(t) + k_3y_4(t); \\ \frac{dy_4(t)}{dt} = k_2y_3(t) - (k_2 + k_4)y_4(t), \end{cases} \quad (4)$$

где $y_1(t), y_2(t), y_3(t), y_4(t)$ – концентрация веществ A, B, C, D в момент времени t [5]. Предположим, что в начальный момент времени $t=0$ концентрация веществ A, B, C, D равна $y_1(0), y_2(0), y_3(0), y_4(0)$ [5].

Рассмотрим две следующие задачи, построенные в рамках математической модели кинетики реакции каталитического крекинга (4).

Задача 1. По заданным константам скорости реакций $k_1 = k_{10}, k_2 = k_{20}, k_3 = k_{30}, k_4 = k_{40}, k_5 = k_{50}$ и концентрациям исходного вещества продуктов A, B, C, D в начальный момент времени $t=0$ определить $y_1(t), y_2(t), y_3(t), y_4(t)$.

Введенные условия представляет собой задачу построения решения линейной системы дифференциальных уравнений с заданными начальными условиями (задачу Коши). Эту задачу будем называть прямой задачей в рамках математической модели (4). Решение задачи 1 позволяет определить концентрацию веществ A, B, C, D в любой момент времени. Можно переписать (4)

в матрично-векторном виде: $\frac{dY(t)}{dt} = AY(t)$, где

$A = (a_{ij})_{i,j=1}^4$ – матрица с элементами $k_1, k_2, k_3,$

k_4, k_5 ; $Y(t) = (y_1(t), y_2(t), y_3(t), y_4(t))^T$. Нетрудно доказать, что прямая задача корректно поставлена по Адамару (см. определение корректной задачи в [7–10]). Действительно, существование и единственность решения задачи 1 можно вывести из теоремы существования и единственности решения задачи Коши. Решение задачи 1 имеет следующий вид: $Y(t) = e^{At}Y(0)$, где

$Y(0) = (y_1(0), y_2(0), y_3(0), y_4(0))^T$ – вектор начальных условий; e^{At} определяется соотно-

шением: $e^{At} = \sum_{k=0}^{\infty} \frac{(At)^k}{k!}$. В данной работе рас-

смотрим случай, когда начальное условие $Y(0)$ не изменяется. Будем доказывать, что при малом изменении матрицы A решение $Y(t)$ задачи 1 тоже мало изменяется.

То есть, $\|Y_1(t) - Y_0(t)\| \rightarrow 0$ при $\|A_1 - A_0\| \rightarrow 0$,

где $Y_1(t) = e^{A_1 t}Y_0(0)$, $Y_0(t) = e^{A_0 t}Y_0(0)$ – решение задачи 1 при $A = A_1, A = A_0$. Разность между $Y_1(t)$ и $Y_0(t)$ определяется неравен-

ством: $\|Y_1(t) - Y_0(t)\| \leq \|(e^{At} - e^{A_0t})\| \|Y_0(0)\|$.

Имеем:

$$\begin{aligned} \|(e^{At} - e^{A_0t})\| &= \left\| \sum_{k=0}^{\infty} \frac{(At)^k}{k!} - \sum_{k=0}^{\infty} \frac{(A_0t)^k}{k!} \right\| = \\ &= \left\| \sum_{k=0}^{\infty} \frac{t^k}{k!} (A^k - A_0^k) \right\| \leq \sum_{k=0}^{\infty} \frac{t^k}{k!} \|A^k - A_0^k\|. \end{aligned} \quad (5)$$

Проанализируем (5): при $k = 1$ имеем $\|A_1 - A_0\| \rightarrow 0$. Пусть $A_1 = A_0 + \Delta$. Из $\|A_1 - A_0\| \rightarrow 0$ следует, что $\|\Delta\| \rightarrow 0$. При $k = 2$ имеем $\|A_1^2 - A_0^2\| \leq \|\Delta\| (2\|A_0\| + \|\Delta\|) \rightarrow 0$ при $\|\Delta\| \rightarrow 0$. При $k = 3$ имеем $\|A_1^3 - A_0^3\| \leq \|\Delta\| (3\|A_0\|^2 + 3\|A_0\|\|\Delta\| + \|\Delta\|^2)$ при $\|\Delta\| \rightarrow 0$. Делая то же самое для $k = 4, 5, 6, \dots$, получаем $\|A_1^k - A_0^k\| \rightarrow 0$ при $\|\Delta\| \rightarrow 0$. Отсюда $\|(e^{At} - e^{A_0t})\| \leq \sum_{k=0}^{\infty} \frac{t^k}{k!} \|A_1^k - A_0^k\| \rightarrow 0$ при $\|\Delta\| \rightarrow 0$. В итоге получаем $\|Y_1(t) - Y_0(t)\| \rightarrow 0$ при $\|A_1 - A_0\| \rightarrow 0$, т. е. решение задачи 1 непрерывно зависит от матричного оператора A .

Задача 2. По заданным концентрациям исходного вещества и продуктов A, B, C, D в моменты времени $t_i, i = 1, 2, \dots$ определить k_1, k_2, k_3, k_4, k_5 (т. е. $k_{10}, k_{20}, k_{30}, k_{40}, k_{50}$).

Задачу 2 будем называть обратной задачей по отношению к задаче 1 в рамках математической модели (4). В прикладных исследованиях, как правило, концентрацию веществ A, B, C, D можно измерить в любой момент времени t , т. е. нам известны $y_1(t), y_2(t), y_3(t), y_4(t)$. Для решения обратной задачи 2 будем использовать следующие методы: метод конечных разностей, метод кубического сплайна, метод регуляризации Тихонова. В результате проведенных расчетов будут найдены приближенные константы скорости реакций.

Метод конечных разностей. Применяя метод конечных разностей, из (4) получаем систему алгебраических уравнений относительно k_1, k_2, k_3, k_4, k_5 :

$$\begin{cases} \frac{y_1(t+h) - y_1(t-h) - o(h^3)}{2h} = y_3(t)k_1 + y_4(t)k_4 + y_2(t)k_5, \\ \frac{y_2(t+h) - y_2(t-h) - o(h^3)}{2h} = y_3(t)k_1 + y_4(t)k_4 - y_2(t)k_5, \\ \frac{y_3(t+h) - y_3(t-h) - o(h^3)}{2h} = -y_3(t)k_1 - y_3(t)k_2 + y_4(t)k_3, \\ \frac{y_4(t+h) - y_4(t-h) - o(h^3)}{2h} = [y_3(t) - y_4(t)]k_2 - y_4(t)k_4. \end{cases} \quad (6)$$

Система (6) содержит пять неизвестных k_1, k_2, k_3, k_4, k_5 . Подставляя $t = t_1, t = t_2$ в (6), получаем систему восьми алгебраических уравнений. Из них выбираем любых пять уравнений и имеем следующую систему уравнений:

$$\begin{cases} \frac{y_1(t_1+h) - y_1(t_1-h) - o(h^3)}{2h} = y_3(t_1)k_1 + y_4(t_1)k_4 + y_2(t_1)k_5, \\ \frac{y_2(t_1+h) - y_2(t_1-h) - o(h^3)}{2h} = y_3(t_1)k_1 + y_4(t_1)k_4 - y_2(t_1)k_5, \\ \frac{y_3(t_1+h) - y_3(t_1-h) - o(h^3)}{2h} = -y_3(t_1)k_1 - y_3(t_1)k_2 + y_4(t_1)k_3, \\ \frac{y_4(t_1+h) - y_4(t_1-h) - o(h^3)}{2h} = [y_3(t_1) - y_4(t_1)]k_2 - y_4(t_1)k_4, \\ \frac{y_1(t_2+h) - y_1(t_2-h) - o(h^3)}{2h} = y_3(t_2)k_1 + y_4(t_2)k_4 + y_2(t_2)k_5. \end{cases} \quad (7)$$

Систему (7) можно представить в векторно-матричном виде $AK = B$, где $A = (a_{ij})_{i,j=1}^5$ – матрица с элементами $y_2(t_1), y_3(t_1), y_4(t_1), y_2(t_2), y_3(t_2), y_4(t_2)$; $K = (k_1, k_2, k_3, k_4, k_5)^T$; B – вектор левой части. Отметим, что процесс кинетики реакции каталитического крекинга произошел, поэтому система (7) всегда имеет хотя бы одно решение. Искомое решение $K_0 = (k_{10}, k_{20}, k_{30}, k_{40}, k_{50})^T$ задачи 2 является точным решением системы (7).

Пренебрегая $o(h^3)$, из (7) получаем систему уравнений:

$$\begin{cases} \frac{y_1(t_1+h) - y_1(t_1-h)}{2h} = y_3(t_1)k_1 + y_4(t_1)k_4 + y_2(t_1)k_5, \\ \frac{y_2(t_1+h) - y_2(t_1-h)}{2h} = y_3(t_1)k_1 + y_4(t_1)k_4 - y_2(t_1)k_5, \\ \frac{y_3(t_1+h) - y_3(t_1-h)}{2h} = -y_3(t_1)k_1 - y_3(t_1)k_2 + y_4(t_1)k_3, \\ \frac{y_4(t_1+h) - y_4(t_1-h)}{2h} = [y_3(t_1) - y_4(t_1)]k_2 - y_4(t_1)k_4, \\ \frac{y_1(t_2+h) - y_1(t_2-h)}{2h} = y_3(t_2)k_1 + y_4(t_2)k_4 + y_2(t_2)k_5. \end{cases} \quad (8)$$

Метод кубического сплайна. Значения $y_2(t_1)$, $y_3(t_1)$, $y_4(t_1)$, $y_2(t_2)$, $y_3(t_2)$, $y_4(t_2)$ получаются из эксперимента, а $y_1(t_1+h)$, $y_1(t_1-h)$, $y_2(t_1+h)$, $y_2(t_1-h)$, $y_3(t_1+h)$, $y_3(t_1-h)$, $y_4(t_1+h)$, $y_4(t_1-h)$, $y_1(t_2+h)$, $y_1(t_2-h)$ получаются методом кубического сплайна.

Метод регуляризации Тихонова. Поскольку $y_2(t_1)$, $y_3(t_1)$, $y_4(t_1)$, $y_2(t_2)$, $y_3(t_2)$, $y_4(t_2)$ измеряются экспериментально, то они могут содержать ошибки измерения и округления. Значения $y_1(t_1+h)$, $y_1(t_1-h)$, $y_2(t_1+h)$, $y_2(t_1-h)$, $y_3(t_1+h)$, $y_3(t_1-h)$, $y_4(t_1+h)$, $y_4(t_1-h)$, $y_1(t_2+h)$, $y_1(t_2-h)$ найдены интерполированием, поэтому они также могут содержать не только ошибки округления, но и ошибки интерполяции. Можно переписать (8) в виде: $A_h K = B_b$, где A_h – приближение к матрице A , а B_b – приближение к вектору B . Задача решения системы (8) некорректно поставлена, потому что при малом изменении A_h и B_b ее решение может сильно измениться. В этом случае для решения системы (8) будем использовать метод регуляризации Тихонова [7–10]. Необходимо найти приближение к искомому решению K_0 , которое является единственным и непрерывно зависит от A_h , B_b .

Будут найдены приближенные константы скорости реакций по следующему условию:

$$\|A_h K - B_b\|^2 + \alpha \|K\|^2 \rightarrow \min_{\alpha} \min_K, \quad (9)$$

где $\alpha = \text{const} > 0$ – параметр регуляризации.

Из (2) вытекает регуляризирующее уравнение:

$$A_h^* A_h K + \alpha K = A_h^* B_b, \quad (10)$$

где A_h^* – сопряженный к матрице A_h . Решение системы (10), обозначаемое $K_\alpha = (k_{1\alpha}, k_{2\alpha}, k_{3\alpha}, k_{4\alpha}, k_{5\alpha})^T$, является регуляризованным решением уравнения $A_h K = B_b$ [7–10].

В методе регуляризации Тихонова необходимо найти параметр регуляризации α так, что K_α стремится к искомому решению K_0 , т. е. $\|K_\alpha - K_0\| \rightarrow 0$.

Проанализируем условие (9) и уравнение (10). С увеличением же параметра α регуляризованное решение K_α становится глаже и устойчивей, т. е. уменьшается норма решения $\|K_\alpha\|^2$, но увеличивается невязка $\|A_h K_\alpha - B_b\|^2$. Истина – посередине, т. е. при некотором умеренном параметре α регуляризованное решение K_α будет иметь и умеренную гладкость, и умеренную невязку. Существует несколько способов выбора параметра регуляризации [7–10]. В данной работе параметр α будет выбран исходя из непрерывности решения задачи 1.

Рассмотрим задачу 1. Матричный оператор A с элементами $k_1 = k_{1\alpha}$, $k_2 = k_{2\alpha}$, $k_3 = k_{3\alpha}$, $k_4 = k_{4\alpha}$, $k_5 = k_{5\alpha}$ обозначаем A_α , а с элементами $k_1 = k_{10}$, $k_2 = k_{20}$, $k_3 = k_{30}$, $k_4 = k_{40}$, $k_5 = k_{50}$ обозначаем A_0 . Решение задачи 1 с операторами $A = A_\alpha$ и $A = A_0$ обозначаем $Y_\alpha(t)$ и $Y_0(t)$. При малом изменении оператора A_α решение $Y_\alpha(t)$ тоже мало изменяется. Из $\|K_\alpha - K_0\| \rightarrow 0$ следует, что $\|A_\alpha - A_0\| \rightarrow 0$. Из свойства непрерывности решения задачи 1 следует, что $\|Y_\alpha(t) - Y_0(t)\| \rightarrow 0$ при $\|A_\alpha - A_0\| \rightarrow 0$. Отсюда $Y_\alpha(t)$ непрерывно зависит от K_α . При этом будет выбран параметр α так, что $\|Y_\alpha(t) - Y_0(t)\|$ достигает минимума. Для этого будем подставлять найденные приближенные константы скорости реакций $k_{1\alpha}$, $k_{2\alpha}$, $k_{3\alpha}$, $k_{4\alpha}$, $k_{5\alpha}$ в (4) и решать прямую задачу. Из решения прямой задачи рассчитываются концентрации веществ A , B , C , D в разные моменты времени, т. е. $Y_\alpha(t)$. Расчетные концентрации веществ A , B , C , D будут сравниваться с измеренными значениями.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Рассмотрим пример. Пусть нам известны концентрации веществ A, B, C, D в разные

моменты времени, т. е. $y_1(t), y_2(t), y_3(t), y_4(t)$ (табл. 1).

Таблица 1

Измеренные концентрации веществ A, B, C, D

$t(c)$	$y_1(t)$	$y_2(t)$	$y_3(t)$	$y_4(t)$
0	0	0	90	10
30	77,76206	12,47561	48,11464	6,76653
60	132,29327	7,46789	25,74512	4,37430
90	162,81084	4,12939	13,78771	2,74217
120	179,58861	2,26780	7,39035	1,68145
150	188,79792	1,24467	3,96468	1,01403
180	193,85216	0,68309	2,12871	0,60366
210	196,62600	0,37489	1,14389	0,35567
240	198,14831	0,20574	0,61518	0,20779
270	198,98377	0,11291	0,33111	0,12055

Примечание: составлено авторами на основании данных, полученных в ходе расчета: $t(c)$ – время, измеряемое в секундах; $y_1(t), y_2(t), y_3(t), y_4(t)$ – концентрация веществ A, B, C, D в момент времени t .

Используя метод кубического сплайна, вычислим $y_1(t_1+h), y_1(t_1-h), y_2(t_1+h), y_2(t_1-h), y_3(t_1+h), y_3(t_1-h), y_4(t_1+h), y_4(t_1-h), y_1(t_2+h), y_1(t_2-h)$ и построим систему (8) при $t_1 = 200, t_2 = 220, h = 0,001$. Решение системы (8) будет найдено методом регуляризации Тихонова. В результате проведенных расчетов находим приближенные константы скорости реакций $k_{1\alpha}, k_{2\alpha}, k_{3\alpha}, k_{4\alpha}, k_{5\alpha}$. Как указано выше, будет выбран параметр α так, что $\|Y_\alpha(t) - Y_0(t)\|$ достигает минимума, т. е. отклонение между расчетными и измеренными данными достигает минимума. Чтобы проверить совпадение между расчетными и измеренными значениями, будем использовать статистические параметры: коэффициент корреляции Пирсона R^2 и показатель эффективности Нэша – Сатклиффа (Nash–Sutcliffe efficiency, NSE) [11–13].

Коэффициент R^2 ($0 \leq R^2 \leq 1$) помогает предсказать, существует ли линейная корреляция между измеренными и расчетными концентрациями веществ A, B, C, D . Если $R^2 = 0$, то полно отсутствует линейная корреляция между измеренными и расчетными

концентрациями веществ A, B, C, D . Если $R^2 = 1$, то существует идеальная линейная корреляция между измеренными и расчетными концентрациями веществ A, B, C, D [11–13].

Показатель NSE ($-\infty < NSE \leq 1$) указывает, насколько хорошо график измеренных и расчетных данных соответствует линии 1:1. Если $NSE = 0$, то полностью отсутствует совпадение между измеренными и расчетными концентрациями веществ A, B, C, D . Если $NSE = 1$, то существует идеальное совпадение между расчетными и измеренными концентрациями веществ A, B, C, D [11–13].

Рис. 1 показывает зависимость от параметра регуляризации α суммы показателей NSE , рассчитанной формулой: $\sum NSE = NSE(y_1) + NSE(y_2) + NSE(y_3) + NSE(y_4)$, где $NSE(y_1), NSE(y_2), NSE(y_3), NSE(y_4)$ – показатели NSE при оценке совпадения между измеренными и расчетными концентрациями веществ A, B, C, D . Чем выше $\sum NSE$, тем лучше. Будем искать параметр α так, чтобы $\sum NSE = NSE(y_1) + NSE(y_2) + NSE(y_3) + NSE(y_4) \rightarrow 4$.

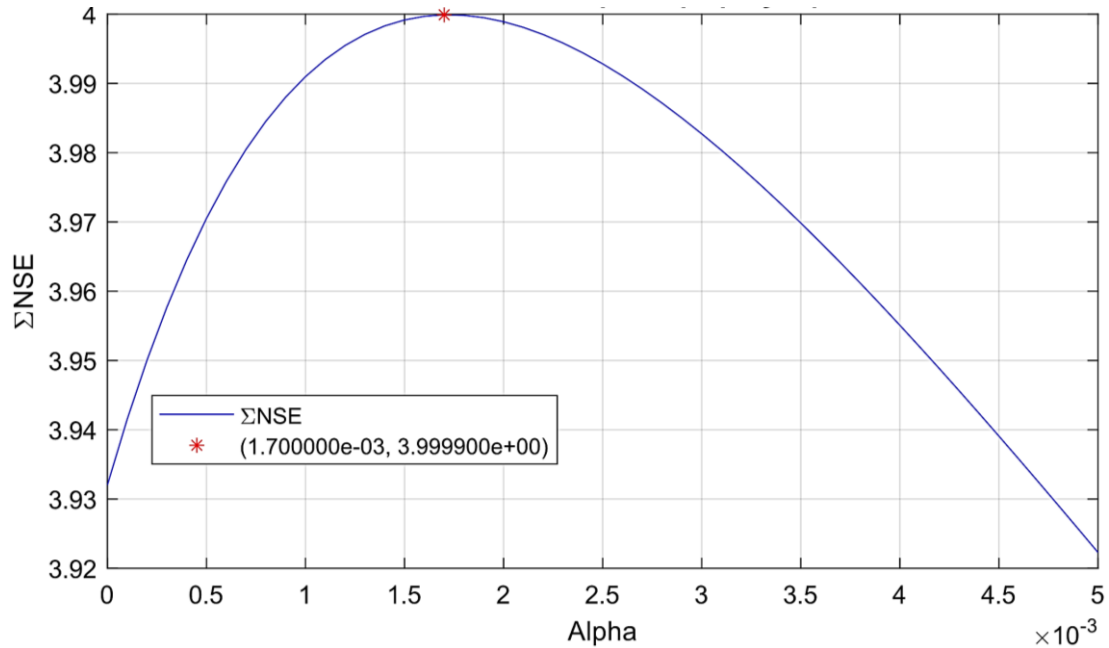


Рис. 1. Зависимость суммы показателей эффективности Нэша – Сатклиффа от параметра регуляризации
Примечание: составлено авторами.

При $\alpha = 0,0017$ имеем $k_{1\alpha} = 0,02001$, $k_{2\alpha} = 0,00103$, $k_{3\alpha} = 0,00114$, $k_{4\alpha} = 0,02008$, $k_{5\alpha} = 0,09936$. Расчетные концентрации ве-

Таблица 2

Расчетные концентрации веществ A, B, C, D при $\alpha = 0,0017$

$t(c)$	$y_1(t)$	$y_2(t)$	$y_3(t)$	$y_4(t)$
0	0	0	90	10
30	77,74094	12,56051	48,08937	6,78712
60	132,32963	7,53090	25,72160	4,39339
90	162,89439	4,16552	13,77158	2,75522
120	179,70009	2,28784	7,38073	1,68919
150	188,92534	1,25575	3,95948	1,01819
180	193,98870	0,68923	2,12614	0,60571
210	196,76777	0,37829	1,14276	0,35657
240	198,29311	0,20763	0,61477	0,20812
270	199,13032	0,11396	0,33102	0,12062

Примечание: составлено авторами на основании данных, полученных в ходе расчета: $t(c)$ – время, измеряемое в секундах; $y_1(t)$, $y_2(t)$, $y_3(t)$, $y_4(t)$ – концентрация веществ A, B, C, D в момент времени t .

Для вещества A , коэффициент $R^2 = 0,9999998401 \approx 1$ показывает, что существует линейная корреляция между измеренными и расчетными концентрациями; $NSE = 0,9999969829 \approx 1$ показывает, что существует совпадение между расчетными и измеренными концентрациями.

Для вещества B имеем, что $R^2 = 0,9999991314 \approx 1$, $NSE = 0,9999140278 \approx 1$ для вещества C : $R^2 = 0,9999998897 \approx 1$, $NSE = 0,9999997925 \approx 1$, а для вещества D : $R^2 = 0,9999949606 \approx 1$, $NSE = 0,9999895044 \approx 1$. Это показывает, что существует линей-

ная корреляция 1:1 между расчетными и измеренными концентрациями веществ A , B , C , D , т. е. расчетная концентрация в любой момент времени немного отличается от измеренной. Найденные приближенные константы $k_{1\alpha} = 0,02001$, $k_{2\alpha} = 0,00103$, $k_{3\alpha} = 0,00114$, $k_{4\alpha} = 0,02008$, $k_{5\alpha} = 0,09936$ можно принять как решение обратной задачи 2.

Рис. 2 показывает, как меняется концентрация веществ A , B , C , D с течением времени при $k_1 = 0,02001$, $k_2 = 0,00103$, $k_3 = 0,00114$, $k_4 = 0,02008$, $k_5 = 0,09936$.

На рис. 2 звездочками обозначены измеренные концентрации A , B , C , D . Кривые $y_1(t)$, $y_2(t)$, $y_3(t)$, $y_4(t)$ выражают изменение расчетной концентрации вещества A , B , C , D с течением времени. Видно, что измеренные значения очень близки к кривым. Это означает, что отклонение между измеренными и расчетными значениями очень малое, и согласуется с тем, что показатель $NSE(y_1) \approx 1$, $NSE(y_2) \approx 1$, $NSE(y_3) \approx 1$, $NSE(y_4) \approx 1$.

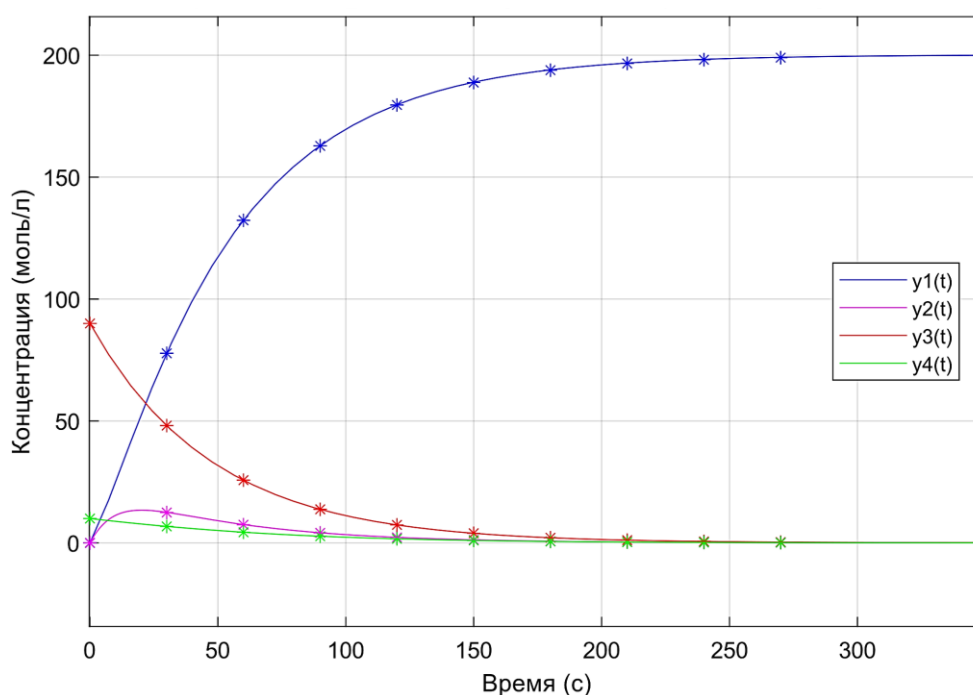


Рис. 2. Изменение концентрации вещества A , B , C , D при $k_1 = 0,02001$, $k_2 = 0,00103$, $k_3 = 0,00114$, $k_4 = 0,02008$, $k_5 = 0,09936$

Примечание: составлено авторами.

Из данных табл. 1 и 2 будем строить диаграммы рассеивания, чтобы показать, что существует линейная корреляция 1:1 между измеренными и рассчитанными концентрациями веществ A , B , C , D (рис. 3). На этих диаграммах координат каждые точки по горизонтальной оси выражают измеренную концентрацию, а по вертикальной оси – расчетную концентрацию. Видно, что точки, выражающие концентрацию веществ A , B , C , D ,

распределяются близко к линейной кривой 1:1. Это согласуется с тем, что существует линейная корреляция 1:1 между измеренными и расчетными концентрациями. Отсюда найденные приближенные константы скорости реакций $K_\alpha = (0,02001, 0,00103, 0,00114, 0,02008, 0,09936)^T$ можно принять как решение обратной задачи 2.

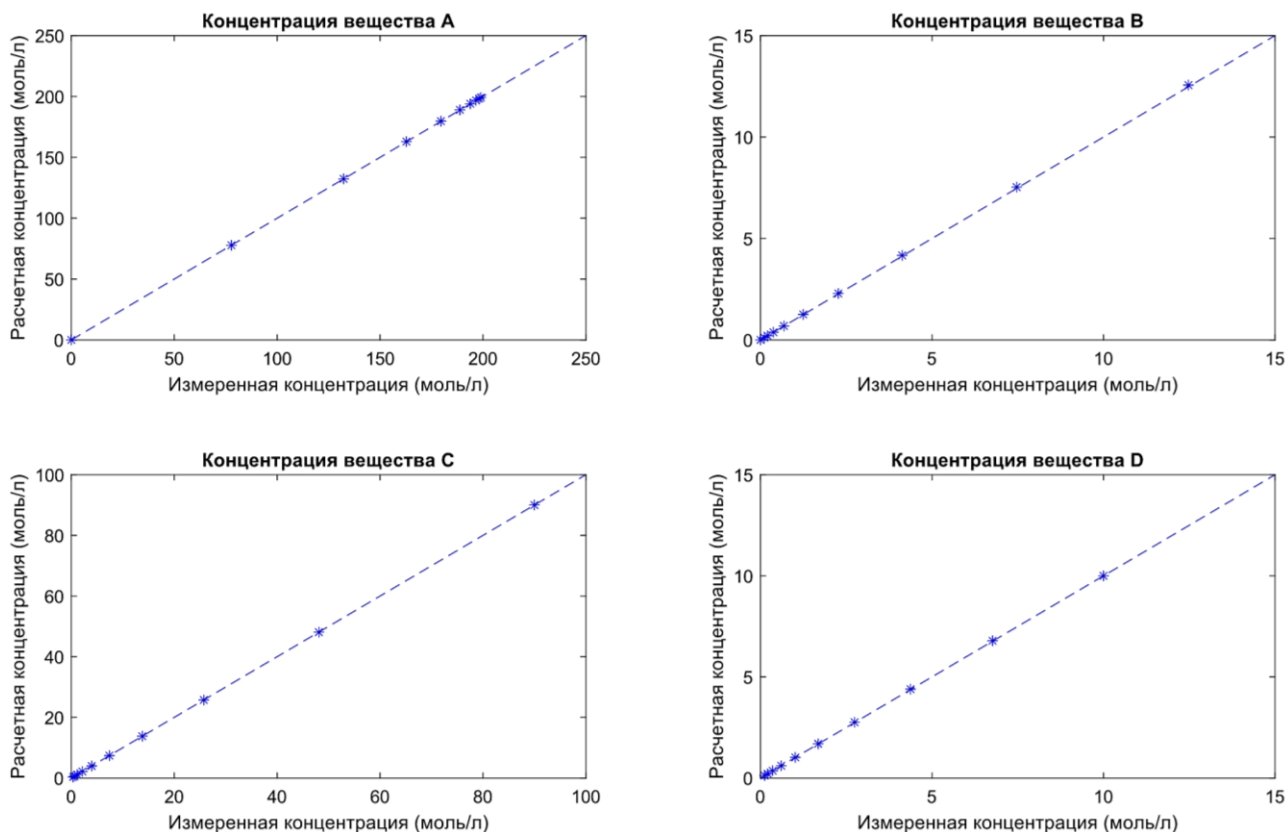


Рис. 3. Линейная корреляция 1:1 между измеренными и расчетными концентрациями вещества А, В, С, D при $k_1 = 0,02001$, $k_2 = 0,00103$, $k_3 = 0,00114$, $k_4 = 0,02008$, $k_5 = 0,09936$

Примечание: составлено авторами.

ЗАКЛЮЧЕНИЕ

В работе была рассмотрена математическая модель кинетики процесса нефтепереработки, описанная линейной системой дифференциальных уравнений с заданными начальными условиями (задача Коши). В рамках этой математической модели была поставлена обратная задача: по заданным концентрациям исходного вещества и продуктов определить константы скорости реакций. В результате проведенных расчетов найдены методом регуляризации Тихонова приближенные константы скорости реакций в про-

цессе нефтепереработки. Для проверки совпадения между измеренными и расчетными концентрациями исходного вещества и продуктов были использованы коэффициент корреляции Пирсона R^2 и показатель эффективности Нэша – Сатклиффа NSE . Установлено, что существует линейная корреляция и совпадение (т. е. линейная корреляция 1:1) между измеренными и расчетными концентрациями веществ А, С, D. В этом случае были приняты найденные приближенные константы скорости реакций в процессе нефтепереработки.

Список источников

1. Колинко П. А., Козлов Д. В. Химическая кинетика в курсе физической химии. Новосибирск : НГУ, 2013. 99 с.
2. Погореловский М. А., Микшина В. С., Назина Н. Б. К вопросу математического моделирования кинетики реакций процессов нефтепереработки // Вестник кибернетики. 2017. № 4. С. 92–99.
3. Абдуллин А. И., Сираев И. Р. Гидрокрекинг как процесс получения дизельного топлива //

References

1. Kolinko P. A., Kozlov D. V. Khimicheskaya kinetika v kurse fizicheskoi khimii. Novosibirsk : Novosibirsk State University, 2013. 99 p. (In Russian).
2. Pogorelovsky M. A., Mikshina V. S., Nazina N. B. Mathematical Modeling Issue of Reaction Kinetics of Refining Processes // Proceedings in Cybernetics. 2017. No. 4. P. 92–99. (In Russian).
3. Abdullin A. I., Siraev I. R. Hidrokrekning kak protsess polucheniia dizelnogo topliva // Bulletin of

- Вестник технолог. ун-та. 2016. Т. 19, № 10. С. 41–43.
4. Лысенкова С. А. О математическом моделировании каталитического крекинга // Вестник кибернетики. 2018. № 4. С. 107–110.
 5. Заикин П. В., Лысенкова С. А., Микшина В. С. Аналитическое решение системы дифференциальных уравнений математической модели кинетики процесса нефтепереработки // Вестник кибернетики. 2018. № 2. С. 120–126.
 6. Микшина В. С. О математическом моделировании каталитического крекинга : моногр. СПб. : Научные технологии, 2021. 120 с.
 7. Ольховой А. Ф. Введение в теорию обратных и некорректных задач : моногр. 2012. 116 с.
 8. Colton D., Kress R. Ill-Posed Problems // Inverse Acoustic and Electromagnetic Scattering Theory. 2019. P. 111–136.
 9. Kabanikhin S. I. Inverse Problems of Natural Science // Comput Math Math Phys. 2020. Vol. 60, Is. 6. P. 911–914.
 10. Kirsch A. An Introduction to the Mathematical Theory of Inverse Problems. 2021. 400 p.
 11. Борщ С. В., Симонов Ю. А., Христофоров А. В. Эффективность моделирования и прогнозирования речного стока // Гидрометеоролог. исслед. и прогнозы. 2020. Т. 375, № 1. С. 176–189.
 12. Suryaningtyas L. S., Ery S., Rispiningtati R. Hydrological Analysis of TRMM (Tropical Rainfall Measuring Mission) Data in Lesti Sub Watershed // Civ Environ Sci. 2020. Vol. 3, No. 1. P. 18–30.
 13. Zhong X., Dutta U. Engaging Nash-Sutcliffe Efficiency and Model Efficiency Factor Indicators in Selecting and Validating Effective Light Rail System Operation and Maintenance Cost Models // J Traffic Transp Eng. 2015. Vol. 3, No. 5. P. 255–265.
 - the Technological University. 2016. Vol. 19, No. 10. P. 41–43. (In Russian).
 4. Lysenkova S. A. On Mathematical Modeling of Catalytic Cracking // Proceedings in Cybernetics. 2018. No. 4. P. 107–110. (In Russian).
 5. Zaikin P. V., Lysenkova S. A., Mikshina V. S. Analytical Solution of Differential Equation System of Mathematical Model of Refining Process Kinetics // Proceedings in Cybernetics. 2018. No. 2. P. 120–126. (In Russian).
 6. Mikshina V. S. O matematicheskom modelirovanii kataliticheskogo krekinga : Monograph. St. Petersburg : Naukoemkie tekhnologii, 2021. 120 p. (In Russian).
 7. Olkhovoi A. F. Vvedenie v teoriyu obratnykh i nekorrektnykh zadach : Monograph. 2012. 116 p. (In Russian).
 8. Colton D., Kress R. Ill-Posed Problems // Inverse Acoustic and Electromagnetic Scattering Theory. 2019. P. 111–136.
 9. Kabanikhin S. I. Inverse Problems of Natural Science // Comput Math Math Phys. 2020. Vol. 60, Is. 6. P. 911–914.
 10. Kirsch A. An Introduction to the Mathematical Theory of Inverse Problems. 2021. 400 p.
 11. Borshch S. V., Simonov Yu. A., Khristoforov A. V. Efficiency of Streamflow Modeling and Forecasting // Hydrometeorological Research and Forecasting. 2020. Vol. 375, No. 1. P. 176–189. (In Russian).
 12. Suryaningtyas L. S., Ery S., Rispiningtati R. Hydrological Analysis of TRMM (Tropical Rainfall Measuring Mission) Data in Lesti Sub Watershed // Civ Environ Sci. 2020. Vol. 3, No. 1. P. 18–30.
 13. Zhong X., Dutta U. Engaging Nash-Sutcliffe Efficiency and Model Efficiency Factor Indicators in Selecting and Validating Effective Light Rail System Operation and Maintenance Cost Models // J Traffic Transp Eng. 2015. Vol. 3, No. 5. P. 255–265.

Информация об авторах

В. Х. Лэ – аспирант.

А. Н. Фирсов – доктор технических наук, профессор.

Information about the authors

V. H. Le – Postgraduate.

A. N. Firsov – Doctor of Sciences (Engineering), Professor.

Научная статья
УДК 681.51
doi: 10.34822/1999-7604-2022-4-59-68

ПРИМЕНЕНИЕ МЕТОДА ПРОСТРАНСТВА СОСТОЯНИЙ ДЛЯ ОБОБЩЕННОЙ СИСТЕМЫ ТИПА «ОДИН ВХОД – ОДИН ВЫХОД»

Инна Васильевна Музылева^{1✉}, **Любовь Николаевна Языкова**²,
Виктор Николаевич Мещеряков³

^{1, 2, 3} *Липецкий государственный технический университет, Липецк, Россия*

¹ *lstu-miv2010@yandex.ru ✉, <https://orcid.org/0000-0001-5958-5144>*

² *yaln29@yandex.ru, <https://orcid.org/0000-0002-6035-5263>*

³ *mesherek@yandex.ru, <https://orcid.org/0000-0003-0984-5133>*

Аннотация. Объектом исследования являются аналитические и топологические методы расчета систем автоматического управления. Целью работы является построение практико-ориентированного способа применения метода пространства состояний раздела неметрической математики для исследования систем типа «один вход – один выход». Исследована обобщенная система n -го порядка в виде «вход – выход», разработана структурная схема ее моделирования согласно методу пространства состояний, апробированная на примере системы второго порядка с проверкой методом контрольного моделирования. Исследуемая система n -го порядка преобразована к виду «вход – внутреннее состояние – выход», разработана структурная схема ее моделирования по методу пространства состояний в различных системах визуальной симуляции, апробированная на примере системы третьего порядка с проверкой методом контрольного моделирования. Критерием правильности является тождественность графиков выходных сигналов исследуемой системы, полученных методом пространства состояний, методом структурного моделирования и заданием передаточной функции. Разработанный универсальный математический инструментарий позволяет проводить исследование систем типа «один вход – один выход» любой степени сложности.

Ключевые слова: пространство состояний, переменные состояния, SISO-система, структурный анализ, моделирование

Для цитирования: Музылева И. В., Языкова Л. Н., Мещеряков В. Н. Применение метода пространства состояний для обобщенной системы типа «один вход – один выход» // Вестник кибернетики. 2022. № 4 (48). С. 59–68. DOI 10.34822/1999-7604-2022-4-59-68.

Original article

APPLICATION OF THE STATE SPACE METHOD FOR A GENERALIZED SINGLE INPUT SINGLE OUTPUT SYSTEM

Inna V. Muzyleva^{1✉}, **Lyubov N. Yazykova**², **Viktor N. Meshcheryakov**³

^{1, 2, 3} *Lipetsk State Technical University, Lipetsk, Russia*

¹ *lstu-miv2010@yandex.ru ✉, <https://orcid.org/0000-0001-5958-5144>*

² *yaln29@yandex.ru, <https://orcid.org/0000-0002-6035-5263>*

³ *mesherek@yandex.ru, <https://orcid.org/0000-0003-0984-5133>*

Abstract. The object of the study is analytical and topological methods for automatic control systems engineering. The aim of the study is to build a practice-oriented approach for applying the state space method, which is part of nonmetric mathematics, to analyze the single input single output systems. A generalized n^{th} -order input output system is investigated. A structural scheme for such system modeling is developed according to the state space method. The scheme was tested on the example of the second order system using verification of the control modeling method. The n^{th} -order system is transformed into “input – internal state – output” type. The structural scheme of its modeling is developed according to the state space method in various visual

simulation systems and tested on the example of the third order system using verification of the control modeling method. The criterion of correctness is equality of graphs of output signals of the system analyzed, which were obtained by the state space method, the structural modeling method and the assignment of the transfer function. The developed universal mathematical tools make it possible to study single input single output systems of any complexity.

Keywords: state space, state variables, SISO system, structural analysis, modeling

For citation: Muzyleva I. V., Yazykova L. N., Meshcheryakov V. N. Application of the State Space Method for a Generalized Single Input Single Output System // Proceedings in Cybernetics. 2022. No. 4 (48). P. 59–68. DOI 10.34822/1999-7604-2022-4-59-68.

ВВЕДЕНИЕ

Исследование систем автоматического управления во временном домене [1] с помощью пространства состояний очень удобна по нескольким причинам. С одной стороны, по методическим соображениям, система характеризуется с помощью понятия «состояние системы», которому соответствует точка в определенном евклидовом пространстве, а поведение системы во временной области – это траектория этой точки [2]. С другой стороны, матрично-векторное описание системы компактно и наглядно, что позволяет работать с линейными и нелинейными системами с помощью одного и того же математического инструментария.

МАТЕРИАЛЫ И МЕТОДЫ

Обобщенная система автоматического управления (CAУ) типа SISO (single input single output) может быть представлена как «черный ящик» (рис. 1). Связь между входным и выходным сигналами этой системы в виде

«вход – выход» описывается дифференциальным уравнением, у которого в левой части размещается сумма всех производных выходного сигнала $x_{\text{ВЫХ}}(t)$, а в правой части – сумма всех производных входного сигнала $x_{\text{ВХ}}(t)$ [3, 4]:

$$a_n \frac{d^n x_{\text{ВЫХ}}(t)}{dt^n} + a_{n-1} \frac{d^{n-1} x_{\text{ВЫХ}}(t)}{dt^{n-1}} + \dots + a_1 \frac{dx_{\text{ВЫХ}}(t)}{dt} + a_0 x_{\text{ВЫХ}}(t) = b_m \frac{d^m x_{\text{ВХ}}(t)}{dt^m} + b_{m-1} \frac{d^{m-1} x_{\text{ВХ}}(t)}{dt^{m-1}} + \dots + b_1 \frac{dx_{\text{ВХ}}(t)}{dt} + b_0 x_{\text{ВХ}}(t). \quad (1)$$

Здесь $a_n, a_{n-1}, \dots, a_1, a_0, b_m, b_{m-1}, \dots, b_1, b_0$ – параметры, определяющие динамические свойства системы (параметры САУ). В том случае, когда параметры являются константами, САУ является линейной. Если хотя бы один параметр является нелинейным (например, изменяется во времени по релейному закону), САУ является нелинейной. Поэтому решение классической задачи теории автоматического управления – нахождение $x_{\text{ВЫХ}}(t)$ при задании $x_{\text{ВХ}}(t)$ путем решения дифференциального уравнения представляется затруднительным.

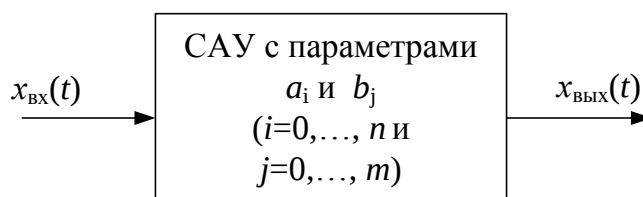


Рис. 1. Структурная схема обобщенной системы автоматического управления

Примечание: составлено авторами.

Универсальным математическим приемом, позволяющим решать указанную задачу, является метод пространства состояний [5]. Математическое уравнение системы можно представить в виде функциональной схемы, состоящей из элементарных динамических звеньев – идеальных интеграторов и сумматоров.

Введем обозначения, являющиеся традиционными для метода пространства состояний:

$$y_n = \frac{d^n x_{\text{ВЫХ}}(t)}{dt^n}; y_{n-1} = \frac{d^{n-1} x_{\text{ВЫХ}}(t)}{dt^{n-1}}; \dots y_1 = \frac{dx_{\text{ВЫХ}}(t)}{dt}; y_0 = x_{\text{ВЫХ}}(t); \\ g_m = \frac{d^m x_{\text{ВХ}}(t)}{dt^m}; g_{m-1} = \frac{d^{m-1} x_{\text{ВХ}}(t)}{dt^{m-1}}; \dots g_1 = \frac{dx_{\text{ВХ}}(t)}{dt}; g_0 = x_{\text{ВХ}}(t). \quad (2)$$

С учетом принятых обозначений, дифференциальное уравнение (1) преобразуется к виду:

$$\begin{aligned} a_n y_n + a_{n-1} y_{n-1} + \dots + a_1 y_1 + a_0 y_0 = \\ = b_m g_m + b_{m-1} g_{m-1} + \dots + b_1 g_1 + b_0 g_0. \end{aligned} \quad (3)$$

Здесь g_0 и y_0 – входной и выходной сигнал обобщенной САУ. Соответственно методу пространства состояний:

$$\begin{cases} g_m = \frac{1}{b_m} \left[\sum_{i=1}^n a_i y_i - \sum_{i=1}^{m-1} b_i g_i \right], \\ y_n = \frac{1}{a_n} \left[\sum_{i=1}^m b_i g_i - \sum_{i=1}^{n-1} a_i y_i \right]. \end{cases} \quad (4)$$

После выделения общих для обоих уравнений компонентов:

$$\Xi_1 = \sum_{i=1}^{m-1} b_i g_i, \quad \Xi_2 = \sum_{i=1}^{n-1} a_i y_i \quad (5)$$

получается:

$$\begin{cases} g_m = \frac{1}{b_m} (a_n y_n + \Xi_2 - \Xi_1), \\ y_n = \frac{1}{a_n} (b_m g_m + \Xi_1 - \Xi_2). \end{cases} \quad (6)$$

Соответствующая структура представлена на рис. 2.

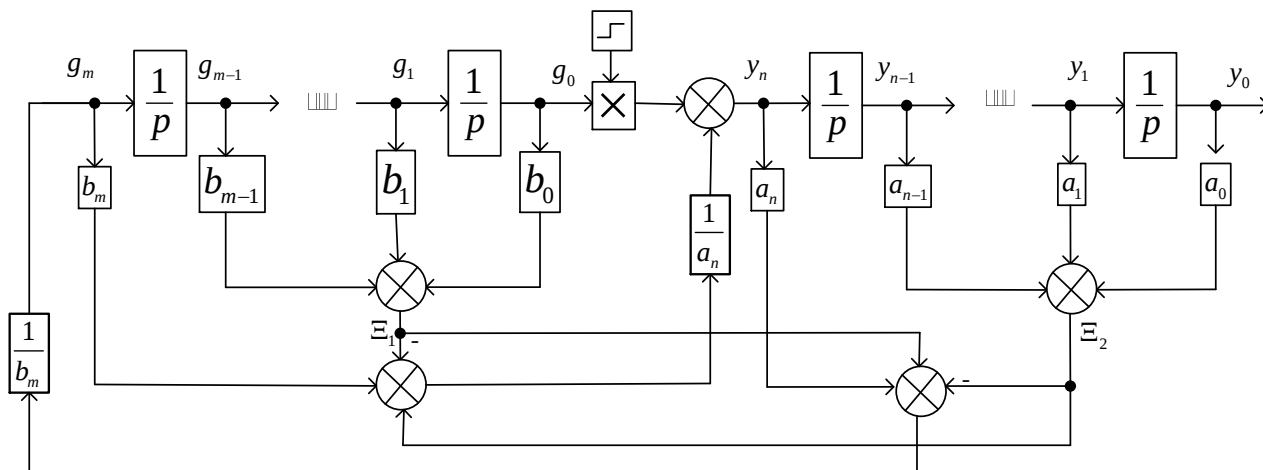


Рис. 2. Структурная модель пространства состояний вида «вход – выход» для системы автоматического управления n -го порядка

Примечание: составлено авторами на основании данных, полученных в исследовании.

Данная методика легко применима к системам любого порядка, но с $m = 0$. Численный пример рассмотрен для системы второго порядка, описываемой дифференциальным уравнением:

$$\begin{aligned} a_2 \frac{d^2 x_{\text{вых}}(t)}{dt^2} + a_1 \frac{dx_{\text{вых}}(t)}{dt} + \\ + a_0 x_{\text{вых}}(t) = b_0 x_{\text{вх}}(t). \end{aligned}$$

И, соответственно, передаточной функцией:

$$W(p) = \frac{b_0}{a_2 p^2 + a_1 p + a_0}.$$

Правильность расчета проверяется методом контрольного моделирования [6]. На два

входа плоттера подается два способа моделирования системы (рис. 3): структурная модель пространства состояний и передаточная функция. Графики совпадают, что подтверждает правильность схемы структурного моделирования. Поскольку в программе визуального моделирования VisSim блок «Передаточная функция» относится к группе «Линейные системы», параметры в данном примере представляют собой числовые значения. Этого достаточно для проверки правильности предлагаемой структуры. В дальнейшем параметры a_i можно задать в виде формул и применять предлагаемую структуру для исследования нелинейных систем.

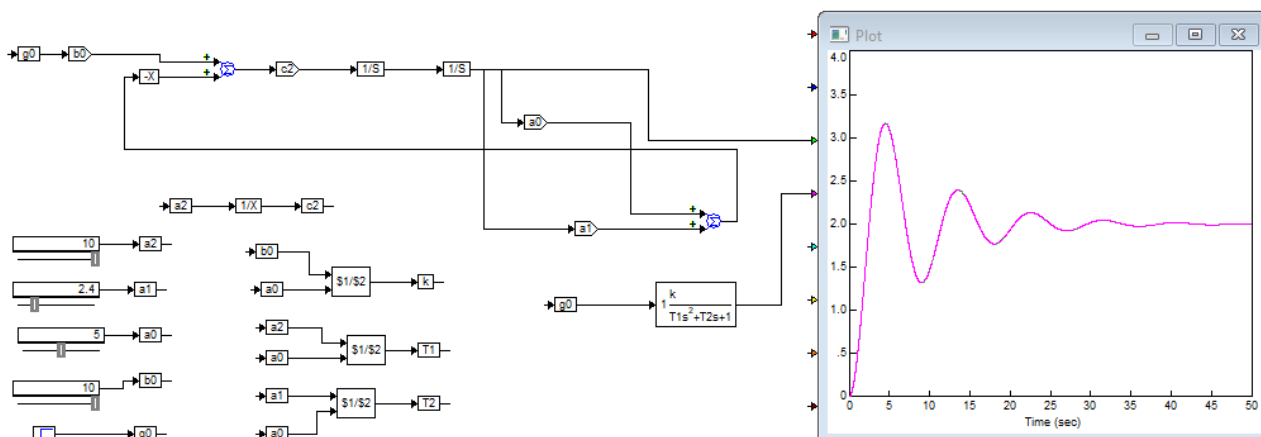


Рис. 3. Пример структурной модели системы второго порядка с $m = 0$

Примечание: составлено авторами на основании данных, полученных в исследовании.

Более универсальным является метод комбинирования производных [7], когда исследуется система «вход – внутреннее состояние – выход». Прежде всего, выполним приведение дифференциального уравнения (1) к виду, чтобы коэффициент при старшей производной выходного сигнала был равен 1, входной сигнал приведенной системы – это управляющее воздействие $u(t)$, выходной сигнал – $y(t)$:

$$\begin{aligned} \frac{d^n y(t)}{dt^n} + a_{n-1}' \frac{d^{n-1} y(t)}{dt^{n-1}} + \dots + a_1' \frac{dy(t)}{dt} + a_0' y(t) = \\ = b_m' \frac{d^m u(t)}{dt^m} + b_{m-1}' \frac{d^{m-1} u(t)}{dt^{m-1}} + \dots + b_1' \frac{du(t)}{dt} + b_0' u(t), \end{aligned} \quad (7)$$

где $a_{i-1}' = \frac{a_{i-1}}{a_n}$, $b_j' = \frac{b_j}{a_n}$ – параметры обобщенной системы n -го порядка, приведенной к стандартному виду. Здесь $i = 1, \dots, n$, $j = 0, \dots, m$, причем $m < n$.

В операторной форме уравнение (7) принимает вид:

$$\begin{aligned} p^n Y(p) + a_{n-1}' p^{n-1} Y(p) + \dots + a_1' p Y(p) + a_0' Y(p) = \\ = b_m' p^m U(p) + b_{m-1}' p^{m-1} U(p) + \dots + b_1' p U(p) + b_0' U(p) \end{aligned}$$

или:

$$K(p)Y(p) = L(p)U(p), \quad (8)$$

где p – комплексная переменная, используемая в преобразовании Лапласа, $X(p)$ и $Y(p)$ – изображения входного $x_{\text{вх}}(t)$ и выходного $x_{\text{вых}}(t)$ сигналов системы (рис. 1):

$$K(p) = p^n + \sum_{i=0}^{n-1} a_i' p^i \text{ и } L(p) = \sum_{i=0}^m b_i' p^i.$$

Выражение (8) можно записать в виде равенства дробей:

$$\frac{Y(p)}{L(p)} = \frac{U(p)}{K(p)}. \quad (9)$$

Введем переменную внутреннего состояния системы $x(t)$ (рис. 4) – такую, что ее изображение $X(p)$ определяется как:

$$X(p) = \frac{Y(p)}{L(p)} = \frac{U(p)}{K(p)}. \quad (10)$$

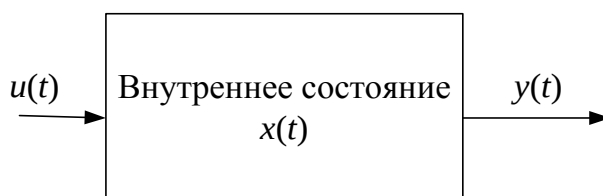


Рис. 4. Структура «вход – состояние – выход» для обобщенной системы автоматического управления

Примечание: составлено авторами на основании данных, полученных в исследовании.

Тогда для управляющего воздействия получается изображение:

$$U(p) = K(p) \cdot X(p)$$

и, соответственно, оригинал:

$$u(t) = \frac{d^n x(t)}{dt^n} + a_{n-1} \frac{d^{n-1} x(t)}{dt^{n-1}} + \dots + a_1 \frac{dx(t)}{dt} + a_0 x(t). \quad (11)$$

А для выходного сигнала системы получается изображение:

$$Y(p) = L(p) \cdot X(p)$$

и, соответственно, оригинал:

$$y(t) = b_m \frac{d^m x(t)}{dt^m} + b_{m-1} \frac{d^{m-1} x(t)}{dt^{m-1}} + \dots + b_1 \frac{dx(t)}{dt} + b_0 x(t). \quad (12)$$

Уравнения (11) и (12) являются основой для построения структурной модели приведенной обобщенной системы (рис. 5).

После ввода обычных для метода пространства состояний обозначений:

$$\begin{cases} x_1(t) = x(t) \\ x_2(t) = \frac{dx_1(t)}{dt} = \frac{dx(t)}{dt} \\ x_3(t) = \frac{dx_2(t)}{dt} = \frac{d^2 x(t)}{dt^2} \\ \dots \\ x_n(t) = \frac{dx_{n-1}(t)}{dt} = \frac{d^{n-1} x(t)}{dt^{n-1}} \end{cases}$$

получается система уравнений для производных первого порядка:

$$\begin{cases} \frac{dx_1(t)}{dt} = 0 \cdot x_1(t) + 1 \cdot x_2(t) + 0 \cdot x_3(t) + \dots + 0 \cdot x_{n-1}(t) + 0 \cdot x_n(t) + 0 \cdot u(t) \\ \frac{dx_2(t)}{dt} = 0 \cdot x_1(t) + 0 \cdot x_2(t) + 1 \cdot x_3(t) + \dots + 0 \cdot x_{n-1}(t) + 0 \cdot x_n(t) + 0 \cdot u(t) \\ \dots \\ \frac{dx_{n-1}(t)}{dt} = 0 \cdot x_1(t) + 0 \cdot x_2(t) + 0 \cdot x_3(t) + \dots + 0 \cdot x_{n-1}(t) + 1 \cdot x_n(t) + 0 \cdot u(t) \\ \frac{dx_n(t)}{dt} = -a_0 \cdot x_1(t) - a_1 \cdot x_2(t) - a_2 \cdot x_3(t) - \dots - a_{n-2} \cdot x_{n-1}(t) - a_{n-1} \cdot x_n(t) + 1 \cdot u(t) \\ y(t) = b_0 \cdot x_1(t) + b_1 \cdot x_2(t) + \dots + b_m \cdot x_{m+1}(t) + 0 \cdot x_{m+2}(t) + \dots + 0 \cdot x_n(t) + 0 \cdot u(t). \end{cases}$$

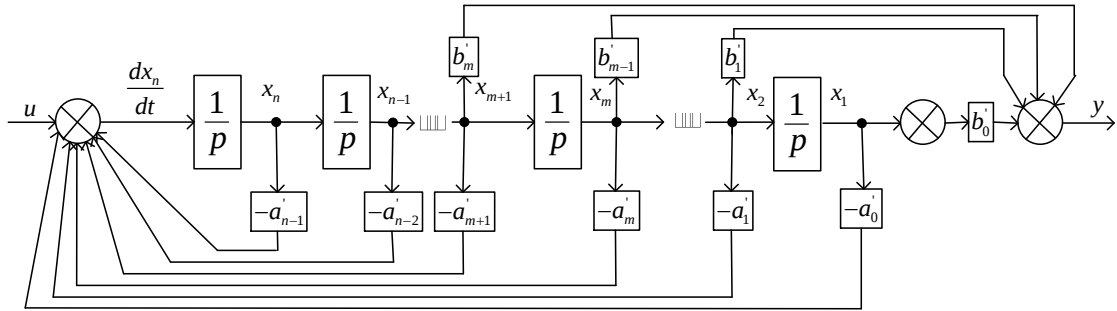


Рис. 5. Структурная модель обобщенной системы автоматического управления n -го порядка в форме «вход – внутреннее состояние – выход»

Примечание: составлено авторами на основании данных, полученных в исследовании.

В матричной форме она запишется в виде:

$$\frac{dX_{\text{сост}}(t)}{dt} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -a_0 & -a_1 & -a_2 & \dots & -a_{n-2} & -a_{n-1} \end{bmatrix} \cdot X_{\text{сост}}(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix} \cdot u(t) \quad (13)$$

$$Y(t) = [b_0 \ b_1 \ \dots \ b_m \ 0 \ \dots \ 0 \ 0] \cdot X_{\text{сост}}(t) + 0 \cdot u(t).$$

ной приведенной САУ n -го порядка в пространстве состояний [7] для нулевых начальных условий:

$$\begin{cases} \frac{dX_{\text{сост}}(t)}{dt} = A(t) \cdot X_{\text{сост}}(t) + B(t) \cdot u(t) \\ Y(t) = C(t) \cdot X_{\text{сост}}(t) + D(t) \cdot u(t) \end{cases} \quad (14)$$

Таким образом, получена стандартная форма записи уравнения динамики обобщен-

Здесь $\frac{dX_{\text{cocr}}(t)}{dt}$ – n -мерная матрица – столбец первых производных переменных состояний:

$$\frac{dX_{\text{cocr}}(t)}{dt} = \begin{bmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \\ \dots \\ \frac{dx_{n-1}(t)}{dt} \\ \frac{dx_n(t)}{dt} \end{bmatrix},$$

$X_{\text{cocr}}(t)$ – n -мерная матрица – столбец переменных состояний:

$$X_{\text{cocr}}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \dots \\ x_{n-1}(t) \\ x_n(t) \end{bmatrix},$$

$A(t)$, $B(t)$, $C(t)$, $D(t)$ – это матрицы пространства состояний обобщенной системы, параметры которой в общем случае зависят от времени:

$$A(t) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -a_0' & -a_1' & -a_2' & \dots & -a_{n-2}' & -a_{n-1}' \end{bmatrix}, B(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \\ 1 \end{bmatrix} \quad (15)$$

$$C(t) = [b_0' \ b_1' \ \dots \ b_m' \ 0 \ \dots \ 0 \ 0], D(t) = [0].$$

$A(t)$ – квадратная матрица n -го порядка, которая показывает порядок системы, параметры входящих в нее элементов и их взаимные связи. Для ее составления необходимо в n -й строке поставить параметры из левой части приведенного дифференциального уравнения. Заполнение всех остальных строк неизменно.

$B(t)$ – матрица – столбец n -го порядка, которая показывает связь управляющего воздействия с переменными состояниями. Последняя строка этой матрицы равна 1, остальные – нулю.

$C(t)$ – матрица – строка n -го порядка, формирующая выходные переменные из переменных состояний. Она состоит из коэффициентов правой части приведенного дифференциального уравнения (m параметров), дополненных нулями до размера матрицы n .

$D(t)$ – матрица обхода системы, представляющая собой нулевую матрицу размера 1×1 .

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В качестве численного эксперимента проведено исследование линейной системы третьего порядка ($n = 3$, $m = 1$), описываемой уравнением:

$$0,01 \frac{d^3 x_{\text{вых}}(t)}{dt^3} + 0,5 \frac{d^2 x_{\text{вых}}(t)}{dt^2} + \frac{dx_{\text{вых}}(t)}{dt} + x_{\text{вых}}(t) = 0,5 \frac{dx_{\text{вх}}(t)}{dt} + x_{\text{вх}}(t)$$

и, соответственно, передаточной функцией:

$$W(p) = \frac{0,5p + 1}{0,01p^3 + 0,5p^2 + p + 1}.$$

Приведенное уравнение этой системы:

$$\frac{d^3 x_{\text{вых}}(t)}{dt^3} + 50 \frac{d^2 x_{\text{вых}}(t)}{dt^2} + 100 \frac{dx_{\text{вых}}(t)}{dt} + 100 x_{\text{вых}}(t) = 50 \frac{dx_{\text{вх}}(t)}{dt} + 100 x_{\text{вх}}(t).$$

Матрицы для пространства состояний:

$$A(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -100 & -100 & -50 \end{bmatrix}, B(t) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C(t) = [100 \ 50 \ 0], D(t) = [0].$$

Результаты проверочного моделирования, проведенного по методике [3], и соответствующих m -файл приведены на рис. 6–11 в программах VisSim, MatLab/Simulink и SimInTech. Правильность расчетов подтверждается совпадением графиков переходных процессов, полученных при задании системы тремя способами:

- 1) через пространство состояний;
- 2) через передаточную функцию;
- 3) путем структурного моделирования.

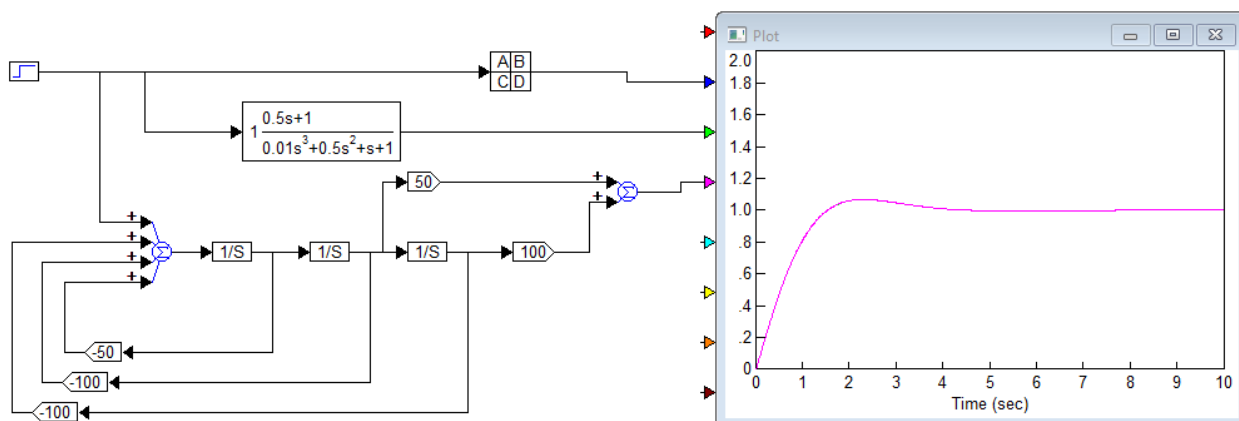


Рис. 6. Пример структурной модели системы третьего порядка с $m = 1$ в VisSim
 Примечание: составлено авторами на основании данных, полученных в исследовании.

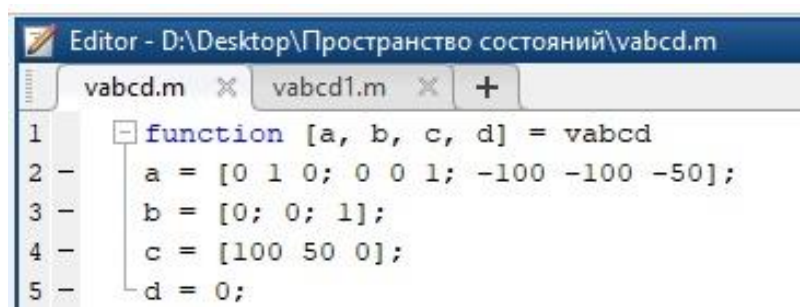


Рис. 7. Задание матриц пространства состояний в .m-файле
 Примечание: составлено авторами на основании данных, полученных в исследовании.

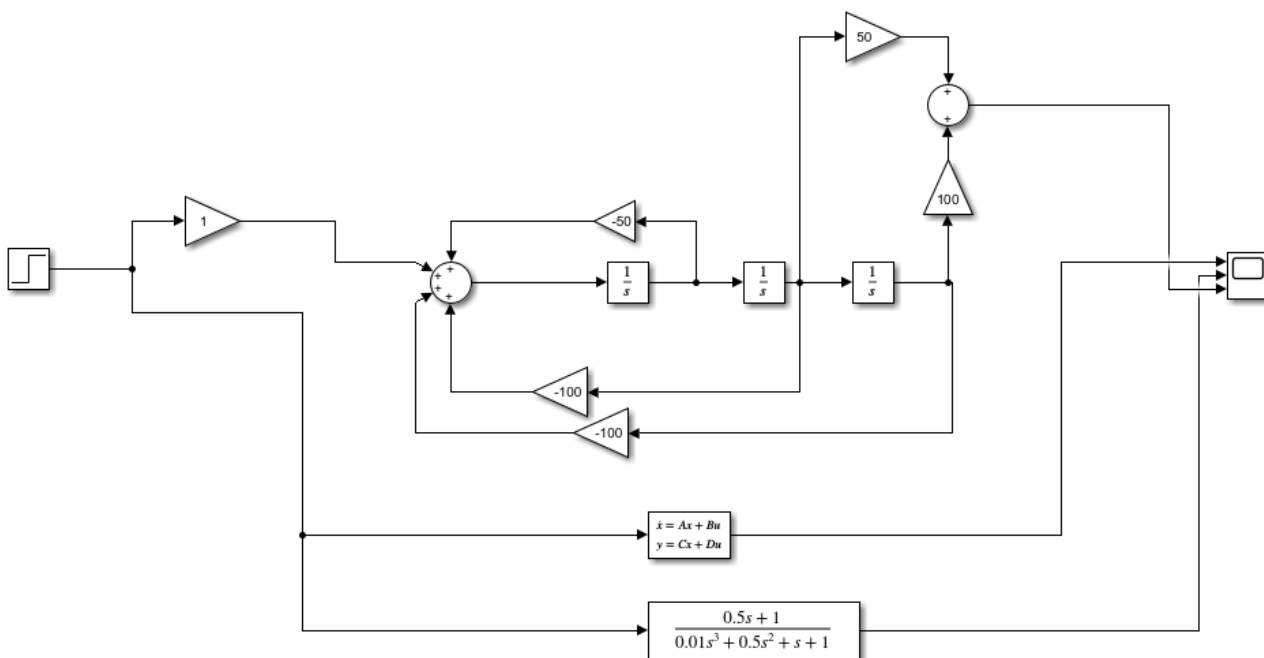


Рис. 8. Структурное моделирование рассматриваемой системы в Matlab/Simulink
 Примечание: составлено авторами на основании данных, полученных в исследовании.

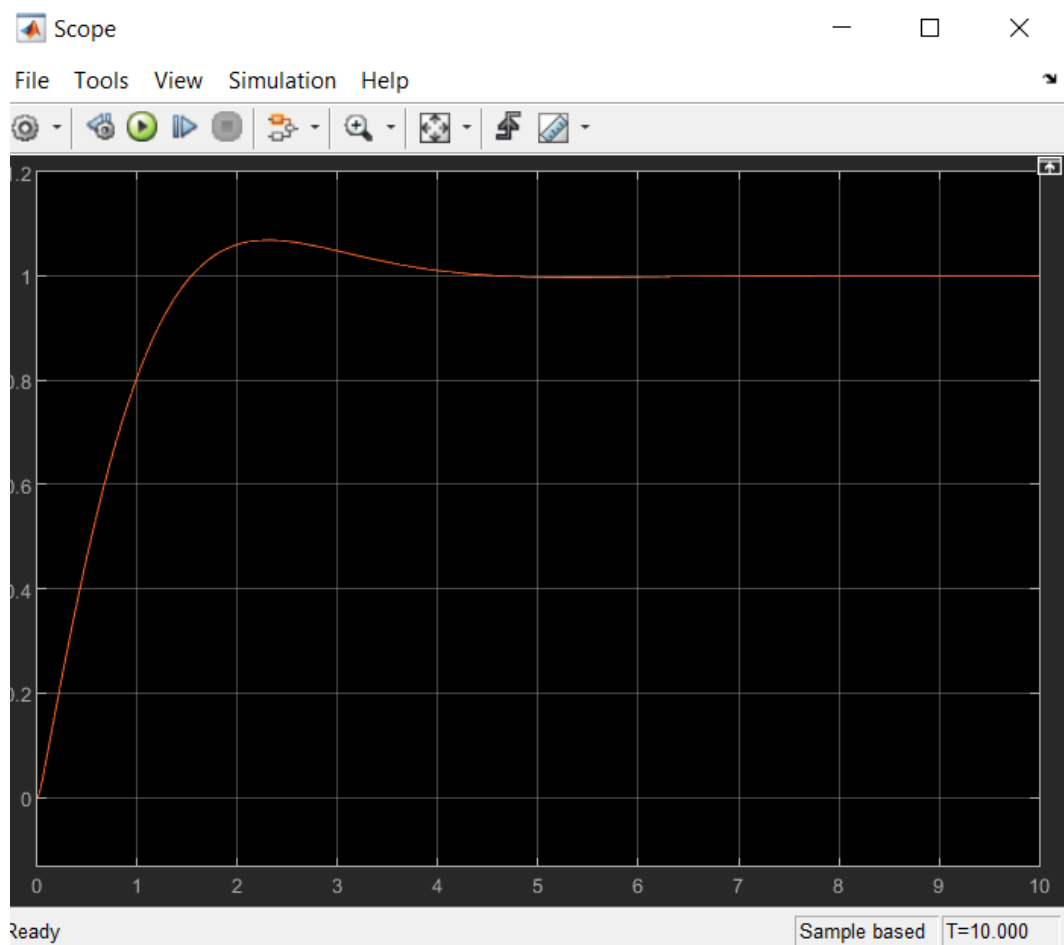


Рис. 9. Переходный процесс, полученный в результате структурного моделирования рассматриваемой системы в Matlab/Simulink

Примечание: составлено авторами на основании данных, полученных в исследовании.

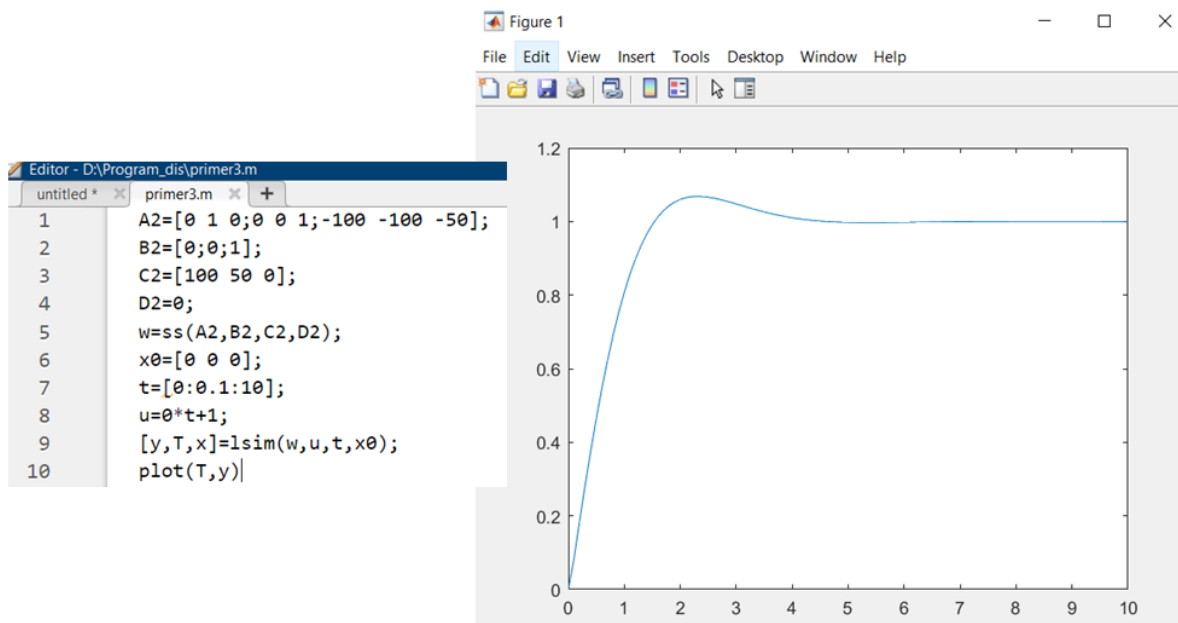
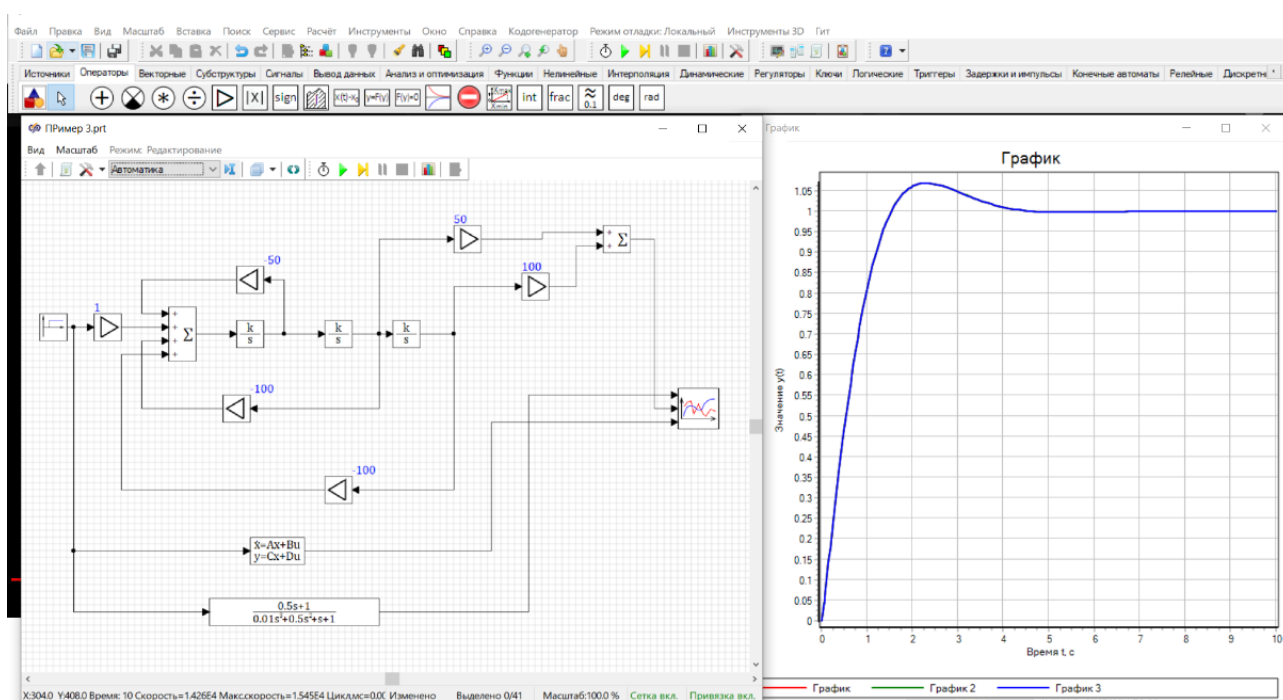


Рис. 10. Скрипт Matlab и результат расчетов для рассматриваемого примера

Примечание: составлено авторами на основании данных, полученных в исследовании.



**Рис. 11. Функциональная схема и результаты моделирования
для численного примера в SimInTech**

Примечание: составлено авторами на основании данных, полученных в исследовании.

ЗАКЛЮЧЕНИЕ

Показанная методика составления матриц пространства состояний применима для линейных и нелинейных SISO-систем, поскольку в таком виде можно записать уравнения динамики любой степени сложности [8], а осуществление преобразований над матрицами $A(t)$, $B(t)$, $C(t)$ и $D(t)$ возможно в любом

доступном пакете симуляции (VisSim, Matlab/Simulink, SimInTech и т. п.), что существенно облегчает процесс решения прямой задачи теории автоматического управления, заключающейся в определении реакции системы с известным математическим описанием.

Список источников

1. Бесекерский В. А., Попов Е. П. Теория систем автоматического управления. СПб. : Профессия, 2004. 752 с.
2. Калиева С. Т., Панченко В. Н., Иванов В. В. Рациональное распределение парка локомотивов и анализ их надежности в эксплуатации // Вестн. транспорта Поволжья. 2019. № 6 (78). С. 13–18.
3. Музылева И. В. Элементарная теория линейных систем в задачах и упражнениях. СПб. : Лань, 2022. 428 с. URL: <https://e.lanbook.com/book/-209948> (дата обращения: 28.10.2022).
4. Прошина Р. Д. Математическое моделирование технических систем в нормальной форме пространства состояний // Изв. Самар. науч. центра РАН. 2011. № 1–3. URL: <https://cyberleninka.ru/> (дата обращения: 07.12.2022).
5. Деруссо П. М., Рой Р. Дж., Клоуз Ч. М. Пространство состояний в теории управления: для инженеров. М. : Наука, 1970. 620 с.

References

1. Besekersky V. A., Popov E. P. Teoriia sistem avtomaticheskogo upravleniia. St. Petersburg : Professii, 2004. 752 p. (In Russian).
2. Kalieva S. T., Panchenko V. N., Ivanov V. V. Rational Distribution of Locomotive Fleet and Analysis of Its Operation Reliability // Vestn. transporta Povolzhia. 2019. No. 6 (78). P. 13–18. (In Russian).
3. Muzyleva I. V. Elementarnaia teoriia lineinykh sistem v zadachakh i uprazhneniiakh. St. Petersburg : Lan, 2022. 428 p. URL: <https://e.lanbook.com/book/209948> (accessed: 28.10.2022). (In Russian).
4. Proshina R. D. Mathematical Modeling of Technical Systems in Normal Form of State Space // Izvestia of Samara Scientific Center of the Russian Academy of Sciences. 2011. No. 1–3. URL: <https://cyberleninka.ru/> (accessed: 07.12.2022). (In Russian).
5. Derusso P. M., Roy R. J., Close C. M. State Variables for Engineers. Moscow : Nauka, 1970. 620 p. (In Russian).

6. Музылева И. В., Языкова Л. Н., Мещеряков В. Н. Метод контрольного моделирования для проверки передаточной функции сложноструктурированной системы // Вестник кибернетики. 2022. № 2 (46). С. 46–54. DOI 10.34822/1999-7604-2022-2-46-54.
7. Мейер А., Сю Д. Современная теория автоматического управления и ее применение / под ред. Ю. И. Толчеева. М. : Машиностроение, 1972. 544 с.
8. Андриевский Б. Р., Фрадков А. Л. Избранные главы теории автоматического управления с примерами на языке MATLAB. СПб. : Наука, 2000. 475 с.
6. Muzyleva I. V., Yazykova L. N., Meshcheryakov V. N. Method of Control Modeling for Testing Transfer Function of a Highly Structured System // Proceedings in Cybernetics. 2022. No. 2 (46). P. 46–54. DOI 10.34822/1999-7604-2022-2-46-54. (In Russian).
7. Meyer A. U., Hsu J. C. Modern Control Principles and Applications / Ed. Yu. I. Tolcheeva. Moscow : Mashinostroenie, 1972. 544 p. (In Russian).
8. Andrievsky B. R., Fradkov A. L. Izbrannye glavy teorii avtomaticheskogo upravleniia s primerami na iazyke MATLAB. St. Petersburg : Nauka, 2000. 475 p. (In Russian).

Информация об авторах

И. В. Музылева – кандидат технических наук, доцент.

Л. Н. Языкова – старший преподаватель.

В. Н. Мещеряков – доктор технических наук, профессор.

Information about the authors

I. V. Muzyleva – Candidate of Sciences (Engineering), Associate Professor.

L. N. Yazykova – Senior Lecturer.

V. N. Meshcheryakov – Doctor of Sciences (Engineering), Professor.

ФИЗИКО-МАТЕМАТИЧЕСКИЕ НАУКИ

Научная статья
УДК 519.654:519.2:519.862.6
doi: 10.34822/1999-7604-2022-4-69-76

ОЦЕНКА МЕТОДОМ НАИМЕНЬШИХ КВАДРАТОВ ПРОСТЕЙШИХ НЕЭЛЕМЕНТАРНЫХ ЛИНЕЙНЫХ РЕГРЕССИЙ С ЛИНЕЙНЫМ АРГУМЕНТОМ В БИНАРНОЙ ОПЕРАЦИИ

Михаил Павлович Базилевский

Иркутский государственный университет путей сообщения, Иркутск, Россия
mik2178@yandex.ru, <https://orcid.org/0000-0002-3253-5697>

Аннотация. Простейшая неэлементарная линейная регрессионная модель содержит две объясняющие переменные, преобразованные с помощью бинарной операции минимум или максимум, и один из аргументов бинарной операции в таких моделях содержит только угловой коэффициент. Проведено исследование неэлементарных линейных регрессий, в которых аргумент бинарной операции содержит как единичный угловой коэффициент, так и свободный член. На основе алгоритма приближенной оценки методом наименьших квадратов неэлементарных линейных регрессий разработан алгоритм оценки неэлементарной линейной регрессии, содержащей в аргументе бинарной операции и угловой коэффициент, и свободный член. Предложенные алгоритмы были реализованы на языке программирования *hansl* пакета *gretl* в виде программы, с помощью которой решена задача моделирования железнодорожных грузовых перевозок в Тюменской области. Построены традиционная линейная регрессия и три разновидности неэлементарной линейной регрессии: с угловым коэффициентом в аргументе бинарной операции, с единичным угловым коэффициентом и свободным членом, с угловым коэффициентом и свободным членом. Предложенные в работе неэлементарные модели со свободным членом в бинарной операции на практике оказались лучше, чем их известный аналог.

Ключевые слова: неэлементарная линейная регрессия, метод наименьших квадратов, угловой коэффициент, свободный член, бинарная операция, интерпретация

Для цитирования: Базилевский М. П. Оценка методом наименьших квадратов простейших неэлементарных линейных регрессий с линейным аргументом в бинарной операции // Вестник кибернетики. 2022. № 4 (48). С. 69–76. DOI 10.34822/1999-7604-2022-4-69-76.

Original article

ORDINARY LEAST SQUARES ESTIMATION OF SIMPLE NON-ELEMENTARY LINEAR REGRESSIONS WITH A LINEAR ARGUMENT IN A BINARY OPERATION

Mikhail P. Bazilevsky

Irkutsk State Transport University, Irkutsk, Russia
mik2178@yandex.ru, <https://orcid.org/0000-0002-3253-5697>

Abstract. The simple non-elementary linear regression model contains two explanatory variables transformed by a minimum or maximum binary operation. One of the arguments of a binary operation in such models contains only the slope. Non-elementary linear regressions, in which the argument of a binary operation contains both the unit slope and intercept, are studied. Based on the algorithm of approximate estimation by ordinary least squares for non-elementary linear regressions, an algorithm for estimating a non-elementary linear regression, in which the argument of a binary operation contains the slope and intercept,

is developed. The proposed algorithms were implemented as a program that solves the modeling problem for railway freight traffic in Tyumen Oblast using hansl, a scripting language from the gretl package. A classical linear regression and three options of non-elementary linear regression (with the slope in the argument of a binary operation, the unit slope and intercept, and the slope and intercept) were constructed. The proposed non-elementary models with the intercept in a binary operation were found to be more efficient than their well-known alternatives.

Keywords: non-elementary linear regression, ordinary least squares method, slope, intercept, binary operation, interpretation

For citation: Bazilesky M. P. Ordinary Least Squares Estimation of Simple Non-Elementary Linear Regressions with a Linear Argument in a Binary Operation // Proceedings in Cybernetics. 2022. No. 4 (48). P. 69–76. DOI 10.34822/1999-7604-2022-4-69-76.

ВВЕДЕНИЕ

Для проведения регрессионного анализа [1, 2] разработано значительное количество различных известных структурных спецификаций регрессионных моделей [3], применяемых при решении прикладных задач анализа данных. Одновременно идет поиск новых регрессионных зависимостей, более совершенных форм связи между переменными, и актуальным научным направлением сегодня является построение интерпретируемых моделей машинного обучения [4, 5].

Методы построения нечетких регрессионных моделей активно развиваются, например: в [6] представлен новый подход для оценки параметров модели нечеткой регрессии; в [7] рассмотрена нечеткая регрессия с интервальными значениями и предложена новая унифицированная методология работы с моделями нечеткой регрессии, в [8] представлен нечеткий линейный метод наименьших квадратов (МНК); в [9] исследуются вопросы оценивания нечетких регрессионных моделей с помощью метода наименьших модулей (МНМ).

Широкое распространение получили так называемые кусочно-линейные регрессионные модели [10], оценка которых осуществляется с помощью МНМ, например, для моделирования объемов перевозок пассажиров железнодорожным транспортом [11]. В [12] предложена кусочно-линейная авторегрессионная модель произвольного порядка для расчета обеспеченности жильем в Иркутской области. В [13] исследована кусочно-линейная регрессия с интервальной неопределенностью для зависимой переменной.

В [14] рассмотрены простейшие неэлементарные линейные регрессии (НЛР), состоя-

щие из двух объясняющих переменных, оцениваемых с помощью МНК, а в [15] понятие НЛР было обобщено на случай многих переменных. В общем случае, НЛР содержит в себе не только объясняющие переменные, но и все возможные комбинации их пар, преобразованные с помощью бинарных операций минимум и максимум. Тем самым НЛР есть обобщение линейных регрессий. Для выбора наиболее информативных регрессоров в НЛР было предложено две стратегии, требующие реализации переборных процедур [16], а в [17] задача выбора оптимальной структуры НЛР сведена к задаче частично-булевого линейного программирования и продемонстрированы высокие интерпретационные способности НЛР. В работах [14–17] в НЛР один из аргументов бинарных операций содержит только угловой коэффициент при объясняющей переменной.

Цель данной работы заключается в разработке алгоритма оценки МНК обобщенных простейших НЛР, в которых один из аргументов бинарной операции представляет собой линейную функцию, т. е. содержит как угловой коэффициент, так и свободный член.

МАТЕРИАЛЫ И МЕТОДЫ

В работе [14] рассмотрена простейшая НЛР вида:

$$y_i = \alpha_0 + \alpha_1 \min \{x_{i1}, kx_{i2}\} + \varepsilon_i, \quad i = \overline{1, n}, \quad (1)$$

где n – объем выборки; $y_i, i = \overline{1, n}$ – значения зависимой (объясняемой) переменной y ; $x_{i1} \geq 0, x_{i2} \geq 0, i = \overline{1, n}$ – значения независимых (объясняющих) переменных x_1 и x_2 ; α_0 ,

α_1 , k – неизвестные параметры; ε_i , $i = \overline{1, n}$ – ошибки аппроксимации.

Модель (1) нелинейна по параметрам. Но если придать параметру k определенное значение, то она становится линейной, и оценки параметров α_0 и α_1 можно легко идентифицировать, например, по известным формулам для МНК.

В [1] установлена область возможных значений параметра k . Для этого сначала определяется область его значений, при которых НЛР (1) трансформируется в обычную парную линейную регрессию y от x_1 . В этом случае нужно решить систему линейных неравенств:

$$\begin{cases} x_{11} \leq kx_{12}, \\ x_{21} \leq kx_{22}, \\ \dots \\ x_{n1} \leq kx_{n2}. \end{cases}$$

Решение этой системы можно записать в виде:

$$k \geq k_{\max},$$

$$\text{где } k_{\max} = \max \left\{ \frac{x_{11}}{x_{12}}, \frac{x_{21}}{x_{22}}, \dots, \frac{x_{n1}}{x_{n2}} \right\}.$$

Аналогично для определения области значений параметра k , при которых НЛР (1) трансформируется в парную линейную регрессию y от x_2 , решается система неравенств:

$$\begin{cases} x_{11} \geq kx_{12}, \\ x_{21} \geq kx_{22}, \\ \dots \\ x_{n1} \geq kx_{n2}. \end{cases}$$

Она имеет решение:

$$k \leq k_{\min},$$

$$\text{где } k_{\min} = \min \left\{ \frac{x_{11}}{x_{12}}, \frac{x_{21}}{x_{22}}, \dots, \frac{x_{n1}}{x_{n2}} \right\}.$$

Тогда оптимальная оценка параметра k в НЛР (1) принадлежит отрезку $[k_{\min}, k_{\max}]$.

Разбивая этот отрезок достаточно большим числом точек, вычисляя в каждой точке МНК-оценки параметров α_0 , α_1 и выбирая уравнение, для которого сумма квадратов ошибок минимальна, можно получить НЛР, МНК-оценки которой практически не отличаются от оптимальных.

Модель (1) можно назвать НЛР только с угловым коэффициентом k в бинарной операции. Стоит отметить, что вместо бинарной операции $\min$ в НЛР (1) можно использовать бинарную операцию $\max$.

Введем в рассмотрение НЛР со свободным членом b и единичным угловым коэффициентом в бинарной операции:

$$y_i = \alpha_0 + \alpha_1 \min \{x_{i1}, x_{i2} + b\} + \varepsilon_i, \quad i = \overline{1, n}. \quad (2)$$

МНК-оценки неизвестных параметров НЛР (2) можно найти точно по такому же алгоритму, что и для НЛР (1). Для этого сначала определим область возможных значений параметра b .

Если НЛР (2) трансформируется в обычную парную регрессию y от x_1 , то справедливы условия:

$$\begin{cases} x_{11} \leq x_{12} + b, \\ x_{21} \leq x_{22} + b, \\ \dots \\ x_{n1} \leq x_{n2} + b. \end{cases}$$

Решением такой системы линейных неравенств будет промежуток:

$$b \geq b_{\max},$$

$$\text{где } b_{\max} = \max \{x_{11} - x_{12}, x_{21} - x_{22}, \dots, x_{n1} - x_{n2}\}.$$

Если же НЛР (2) трансформируется в парную регрессию y от x_2 , то выполняются условия:

$$\begin{cases} x_{11} \geq x_{12} + b, \\ x_{21} \geq x_{22} + b, \\ \dots \\ x_{n1} \geq x_{n2} + b, \end{cases}$$

откуда:

$$b \leq b_{\min},$$

где $b_{\min} = \min \{x_{11} - x_{12}, x_{21} - x_{22}, \dots, x_{n1} - x_{n2}\}$.

Из этого следует, что оптимальная оценка параметра b в НЛР (2) принадлежит отрезку $[b_{\min}, b_{\max}]$. Тогда для нахождения близких к оптимальным МНК-оценкам параметров НЛР (2) можно воспользоваться следующим алгоритмом:

1. Разбить отрезок $[b_{\min}, b_{\max}]$ максимальным количеством точек.

2. В каждой точке и на концах отрезка с помощью МНК определить оценки параметров α_0 и α_1 НЛР (2).

3. Выбрать точку, в которой сумма квадратов ошибок НЛР (2) минимальна.

Теперь введем в рассмотрение НЛР со свободным членом b и угловым коэффициентом k в бинарной операции:

$$y_i = \alpha_0 + \alpha_1 \min \{x_{i1}, kx_{i2} + b\} + \varepsilon_i, \quad i = \overline{1, n}. \quad (3)$$

Понятно, что если $b = 0$, то НЛР (3) вырождается в НЛР (1), а если $k = 1$, то в НЛР (2). К сожалению, из-за того что степень нелинейности НЛР (3) выше, чем у моделей (1) и (2), определить аналитически в какую область попадают оптимальные оценки ее параметров k и b не представляется возможным. Однако можно сформулировать следующий алгоритм МНК-оценивания НЛР (3), гарантирующий получение модели с величиной суммы квадратов ошибок, не большей, а зачастую меньшей, чем у НЛР (1) и (2):

1. Выбрать область изменения параметра k так, чтобы она включала в себя и промежуток $[k_{\min}, k_{\max}]$, и единицу. Разбить получен-

ный отрезок $[k_1, k_2]$ максимальным количеством точек.

2. В каждой s -й точке и на концах отрезка $[k_1, k_2]$ определить границы возможных значений параметра b по формулам $b_{s, \min} = \min \{x_{11} - k_s x_{12}, x_{21} - k_s x_{22}, \dots, x_{n1} - k_s x_{n2}\}$ и $b_{s, \max} = \max \{x_{11} - k_s x_{12}, x_{21} - k_s x_{22}, \dots, x_{n1} - k_s x_{n2}\}$. В результате для каждой s -й точки отрезка $[k_{\min}, k_{\max}]$ будет сформирован отрезок $[b_{s, \min}, b_{s, \max}]$.

3. Разбить s -й отрезок $[b_{s, \min}, b_{s, \max}]$ максимальным количеством точек.

4. В каждой точке и на концах отрезков $[b_{s, \min}, b_{s, \max}]$ с помощью МНК определить оценки параметров α_0 и α_1 НЛР (3).

5. Выбрать оценки параметров k и b , при которых сумма квадратов ошибок НЛР (3) минимальна.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Предложенные в работе алгоритмы МНК-оценивания НЛР были реализованы на языке программирования `hansl` эконометрического пакета `gretl` в виде программы для решения задачи моделирования железнодорожных грузовых перевозок в Тюменской области. Для этого были использованы ежегодные статистические данные (источник – <https://rosstat.gov.ru/>) за период с 2011 по 2020 гг. по следующим переменным:

y – отправление грузов железнодорожным транспортом общего пользования (млн т);

x_1 – производство электроэнергии (млрд кВт*ч);

x_2 – продукция сельского хозяйства (млн руб.).

Данные приведены в табл. 1.

Таблица 1

Статистические данные

Год	y	x_1	x_2
2011	24,5	101,0	58 660,2
2012	26,4	102,7	56 125,2
2013	30,5	108,1	58 853,2
2014	34,2	109,8	67 110,7
2015	40,6	109,2	70 657

Окончание табл. 1

Год	y	x ₁	x ₂
2016	40,2	110,6	74 486,2
2017	38,4	108,2	76 715,6
2018	37,7	108,79	77 793
2019	37,7	110,63	8 3037
2020	38,2	101,2	87 514

Примечание: составлено автором.

Оцененная по этим данным с помощью МНК традиционная модель множественной линейной регрессии имеет вид:

$$\tilde{y} = -63,8402 + 0,6724x_1 + 0,0003758x_2. \quad (4)$$

Коэффициент детерминации R^2 модели (4) составил 0,848594, что подтверждает ее адекватность.

Затем с помощью МНК численно оценивалась НЛР (1). Область возможных значений параметра k составила $[0,001156, 0,001837]$. Число разбиений этого отрезка задавалось равным 100. График зависимости коэффициента детерминации R^2 оцениваемой модели от величины k представлен на рис. 1.

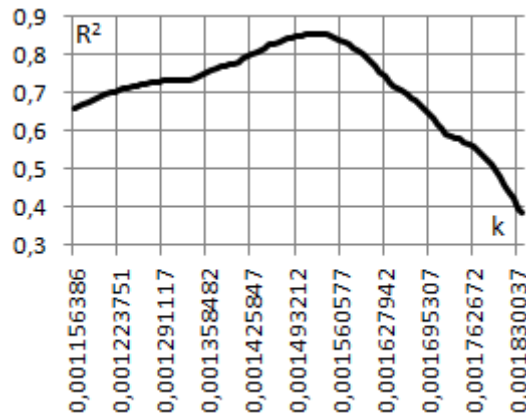


Рис. 1. График зависимости R^2 от k для неэлементарной линейной регрессии (1)

Примечание: составлено автором.

Как видно по рис. 1, максимальное значение 0,854667 коэффициент детерминации R^2 достигает только в одной точке $k = 0,001527$, в которой НЛР (1) имеет вид:

$$\tilde{y} = -21,4316 + 0,5544 \min\{x_1, 0,001527x_2\}. \quad (5)$$

Таким образом, по величине R^2 НЛР (5) оказалась незначительно лучше, чем линейная регрессия (4).

В кусочно-заданной форме НЛР (5) имеет вид:

$$\tilde{y} = \begin{cases} -21,4316 + 0,5544x_1, & \text{при } \frac{x_1}{x_2} \leq 0,001527, \\ -21,4316 + 0,000846x_2, & \text{при } \frac{x_1}{x_2} > 0,001527. \end{cases}$$

Из этого следует, что при переключении функций в НЛР (5) угловые коэффициенты при переменных различны, а свободный $(-21,4316)$ постоянен. При этом переключение осуществляется в зависимости от величины относительного показателя $\frac{x_1}{x_2}$.

После чего с помощью МНК численно оценивалась НЛР (2). Область возможных значений параметра b составила $[-87412,8, -56022,5]$. Число разбиений этого отрезка задавалось равным 100. График зависимости коэффициента детерминации R^2 оцениваемой модели от величины b представлен на рис. 2.

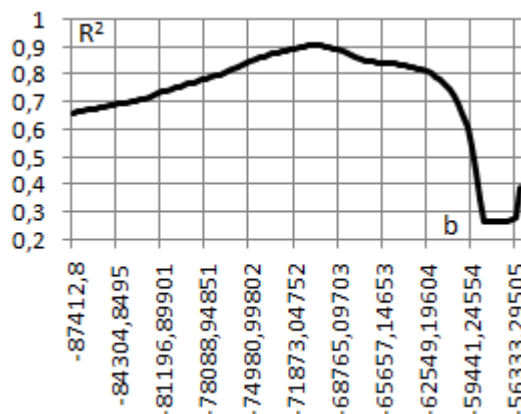


Рис. 2. График зависимости R^2 от b для неэлементарной линейной регрессии (2)

Примечание: составлено автором.

Как видно по рис. 2, максимальное значение 0,906492 коэффициент детерминации R^2 достигает только в одной точке $b = -70629,87$, в которой НЛР (2) имеет вид:

$$\tilde{y} = 38,5499 + 0,0009 \min \{x_1, x_2 - 70629,87\}. \quad (6)$$

В результате по величине R^2 НЛР (6) оказалась гораздо лучше как линейной регрессии (4), так и НЛР (5).

В кусочно-заданной форме НЛР (6) имеет вид:

$$\tilde{y} = \begin{cases} 38,5499 + 0,0009x_1, & \text{при } x_2 - x_1 \geq 70629,87, \\ -25,017 + 0,0009x_2, & \text{при } x_2 - x_1 < 70629,87. \end{cases}$$

Из этого следует, что при переключении функций в НЛР (6) свободные члены различны, а угловой коэффициент при переменных (0,0009) постоянен. При этом переключение осуществляется в зависимости от величины абсолютного показателя $x_2 - x_1$.

Далее с помощью МНК численно оценивалась НЛР (3). Область возможных значений параметра k была выбрана $[0,001156, 2]$. Число разбиений отрезков и для параметра k , и для b задавалось равным 1 000. В результате работы программы было установлено, что максимальное значение 0,90825 коэффициент детерминации R^2 достигает при $k = 0,011768$ и $b = -722,3053$. В этой точке НЛР (3) имеет вид:

$$\tilde{y} = 30,25 + 0,077467 \min \{x_1, 0,011768x_2 - 722,3053\}. \quad (7)$$

По величине R^2 НЛР (7) оказалась лучшей из всех оцененных моделей.

В кусочно-заданной форме НЛР (7) имеет вид:

$$\tilde{y} = \begin{cases} 30,25 + 0,077467x_1, & \text{при } x_1 \leq 0,011768x_2 - 722,3053, \\ -25,7048 + 0,000911x_2, & \text{при } x_1 > 0,011768x_2 - 722,3053. \end{cases}$$

Как видно, при переключении функций в НЛР (7) различны как свободные члены, так и угловые коэффициенты.

Стоит отметить, что при использовании бинарной операции максимум оцененные НЛР оказались хуже, чем модели (5)–(7).

ЗАКЛЮЧЕНИЕ

В работе рассмотрены простейшие неэлементарные линейные регрессии с линейным аргументом в бинарной операции. Предложен алгоритм МНК-оценивания для спецификации с единичным угловым коэффициентом и свободным членом, и с произвольным угловым коэффициентом и свободным членом. Разработана реализующая эти алгоритмы программа, с помощью которой решена задача моделирования железнодорожных грузоперевозок в Тюменской области. Предложенные в работе модели на практике оказа-

лись лучше по качеству, чем линейная регрессия и неэлементарная линейная регрессия только с угловым коэффициентом в бинарной операции.

Предложенный алгоритм оценивания неэлементарной линейной регрессии со свободным членом и единичным угловым коэффициентом в бинарной операции гарантирует близость МНК-оценок к оптимальным, а алгоритм для неэлементарных линейных регрес-

сий с линейным аргументом в бинарной операции – нет. Однако полученная с его помощью модель гарантировано будет не хуже, чем известные на сегодня неэлементарные зависимости. Поэтому весьма перспективным направлением становится интеграция рассмотренных в данной работе простейших конструкций в обобщенную неэлементарную линейную регрессию [17].

Список источников

1. Brook R. J., Arnold G. C. Applied Regression Analysis and Experimental Design. Boca Raton, FL : CRC Press, 2018. 256 p.
2. Montgomery D. C., Peck E. A., Vining G. G. Introduction to Linear Regression Analysis. Hoboken, NJ : John Wiley & Sons, 2021. 704 p.
3. Клейнер Г. Б. Производственные функции: теория, методы, применение. М. : Финансы и статистика, 1986. 240 с.
4. Molnar C. Interpretable Machine Learning. 2020. 312 p.
5. Doshi-Velez F., Kim B. Towards a Rigorous Science of Interpretable Machine Learning // arXiv. 2017. URL: <https://arxiv.org/pdf/1702.08608v2.pdf> (дата обращения: 15.10.2022).
6. Khammar A. H., Arefi M., Akbari M. G. A General Approach to Fuzzy Regression Models Based on Different Loss Functions // Soft Comput. 2021. Vol. 25, Is. 2. P. 835–849.
7. Boukezzoula R., Coquin D. Interval-Valued Fuzzy Regression: Philosophical and Methodological Issues // Appl Soft Comput. 2021. Vol. 103. P. 107145.
8. Hose D., Hanss M. Fuzzy Linear Least Squares for the Identification of Possibilistic Regression Models // Fuzzy Sets and Systems. 2019. Vol. 367. P. 82–95.
9. Chen L.-H., Nien S.-H. Mathematical Programming Approach to Formulate Intuitionistic Fuzzy Regression Model Based on Least Absolute Deviations // Fuzzy Optim Decis Making. 2020. Vol. 19, Is. 2. P. 191–210.
10. Носков С. И. Технология моделирования объектов с нестабильным функционированием и неопределенностью в данных. Иркутск : Облформпечать, 1996. 320 с.
11. Носков С. И., Хоняков А. А. Кусочно-линейные регрессионные модели объемов перевозок пассажиров железнодорожным транспортом // Модели, системы, сети в экономике, технике, природе и обществе. 2021. № 4. С. 80–89.
12. Носков С. И. Построение кусочно-линейной авторегрессионной модели произвольного порядка // Вестн. Югорск. гос. ун-та. 2022. № 2. С. 89–94.

References

1. Brook R. J., Arnold G. C. Applied Regression Analysis and Experimental Design. Boca Raton, FL : CRC Press, 2018. 256 p.
2. Montgomery D. C., Peck E. A., Vining G. G. Introduction to Linear Regression Analysis. Hoboken, NJ : John Wiley & Sons, 2021. 704 p.
3. Kleiner G. B. Proizvodstvennye funktsii: Teoriia, metody, primeneniye. Moscow : Finansy i statistika, 1986. 240 p. (In Russian).
4. Molnar C. Interpretable Machine Learning. 2020. 312 p.
5. Doshi-Velez F., Kim B. Towards a Rigorous Science of Interpretable Machine Learning // arXiv. 2017. URL: <https://arxiv.org/pdf/1702.08608v2.pdf> (accessed: 15.10.2022).
6. Khammar A. H., Arefi M., Akbari M. G. A General Approach to Fuzzy Regression Models Based on Different Loss Functions // Soft Comput. 2021. Vol. 25, Is. 2. P. 835–849.
7. Boukezzoula R., Coquin D. Interval-Valued Fuzzy Regression: Philosophical and Methodological Issues // Appl Soft Comput. 2021. Vol. 103. P. 107145.
8. Hose D., Hanss M. Fuzzy Linear Least Squares for the Identification of Possibilistic Regression Models // Fuzzy Sets and Systems. 2019. Vol. 367. P. 82–95.
9. Chen L.-H., Nien S.-H. Mathematical Programming Approach to Formulate Intuitionistic Fuzzy Regression Model Based on Least Absolute Deviations // Fuzzy Optim Decis Making. 2020. Vol. 19, Is. 2. P. 191–210.
10. Noskov S. I. Tekhnologiya modelirovaniia obektov s nestabilnym funktsionirovaniem i neopredelennosti v dannykh. Irkutsk : Oblinformpechat, 1996. 320 p. (In Russian).
11. Noskov S. I., Khonyakov A. A. Piecewise Linear Regression Models of Passenger Transportation Volumes by Railway // Models, Systems, Networks in Economics, Technology, Nature and Society. 2021. No. 4. P. 80–89. (In Russian).
12. Noskov S. I. Construction of a Piece-Linear Autoregression Model of an Arbitrary Order // Yugra State University Bulletin. 2022. No. 2. P. 89–94. (In Russian).

13. Носков С. И. Построение кусочно-линейной регрессии с интервальной неопределенностью в данных для зависимой переменной // Вестник кибернетики. 2022. № 2. С. 61–65.
14. Базилевский М. П. МНК-оценивание параметров специфицированных на основе функций Леонтьева двухфакторных моделей регрессии // Юж.-Сиб. науч. вестн. 2019. № 2. С. 66–70.
15. Базилевский М. П. Оценивание линейно-неэлементарных регрессионных моделей с помощью метода наименьших квадратов // Моделирование, оптимизация и информационные технологии. 2020. Т. 8, № 4. DOI 10.26102/2310-6018/2020.31.4.026.
16. Базилевский М. П. Отбор информативных операций при построении линейно-неэлементарных регрессионных моделей // International Journal of Open Information Technologies. 2021. Т. 9, № 5. С. 30–35.
17. Базилевский М. П. Метод построения неэлементарных линейных регрессий на основе аппарата математического программирования // Проблемы управления. 2022. № 4. С. 3–14.
13. Noskov S. I. Constructing a Data-Driven Piecewise Linear Regression with Interval Uncertainty for the Dependent Variable // Proceedings in Cybernetics. 2022. No. 2. P. 61–65. (In Russian).
14. Bazilevsky M. P. OLS-Estimation of Two-Factor Regression Models Specified on Leontiev Functions // South-Siberian Scientific Bulletin. 2019. No. 2. P. 66–70. (In Russian).
15. Bazilevsky M. P. Estimation Linear Non-Elementary Regression Models Using Ordinary Least Squares // Modeling, Optimization and Information Technology. 2020. Vol. 8, No. 4. DOI 10.26102/2310-6018/2020.31.4.026. (In Russian).
16. Bazilevsky M. P. Election of Informative Operations in the Construction of Linear Non-Elementary Regression Models // International Journal of Open Information Technologies. 2021. Vol. 9, No. 5. P. 30–35. (In Russian).
17. Bazilevsky M. P. A Method for Constructing Non-elementary Linear Regressions Based on Mathematical Programming // Control Sciences. 2022. No. 4. P. 3–14. (In Russian).

Информация об авторе

М. П. Базилевский – кандидат технических наук, доцент.

Information about the author

M. P. Bazilevsky – Candidate of Sciences (Engineering), Associate Professor.

Научная статья

УДК 519.6+004.43

doi: 10.34822/1999-7604-2022-4-77-90

ОБЗОР МЕТОДОВ И ОЦЕНКА СЛОЖНОСТИ АЛГОРИТМОВ ОПЕРАЦИЙ СРАВНЕНИЯ В МОДУЛЯРНОЙ АРИФМЕТИКЕ И ПЕРЕВОДА ИЗ МОДУЛЯРНОЙ СИСТЕМЫ В ПОЗИЦИОННУЮ СИСТЕМУ СЧИСЛЕНИЯ

Наталья Сергеевна Золотарева

Сургутский государственный университет, Сургут, Россия

zolotareva_ns@surgu.ru, <https://orcid.org/0000-0001-9751-4232>

Аннотация. Изложено математическое описание методов выполнения немодульных операций в модулярной арифметике: перевод из модулярной системы счисления в позиционную систему счисления и операции сравнения в модулярной системе счисления. Разработана программа на языке Python, моделирующая выполнение алгоритмов на электронно-вычислительной машине. Приведены примеры и представлены результаты работы алгоритмов. Выполнены оценки сложности алгоритмов для их сравнения и выявления оптимальных.

Ключевые слова: модулярная арифметика, модульные операции, немодульные операции, сравнение чисел, Китайская теорема об остатках, ранг числа, сложность алгоритмов, математическое моделирование

Для цитирования: Золотарева Н. С. Обзор методов и оценка сложности алгоритмов операций сравнения в модулярной арифметике и перевода из модулярной системы в позиционную систему счисления // Вестник кибернетики. 2022. № 4 (48). С. 77–90. DOI 10.34822/1999-7604-2022-4-77-90.

Original article

METHODS REVIEW AND COMPLEXITY ESTIMATION OF THE ALGORITHMS FOR COMPARISON OPERATIONS IN MODULAR ARITHMETIC AND TRANSFER OPERATIONS FROM A MODULAR NUMBER SYSTEM TO A POSITIONAL NUMBER SYSTEM

Natalya S. Zolotareva

Surgut State University, Surgut, Russia

zolotareva_ns@surgu.ru, <https://orcid.org/0000-0001-9751-4232>

Abstract. The article presents a mathematical description of methods for performing non-modular operations in modular arithmetic, such as transfer operations from a modular number system to a positional number system and comparison operations in a modular number system. A program for simulating algorithms on a computer was developed in Python. The study provides examples and results of the algorithms' performance. The algorithms' complexity is estimated in order to compare them and determine the most effective one.

Keywords: modular arithmetic, modular operations, non-modular operations, number comparison, Chinese remainder theorem, number rank, complexity of algorithms, mathematical modeling

For citation: Zolotareva N. S. Methods Review and Complexity Estimation of the Algorithms for Comparison Operations in Modular Arithmetic and Transfer Operations from a Modular Number System to a Positional Number System // Proceedings in Cybernetics. 2022. No. 4 (48). P. 77–90. DOI 10.34822/1999-7604-2022-4-77-90.

ВВЕДЕНИЕ

Существует метод выполнения арифметических операций над большими целыми чис-

лами, который основан на положениях теории чисел. Идея этого метода состоит в том, чтобы оперировать не непосредственно чис-

лом A , а его остатками или вычетами $\alpha_1 = A(\bmod p_1)$, $\alpha_2 = A(\bmod p_2)$, ..., $\alpha_n = A(\bmod p_n)$, где $p_1, p_2, \dots, p_n$ – модули, не содержащие общих делителей (взаимно-простых). Множество чисел, над которыми можно выполнять операции модулярной арифметики, – это $P = p_1 \cdot p_2 \cdot \dots \cdot p_n$ (произведение модулей), вычеты определяются как $\alpha_i = A - \left\lfloor \frac{A}{p_i} \right\rfloor p_i, i = 1, 2, \dots, n, \left\lfloor \frac{A}{p_i} \right\rfloor$ – целая часть меньшая или равная числу. Любое целое положительное число A из диапазона P можно представить в виде набора остатков от деления этого числа на выбранные основания системы, то есть $A = (\alpha_1, \alpha_2, \dots, \alpha_n)$ [1].

Рассматривая вычисления в модулярной арифметике, все операции можно разбить на две группы: модульные и немодульные.

Операции сложения, вычитания и умножения в модулярной арифметике относятся к модульным. Это операции, в которых действия над числами можно проводить независимо в параллельных каналах [2]. На компьютере с параллельным выполнением операций применение модулярной арифметики дает значительное преимущество в скорости для модульных операций. Операции, связанные с разными модулями, могут выполняться одновременно, и возникает повышение скорости их выполнения.

Преимущество представления чисел в модулярной системе счисления заключается в том, что операции сложения, вычитания и умножения выполняются достаточно просто:

$$\begin{aligned} &(\alpha_1, \alpha_2, \dots, \alpha_n) + (\beta_1, \beta_2, \dots, \beta_n) = \\ &= ((\alpha_1 + \beta_1) \bmod p_1, \dots, (\alpha_n + \beta_n) \bmod p_n), \\ &(\alpha_1, \alpha_2, \dots, \alpha_n) - (\beta_1, \beta_2, \dots, \beta_n) = \\ &= ((\alpha_1 - \beta_1) \bmod p_1, \dots, (\alpha_n - \beta_n) \bmod p_n), \\ &(\alpha_1, \alpha_2, \dots, \alpha_n) \cdot (\beta_1, \beta_2, \dots, \beta_n) = \\ &= ((\alpha_1 \cdot \beta_1) \bmod p_1, \dots, (\alpha_n \cdot \beta_n) \bmod p_n). \end{aligned} \quad (1)$$

Если при выполнении операции вычитания получен результат положительный, то операция выполняется вычитанием соответствующих цифр разряда, а если результат отрицательный, то берется ее дополнение к основанию [1].

Основной недостаток представления чисел в модулярной системе счисления состоит в том, что непросто проверить, является ли $(\alpha_1, \alpha_2, \dots, \alpha_n)$ большим, чем $(\beta_1, \beta_2, \dots, \beta_n)$. Трудно установить возникновение переполнения в результате выполнения операций сложения, вычитания и умножения, сложно выполнять операцию деления. Возникает необходимость выполнения немодульных операций, которые являются медленными, что вызывает определенные трудности.

Немодульные операции – это операции, в которых необходимо знать информацию обо всем числе и приходится восстанавливать позиционное представление числа. К немодульным операциям модулярной арифметики можно отнести следующие: перевод числа из модулярной системы счисления в позиционную, деление, вычисление позиционных характеристик (след, ядро, ранг и др.), расширение системы оснований, сравнение чисел, масштабирование чисел и др. [2–6].

МАТЕРИАЛЫ И МЕТОДЫ

К настоящему времени разработаны и продолжают разрабатывать алгоритмы выполнения перечисленных операций. Однако, несмотря на многочисленное количество работ в этой области [3, 6–12], еще остается много нерешенных вопросов.

В работе сосредоточено внимание на особенностях выполнения сложных и неочевидных алгоритмов. На первом этапе осуществлен анализ различных методов и алгоритмов выполнения модульных и немодульных операций в модулярной системе счисления. Далее в работе излагается материал относительно следующих немодульных операций: перевод из модулярной системы счисления (МСС) в позиционную систему счисления (ПСС) и сравнение чисел в МСС. Для каждой операции было выбрано два метода. Приведены их математическое описание и примеры расчета. На втором этапе были разработаны программы на языке Python, моделирующие выполнение алгоритмов на ЭВМ. На третьем этапе произведено сравнение алгоритмов между собой, а именно выполнена оценка сложности алгоритмов с целью выявления оптимальных.

Рассмотрим анализ существующих методов выполнения немодульных операций.

1. Перевод из модулярной системы счисления в позиционную систему счисления.

Переход из модулярной системы счисления в позиционную систему счисления выполняется с основанием, равным 10. То есть целое положительное число A представимо в виде:

$$A = a_n \cdot 10^n + a_{n-1} \cdot 10^{n-1} + \dots + a_1 \cdot 10^1 + a_0,$$

где a_i – целые десятичные цифры в интервале $[0, 9]$.

Операцию перевода чисел из модулярной системы счисления в позиционную можно считать одной из основных немодульных операций, так как определение величины числа необходимо для других немодульных операций: сравнение чисел по величине, определение знака и др. Эффективность выполнения этих операций будет напрямую зависеть от того, насколько эффективным будет метод определения величины числа.

Существуют как традиционные (метод ортогональных базисов, метод перевода в обобщенную позиционную систему счисления), так и новые методы перевода (интервальные методы перевода) [3, 6, 7].

Метод ортогональных базисов.

Основой метода ортогональных базисов является Китайская теорема об остатках (КТО) [7].

Теорема: Пусть $p_1, p_2, \dots, p_n$ – попарно взаимно простые числа, $P = p_1 \cdot p_2 \cdot \dots \cdot p_n$,

$y_1, y_2, \dots, y_n$ подобраны так, что $\frac{P}{p_i} y_i \equiv$

$\equiv 1 \pmod{p_i}$, $A_0 = \sum_{i=1}^n \frac{P}{p_i} y_i \alpha_i$, $i = 1, 2, \dots, n$. Тогда

решение системы $A = \alpha_i \pmod{p_i}$, $i = 1, 2, \dots, n$ будет иметь вид:

$$A \equiv A_0 \pmod{P}. \quad (2)$$

Пусть основания модулярной системы $p_1, p_2, \dots, p_n$, $P = p_1 \cdot p_2 \cdot \dots \cdot p_n$ – мощность диапазона модулярной системы. С выбором системы определяются ее основные константы – базисы $B_i = (\beta_{1i}, \beta_{2i}, \dots, \beta_{ni})$, $i = 1, 2, \dots, n$. Задача перевода числа $A = (\alpha_1, \alpha_2, \dots, \alpha_n)$ в ПСС

заключается в определении таких чисел M_i , $i = 1, 2, \dots, n$, чтобы $A = \sum_{i=1}^n M_i B_i$. Для однозначного определения M_i на базисы системы B_i накладывается ряд ограничений и показывается, что таким свойством обладают базисы [13]:

$$\begin{aligned} B_1 &= (1, 0, 0, \dots, 0, 0), \\ B_2 &= (0, 1, 0, \dots, 0, 0), \\ &\dots \\ B_n &= (0, 0, 0, \dots, 0, 1), \end{aligned} \quad (3)$$

которые называют ортогональными.

Тогда в случае ортогональных базисов $M_i = \alpha_i$, $i = 1, 2, \dots, n$. Ортогональные базисы определяют по формуле:

$$B_i = \frac{m_i \cdot P}{p_i} = m_i \cdot P_i, \quad i = 1, 2, \dots, n, \quad (4)$$

где $P_i = \frac{P}{p_i}$,

m_i – целые положительные числа, которые называются весами базиса. Их определяют из сравнений:

$$P_i \cdot m_i \equiv 1 \pmod{p_i}. \quad (5)$$

Тогда, по КТО, число:

$$A = (\alpha_1, \alpha_2, \dots, \alpha_n) \equiv \sum_{i=1}^n \alpha_i B_i \pmod{P}. \quad (6)$$

Таким образом, если найдены ортогональные базисы для системы оснований, то для перевода числа $A = (\alpha_1, \alpha_2, \dots, \alpha_n)$ достаточно вычислить $\sum_{i=1}^n \alpha_i B_i$ и ввести эту сумму в диапазон $[0; P)$ вычитанием величины, кратной P , т. е.:

$$A = \left| \sum_{i=1}^n \alpha_i B_i \right|_P = \sum_{i=1}^n \alpha_i B_i - r_A P, \quad (7)$$

где r_A – ранг числа A , показывающий, сколько раз надо вычесть мощность диапазона P из полученного числа, чтобы вернуть его в диапазон [14].

Пример 1. Пусть дана система оснований $p_1 = 32765$, $p_2 = 32767$, $p_3 = 32768$, $p_4 = 32769$, $p_5 = 32771$. Мощность диапазона модулярной системы $P = p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5 = 37778931511113441116160$. Переведем число,

представленное в МСС $A = (36, 4, 0, 4, 36)$, в ПСС.

Вычислим ортогональные базисы. Для этого найдем величин P_i по формуле:

$$P_i = \frac{P}{p_i}, \quad (8)$$

где P – диапазон модулярной системы, $p_1, p_2, \dots, p_n$ – основания.

$$\begin{aligned} P_1 &= \frac{P}{p_1} = \frac{37778931511113441116160}{32765} = \\ &= 1153027056649273344, \\ P_2 &= \frac{P}{p_2} = \frac{37778931511113441116160}{32765} = \\ &= 1152956679314964480, \\ P_3 &= \frac{P}{p_3} = \frac{37778931511113441116160}{32765} = \\ &= 1152921493869428745, \\ P_4 &= \frac{P}{p_4} = \frac{37778931511113441116160}{32765} = \\ &= 1152886310571376640, \\ P_5 &= \frac{P}{p_5} = \frac{37778931511113441116160}{32765} = \\ &= 1152815950416936960. \end{aligned}$$

Веса базисов вычислим по формуле (5). Для нашего случая:

$$115292149386948745 \cdot m_3 \equiv 1 \pmod{32768}.$$

Вычислим m_3 методом цепных дробей, содержащим этапы:

$$\begin{aligned} P_3 \cdot m_3 &\equiv 1 \pmod{p_3} \\ m_3 &= (-1)^n \cdot F_{n-1} \cdot 1 \pmod{p_3}, \end{aligned} \quad (9)$$

где $F_0 = q_0, F_1 = q_0 \cdot q_1 + 1, \dots, F_n = q_n \cdot F_{n-1} + F_{n-2}$.

$$\begin{aligned} \frac{p_3}{P_3} &= \frac{32768}{1152921493869428845} = \\ &= [q_0, q_1, q_2, q_3, q_4] = \\ &= [0, 35184371761152, 36340, 1, 8], \end{aligned}$$

$$\frac{1}{35184371761152 + \frac{1}{3640 + \frac{1}{1 + \frac{1}{8}}}}.$$

Последним элементом является q_4 , и индекс равен 4, следовательно, $n = 4$.

Получим:

$$\begin{aligned} m_3 &= (-1)^n \cdot F_{n-1} \cdot 1 \pmod{p_1} \\ &= (-1)^4 \cdot F_{4-1} \cdot 1 \pmod{32768} \\ &= 1 \cdot F_3 \cdot 1 \pmod{32768} \\ &= 1 \cdot 3641 \cdot 1 \pmod{32768} \\ &= 3641 \\ F_3 &= q_3 \cdot F_{3-1} + F_{3-2} = q_3 \cdot F_2 + F_1 = \\ &= q_3(q_2(q_0 \cdot q_1 + 1) + q_0) + q_0 \cdot q_1 + 1 = \\ &= 1(3640(0 \cdot 35184371761152 + 1) + 0) + \\ &\quad + 0 \cdot 35184371761152 + 1 = 3641 \\ 1152921493869428745 \cdot m_3 &= 3641 \pmod{32768}, \\ m_3 &= 3641 \pmod{32768} = 3641. \end{aligned}$$

Аналогично вычисляются константы m_1, m_2, m_4, m_5 :

$$m_1 = 9784, m_2 = 30719, m_4 = 2048, m_5 = 1934.$$

Определим ортогональные базисы по формуле (4):

$$\begin{aligned} B_1 &= m_1 \cdot P_1 = 9784 \cdot 1153027056649273344 = \\ &= 11281216722256490397696, \\ B_2 &= m_2 \cdot P_2 = 30719 \cdot 1152956679314964480 = \\ &= 35417676231876393861120, \\ B_3 &= m_3 \cdot P_3 = 3641 \cdot 1152921493869428745 = \\ &= 4197787159178590060545, \\ B_4 &= m_4 \cdot P_4 = 2048 \cdot 1152886310571376640 = \\ &= 2361111164050179358720, \\ B_5 &= m_5 \cdot P_5 = 19344 \cdot 1152815950416936960 = \\ &= 22300071744865228554240. \end{aligned}$$

Вычислим величину числа A согласно формуле (6), зная, что $A = (\alpha_1, \alpha_2, \dots, \alpha_n) = (36, 4, 0, 4, 36)$:

$$\begin{aligned}
 A &= (\alpha_1 \cdot B_1 + \alpha_2 \cdot B_2 + \alpha_3 \cdot B_3 + \alpha_4 \cdot B_4 + \alpha_5 \cdot B_5) \times \\
 &\times (\text{mod } P) = (36 \cdot 11281216722256490397696 + \\
 &+ 4 \cdot 35417676231876393861120 + \\
 &+ 0 \cdot 4197787159178590060545 + \\
 &+ 4 \cdot 2361111164050179358720 + \\
 &+ 36 \cdot 22300071744865228554240) \times \\
 &\times (\text{mod } 3778931511113441116160) = \\
 &= (406123802001233654317056 + \\
 &+ 141670704927505575444480 + 0 + \\
 &+ 9444444656200717434880 + \\
 &+ 802802582815148227952640) \times \\
 &\times (\text{mod } 3778931511113441116160) = \\
 &= 1360041534400088175149056 \times \\
 &\times (\text{mod } 3778931511113441116160) = \\
 &= 4294967296.
 \end{aligned}$$

Получаем $A = 4294967296$.

Метод перевода определения величины числа из МСС в обобщенную позиционную систему счисления (ОПСС).

Пусть МСС задается основаниями $p_1, p_2, \dots, p_n$, и $A = (\alpha_1, \alpha_2, \dots, \alpha_n)$ – число в этой системе. Число в ОПСС задается кортежем $[x_1, x_2, \dots, x_n]$, основаниями системы являются следующие константы $p_1, p_1p_2, p_1p_2p_3, \dots, p_1p_2 \dots p_{n-1}$. Число X в ОПСС можно представить в виде:

$$\begin{aligned}
 X &= x_n p_1 p_2 \dots p_{n-1} + x_{n-1} p_1 p_2 \dots p_{n-2} + \\
 &+ \dots + x_3 p_1 p_2 + x_2 p_1 + x_1,
 \end{aligned} \quad (10)$$

где $0 \leq x_k \leq \prod_{i=1}^{k-1} p_i$, $i = 1, 2, \dots, n$ – коэффициенты ОПСС [6].

Очевидно, что диапазоны чисел, представимых в МСС и ОПСС, совпадают, имеется взаимно-однозначное соответствие между множеством представлений чисел в МСС и ОПСС.

Равенство (10) можно переписать в виде:

$$\begin{aligned}
 X &= x_1 + p_1(x_2 + p_2(x_3 + \dots + \\
 &+ p_{n-2}(x_{n-1} + p_{n-1}x_n) \dots)).
 \end{aligned}$$

Откуда следует, что цифры ОПСС могут быть получены из соотношений:

$$\begin{aligned}
 x_1 &= X - \left\lfloor \frac{X}{p_1} \right\rfloor p_1 = X - X_1 p_1, \\
 \text{где } X_1 &= \left\lfloor \frac{X}{p_1} \right\rfloor, \\
 x_2 &= X_1 - \left\lfloor \frac{X_1}{p_2} \right\rfloor p_2 = X_1 - X_2 p_2, \\
 \text{где } X_2 &= \left\lfloor \frac{X_1}{p_2} \right\rfloor, \\
 &\dots \\
 x_n &= X_{n-1} - \left\lfloor \frac{X_{n-1}}{p_n} \right\rfloor p_n = X_{n-1} - X_n p_n, \\
 \text{где } X_n &= \left\lfloor \frac{X_{n-1}}{p_n} \right\rfloor.
 \end{aligned} \quad (11)$$

При определении цифр x_i по формулам (11) все вычисления можно вести в МСС.

Из (11) следует, что $x_i = |X|_{p_i}$, то есть x_1 – первая МСС цифра, или $x_1 = \alpha_1$. Для получения x_1 сперва $X - x_1$ представим в остаточной коде. Очевидно, $X - x_1$ делится на p_1 . Более того, p_1 взаимно просто со всеми другими модулями. Следовательно, для нахождения цифры x_2 может быть использована процедура деления без остатка: $x_2 = \left\lfloor \frac{X - x_1}{p_1} \right\rfloor_{p_2}$.

Таким путем с помощью вычитаний и делений в остаточной записи все цифры ОПСС могут быть получены. При этом

$$\begin{aligned}
 x_1 &= |X|_{p_1}, \quad x_2 = \left\lfloor \frac{X}{p_1} \right\rfloor_{p_2}, \quad x_3 = \left\lfloor \frac{X}{p_1 p_2} \right\rfloor_{p_3}, \quad \dots, \\
 x_i &= \left\lfloor \frac{X}{p_1 p_2 \dots p_{i-1}} \right\rfloor_{p_i}, \quad i > 1.
 \end{aligned}$$

Перевод, осуществляемый согласно алгоритму (11), содержит всего $\frac{n(n-1)}{2}$ остаточные арифметические операции вычитания и деления без остатка, где n – число модулей системы. Может быть предложена некоторая модификация рассмотренного алгоритма

в том плане, что операция деления заменяется операцией умножения. Для этого предварительно вычисляется $\frac{n(n-1)}{2}$ констант t_{kj} , которые удовлетворяют условию:

$$t_{kj} p_k \equiv 1 \pmod{p_j}, (1 \leq k < j \leq n).$$

Эти константы можно получить, например, из расширенного алгоритма Евклида:

$$t_{kj} p_k + \gamma p_j = \text{НОД}(p_k, p_j) = 1 \text{ [15]}.$$

Здесь следует заметить тот факт, что константы t_{kj} полностью определяются к выбранной системой основания, поэтому могут быть вычислены заранее и храниться в некоторой таблице.

Если константы t_{kj} вычислены, то вычисление цифр x_i ОПСС по алгоритму (11) может быть переписано в виде [6]:

$$\begin{aligned} x_1 &\equiv \alpha_1 \pmod{p_1}, \\ x_2 &\equiv (\alpha_2 - x_1) \cdot t_{12} \pmod{p_2}, \\ x_3 &\equiv ((\alpha_3 - x_1) \cdot t_{13} - x_2) \cdot t_{23} \pmod{p_3}, \\ &\vdots \\ x_n &\equiv (((\dots(\alpha_n - x_1) \cdot t_{1n} - x_2) - \dots - x_{n-1}) \times \\ &\quad \times t_{n-1n} \pmod{p_n}). \end{aligned} \quad (12)$$

Константы t_{kj} принято также записывать в виде:

$$t_{kj} = \left| \frac{1}{p_k} \right|_{p_j} \quad (13)$$

и называть обратными элементами по умножению для чисел p_k по модулю p_j .

Пример 2. Пусть дана система оснований $p_1 = 32765$, $p_2 = 32767$, $p_3 = 32768$, $p_4 = 32769$, $p_5 = 32771$. Мощность диапазона модулярной системы $P = p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5 = 37778931511113441116160$. Переведем число, представленное в МСС $A = (36, 4, 0, 4, 36)$, в ПСС.

Сначала найдем константы t_{kj} , используя формулу (13):

$$t_{12} = \left| \frac{1}{p_1} \right|_{p_2} = \left| \frac{1}{32765} \right|_{32767} = 16383.$$

Опишем, как вычислить $\left| \frac{1}{32765} \right|_{32767}$.

Эта запись равносильна $\frac{1}{32765} \pmod{32767}$.

Введем следующее обозначение:

$$\begin{aligned} \frac{1}{p_1} &= x, \\ p_1 \cdot x &\equiv 1 \pmod{p_2}. \end{aligned}$$

То есть $p_1 \cdot x - 1 \vdots p_2$.

Тогда:

$$\frac{1}{32765} = x.$$

Вычислим сравнение:

$$32765 \cdot x \equiv 1 \pmod{32767}.$$

Здесь можно воспользоваться алгоритмом цепных дробей:

$$1 + \frac{1}{16382 + \frac{1}{2}}$$

$$\frac{32767}{32765} = [q_0, q_1, q_2] = [1, 16382, 2].$$

$n = 2$, так как q_2 последний элемент.

$$\begin{aligned} x &= (-1)^n \cdot F_{n-1} \cdot 1 \pmod{p_2} = \\ &= (-1)^2 \cdot F_{2-1} \cdot 1 \pmod{32767} = \\ &= 1 \cdot F_1 \cdot 1 \pmod{32767} = 16383 \pmod{32767} = \\ &= 16383. \end{aligned}$$

$$F_1 = q_0 \cdot q_1 + 1 = 1 \cdot 16382 + 1 = 16383.$$

Вернемся к нашему сравнению: $32765 \cdot x \equiv 1 \pmod{32767}$.

Подставим $x = 16383$:

$$32765 \cdot 16383 \equiv 1 \pmod{32767},$$

$$536788995 \equiv 1 \pmod{32767}.$$

Действительно, $536788995 - 1 : 32767$.
То есть:

$$t_{12} = \left| \frac{1}{p_1} \right|_{p_2} = \left| \frac{1}{32765} \right|_{32767} = 16383.$$

Далее аналогично находим остальные константы t_{kj} :

$$t_{12} = \left| \frac{1}{32765} \right|_{32767} = 16383;$$

$$t_{13} = \left| \frac{1}{32765} \right|_{32768} = 21845;$$

$$t_{14} = \left| \frac{1}{32765} \right|_{32769} = 8192;$$

$$t_{15} = \left| \frac{1}{32765} \right|_{32771} = 27309;$$

$$t_{23} = \left| \frac{1}{32767} \right|_{32768} = 32767;$$

$$t_{24} = \left| \frac{1}{32767} \right|_{32769} = 16384;$$

$$t_{25} = \left| \frac{1}{32767} \right|_{32771} = 24578;$$

$$t_{34} = \left| \frac{1}{32768} \right|_{32769} = 16384;$$

$$t_{35} = \left| \frac{1}{32768} \right|_{32771} = 21847;$$

$$t_{45} = \left| \frac{1}{32769} \right|_{32771} = 16385.$$

Константы t_{kj} запишем в виде матрицы $k \times j$:

$$\begin{pmatrix} 0 & 16383 & 21845 & 8192 & 27309 \\ 0 & 0 & 32767 & 16384 & 24578 \\ 0 & 0 & 0 & 16384 & 21847 \\ 0 & 0 & 0 & 0 & 16385 \end{pmatrix}.$$

Далее найдем x_1, x_2, x_3, x_4 по формуле (12), зная, что:

$$A = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) = (36, 4, 0, 4, 36);$$

$$p_1 = 32765, p_2 = 32767, p_3 = 32768,$$

$$p_4 = 32769, p_5 = 32771,$$

$$x_1 \equiv \alpha_1 \pmod{p_1} = 36 \pmod{32765} = 36,$$

$$\begin{aligned} x_2 &\equiv (\alpha_2 - x_1) \cdot t_{12} \pmod{p_2} = \\ &= (4 - 36) \cdot 16383 \pmod{32767} = \\ &= -524256 \pmod{32767} = 16, \end{aligned}$$

$$\begin{aligned} x_3 &\equiv ((\alpha_3 - x_1) \cdot t_{13} - x_2) \cdot t_{23} \pmod{p_3} = \\ &= ((0 - 36) \cdot 21845 - 16) \cdot 32767 \pmod{32768} = \\ &= -25769148412 \pmod{32768} = 4, \end{aligned}$$

$$\begin{aligned} x_4 &\equiv (((\alpha_4 - x_1) \cdot t_{14} - x_2) \cdot t_{24} - x_3) \cdot t_{34} \pmod{p_4} = \\ &= (((4 - 36) \cdot 8192 - 16) \cdot 16384 - 4) \cdot 16384 \times \\ &\times \pmod{32769} = -70373039210496 \pmod{32769} = 0, \end{aligned}$$

$$\begin{aligned} x_5 &\equiv (((\alpha_5 - x_1) \cdot t_{15} - x_2) \cdot t_{25} - x_3) \cdot t_{35} - x_4 \cdot t_{45} \times \\ &\times \pmod{p_5} = (((36 - 36) \cdot 27309 - 16) \cdot 24578 - 4) \times \\ &\times 21847 - 0 \cdot 16385 \pmod{32771} = \\ &= -140769703034940 \pmod{32771} = 0. \end{aligned}$$

Запишем результат в виде табл. 1.

Таблица 1

**Промежуточные вычисления метода перевода
в обобщенную позиционную систему счисления**

Операции	Модули					Цифры ОПСС
	$p_1 = 32765$	$p_1 = 32767$	$p_1 = 32768$	$p_1 = 32769$	$p_1 = 32771$	
A	36	4	0	4	36	$x_1 = 36$
-						
x_1	36	36	36	36	36	

Окончание табл. 1

Операции	Модули					Цифры ОПСС
	$p_1 = 32765$	$p_1 = 32767$	$p_1 = 32768$	$p_1 = 32769$	$p_1 = 32771$	
$A - x_1$ $\times$ t_{1j}	0	32735 16383	32732 21845	32737 8192	0 27309	
A_1 — x_2		16 16	12 16	8 16	0 16	$x_2 = 16$
$A_1 - x_2$ $\times$ t_{2j}		0	32764 32767	32761 16384	32755 24578	
A_2 — x_3			4 4	4 4	4 4	$x_3 = 4$
$A_2 - x_3$ $\times$ t_{3j}			0	0 16384	0 21847	
A_3 — x_4				0 0	0 0	$x_4 = 0$
$A_3 - ax_4$ $\times$ t_{4j}				0	0 16385	
A_4					0	$x_5 = 0$

Примечание: составлено автором на основании данных, полученных в исследовании.

Применяя формулу (10), получим:

$$\begin{aligned}
 X &= x_5 p_1 p_2 p_3 p_4 + x_4 p_1 p_2 p_3 + x_3 p_1 p_2 + x_2 p_1 + x_1 = \\
 &= 0 \cdot 32765 \cdot 32767 \cdot 32768 \cdot 32769 + 0 \cdot 32765 \cdot 32767 \times \\
 &\quad \times 32768 + 4 \cdot 32765 \cdot 32767 + 16 \cdot 32765 + 36 = \\
 &= 0 + 0 + 4294443020 + 524240 + 36 = 42944967296.
 \end{aligned}$$

2. Сравнение чисел в МСС.

Исследование и совершенствование существующих, а также разработка новых методов и алгоритмов реализации немодульной операции сравнения данных, представленных в модулярной арифметике, является важной и актуальной задачей [8–12]. Операция сравнения чисел широко используется при реализации большинства алгоритмов. Можно заметить, что в модулярной арифметике имеется три группы методов сравнения [8–10]. К первой группе относятся методы, основанные на преобразовании чисел из модулярной системы счисления в позиционную, далее выполняются

сравнение полученных чисел. Ко второй группе относятся методы, основанные на применении понятия нулевизации [3]. К третьей группе относятся методы, основанные на использовании позиционных характеристик.

Метод, основанный на переводе из чисел из МСС в ПСС.

Для перевода числа из МСС в ППС используется стандартное восстановление с помощью КТО [3], которую можно записать формулой:

$$X = \left| \sum_{i=1}^n P_i \cdot x_i \cdot |P_i^{-1}|_{p_i} \right|_P, \quad (14)$$

где $P_i = \frac{P}{p_i}$, а $|P_i^{-1}|_{p_i}$ — мультипликативная инверсия P_i по модулю p_i .

Рассмотрим пример восстановления числа по формуле (14) и сравнения чисел.

Пример 3. Пусть задана МСС $p_1 = 32765, p_2 = 32767, p_3 = 32768, p_4 = 32769$, и $p_5 = 32771$ числа $X = (36, 4, 0, 4, 36)$, $Y = (9, 1, 0, 1, 9)$. Мощность диапазона модулярной системы счисления равна $P = p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5 = 37778931511113441116160$. Сравним числа X и Y .

Применяя формулы (8) и (5), вычислим P_i, m_i .

Из примера 1 $P_1, P_2, P_3, P_4, P_5, m_1, m_2, m_3, m_4, m_5$, соответственно, равны:

$$\begin{aligned} P_1 &= 1153027056649273344, \\ P_2 &= 1152956679314964480, \\ P_3 &= 1152921493869428745, \\ P_4 &= 1152886310571376640, \\ P_5 &= 1152815950416936960, \\ m_1 &= 9784, \quad m_2 = 30719, \quad m_3 = 3641, \\ m_4 &= 2048, \quad m_5 = 19344. \end{aligned}$$

Применяя формулу (14), найдем значение чисел X и Y :

$$\begin{aligned} X &= \left\lfloor \frac{1153027056649273344 \cdot 36 \cdot 9784 + 1152956679314964480 \cdot 4 \cdot 30719 + 1152921493869428745 \cdot 0 \cdot 3641 + 1152886310571376640 \cdot 4 \cdot 2048 + 1152815950416936960 \cdot 36 \cdot 19344}{37778931511113441116160} \right\rfloor = \\ &= \left\lfloor \frac{406123802001233654317056 + 141670704927505575444480 + 9444444656200717434880 + 802802582815148227952640}{37778931511113441116160} \right\rfloor = \\ &= \left\lfloor \frac{1360041534400088175149056}{37778931511113441116160} \right\rfloor = 4294967296. \\ Y &= \left\lfloor \frac{1153027056649273344 \cdot 9 \cdot 9784 + 1152956679314964480 \cdot 1 \cdot 30719 + 1152921493869428745 \cdot 0 \cdot 3641 + 1152886310571376640 \cdot 1 \cdot 2048 + 1152815950416936960 \cdot 9 \cdot 19344}{37778931511113441116160} \right\rfloor = \\ &= \left\lfloor \frac{101530950500308413579264 + 35417676231876393861120 + 2361111164050179358720 + 200700645703787056988160}{37778931511113441116160} \right\rfloor = \\ &= \left\lfloor \frac{340010383600022043787264}{37778931511113441116160} \right\rfloor = 1073741824. \end{aligned}$$

Так как $4294967296 > 1073741824$, значит $X > Y$.

Метод с использованием позиционных характеристик.

Отличным от вышеизложенного метода сравнения чисел является метод на основе использования минимальной функции ядра Акушского [10].

Была предложена аналитическая функция для вычисления *Pirlo* функции:

$$Pi(X) = \left\lfloor \sum_{i=1}^n k_i^{**} \cdot x_i \right\rfloor_{p_n}, \quad (15)$$

$$\text{где } k_i^{**} = \frac{\left\lfloor P_i^{-1} \right\rfloor_{p_i} \cdot P_i}{p_n}.$$

Так как функция *Pirlo* является монотонно возрастающей, то она может быть использована для сравнения чисел, т. е. если $Pi(X) < Pi(Y)$, то $X < Y$. Однако возможны случаи, когда $Pi(X) = Pi(Y)$, и в этом случае $X < Y$, когда $x_n < y_n$.

Пример 4. Пусть задана МСС $p_1 = 32765, p_2 = 32767, p_3 = 32768, p_4 = 32769, p_5 = 32771$ и числа $X = (36, 4, 0, 4, 36)$, $Y = (9, 1, 0, 1, 9)$. Мощность диапазона модулярной системы счисления равна $P = p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5 = 37778931511113441116160$. Сравним числа X и Y .

Вычислим значения:

$$\begin{aligned} P_5 &= 1152815950416936960, \\ k_1^{**} &= \frac{\left\lfloor P_1^{-1} \right\rfloor_{p_1} \cdot P_1}{p_5} = \frac{9784 \cdot 1153027056649273344}{32771} = \\ &= 344243896196530176, \\ k_2^{**} &= \frac{\left\lfloor P_2^{-1} \right\rfloor_{p_2} \cdot P_2}{p_5} = \frac{30719 \cdot 1152956679314964480}{32771} = \\ &= 1080762754626846720, \\ k_3^{**} &= \frac{\left\lfloor P_3^{-1} \right\rfloor_{p_3} \cdot P_3}{p_5} = \frac{3641 \cdot 1152921493869428745}{32771} = \\ &= 128094570174196395, \\ k_4^{**} &= \frac{\left\lfloor P_4^{-1} \right\rfloor_{p_4} \cdot P_4}{p_5} = \frac{2048 \cdot 1152886310571376640}{32771} = \\ &= 72048798146232320, \end{aligned}$$

$$k_5^{**} = \frac{|P_5^{-1}|_{p_5} \cdot P_5}{p_5} = \frac{19344 \cdot 1152815950416936960}{32771} = 680481881690068309.$$

$$Pi(Y) = \frac{\begin{vmatrix} 344243896196530176 \cdot 9 + \\ +1080762754626846720 \cdot 1 + \\ 128094570174196395 \cdot 0 + \\ +72048798146232320 \cdot 1 + \\ +680481881690068309 \cdot 9 \end{vmatrix}}{1152815950416936960} =$$

Найдем значение функции *Pirlo* для чисел *X* и *Y* по формуле (15):

$$Pi(X) = \frac{\begin{vmatrix} 344243896196530176 \cdot 36 + \\ +1080762754626846720 \cdot 4 + \\ +128094570174196395 \cdot 0 + \\ +72048798146232320 \cdot 4 + \\ +680481881690068309 \cdot 36 \end{vmatrix}}{1152815950416936960} =$$

$$= \frac{\begin{vmatrix} 3098195065768771584 + \\ +1080762754626846720 + \\ +72048798146232320 + \\ 6124336935210614781 \end{vmatrix}}{1152815950416936960} =$$

$$= \frac{10375343553752465405}{1152815950416936960} = 32765$$

и поскольку $Pi(X) > Pi(Y)$, то $X > Y$.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

На втором этапе на высокоуровневом языке программирования Python были разработаны алгоритмы выполнения рассмотренных немодульных операций. Приведем фрагменты алгоритмов и результаты расчетов (рис. 1, 2).

```
Base = [32765, 32767, 32768, 32769, 32771] M = 37778931511113441116160
##### Перевод из модулярной системы счисления в позиционную систему счисления #####
##### Метод ортогональных базисов #####
(36, 4, 0, 4, 36)
4294967296
rank (простая версия) = 36
----- Определение ранга по алгоритму -----
M = [[1, 1, 1, 1, 1], [0, 32765, 32765, 32765, 32765], [0, 0, 3, 8, 24], [0, 0, 0, 32761, 32699], [0, 0, 0, 0, 144]]
M_rank = [2, 55746, 15, 21349, 85]
[0, 32733, 32729, 32733, 32765] 55674
[0, 0, 15, 40, 120] 39
[0, 0, 0, 32761, 32699] 21313
[0, 0, 0, 0, 144] 49
[0, 0, 0, 0, 0] -37
rank (по алгоритму) = 36
##### Метод перевода в обобщенную позиционную систему счисления #####
(36, 4, 0, 4, 36)
----- Перевод в ПСС (вариант 2) -----
t_matrix = [[0, 16383, 21845, 8192, 27309], [0, 0, 32767, 16384, 24578], [0, 0, 0, 32768, 21847], [0, 0, 0, 0, 16385]]
A_matrix = [36, 16, 4, 0, 0]
4294967296
```

Рис. 1. Результаты расчетов по алгоритму

«Перевод из модулярной системы счисления в позиционную систему счисления»

Примечание: составлено автором на основании данных, полученных в исследовании.

```
##### Сравнение чисел в МСС #####
##### Метод, основанный на переводе из чисел из МСС в ПСС #####
Base = [32765, 32767, 32768, 32769, 32771] M = 37778931511113441116160
(36, 4, 0, 4, 36) < (9, 1, 0, 1, 9)
False

##### Метод с использованием позиционных характеристик #####
Base = [32765, 32767, 32768, 32769, 32771] M = 37778931511113441116160
(36, 4, 0, 4, 36) 4294967296
(9, 1, 0, 1, 9) 1073741824
----- сравнения чисел в СОК с использованием позиционных характеристик -----
k = [344243896196530176, 1080762754626846720, 128094570174196395, 72048798146232320, 680481881690068309]
Pi (36, 4, 0, 4, 36) = 131060
Pi (9, 1, 0, 1, 9) = 32765
1
```

Рис. 2. Результаты расчетов по алгоритму «Сравнение чисел в МСС»

Примечание: составлено автором на основании данных, полученных в исследовании.

Сравнив математическое описание и результаты расчетов по разработанным алгоритмам, убеждаемся в совпадении результатов, что свидетельствует о корректности вычислений. Сравнение проводилось с применением процесса математического моделирования. Объектом моделирования являются методы выполнения операций перевода и сравнения чисел в МСС. В качестве математической модели выступает совокупность математических формул и отношений между ними, которая адекватно отражает концепцию рассматриваемых методов. Процесс математического моделирования можно представить в виде следующих этапов: исследование объекта моделирования, постановка задачи, анализ, выбор методов, поиск решения, разработка алгоритма решения, проверка, т. е. соответствие результатов математической модели и разработанных алгоритмов, практическое использование и анализ результатов моделирования.

Выполнена оценка сложности разработанных алгоритмов. При оценке вычислительной сложности алгоритмов используют два подхода: временная вычислительная сложность и асимптотическая вычислительная сложность.

Определение: вычислительная сложность (алгоритмическая сложность) – функция зависимости объема работы алгоритма от размера обрабатываемых данных.

Вычислительная сложность отвечает на центральный вопрос при разработке алгоритмов: как изменится время исполнения и объем занятой памяти в зависимости от размера входных данных.

Определение: временная сложность алгоритма – это функция от размера входных дан-

ных, равная количеству элементарных операций, выполняемых алгоритмом для решения экземпляра задачи указанного размера.

Вычислительная сложность – это количественная оценка ресурсов, затрачиваемых алгоритмом. Вычислительная сложность является более общим термином, чем временная сложность, поскольку процессорное время – это не единственный ресурс, который необходим для выполнения алгоритма. Временная сложность – это количество времени, необходимое на выполнение алгоритма. Другим ресурсом является память, т. е. пространственная сложность – количество памяти, которое требуется для выполнения алгоритма.

Определение: асимптотическая сложность – оценка сложности алгоритма с использованием предельного перехода при стремлении к бесконечности размерности входных данных.

Для обозначения оценки сложности алгоритмов используется O -нотация, которая определяет характеристики функции, показывающей, как изменяется вычислительная сложность алгоритма при изменении количества входных данных в худшем случае.

Введем обозначения: $O(n)$ – оценка количества операций, где n – это количество модулярных оснований на входе алгоритма (m – максимальное основание МСС). В работе количество модулярных оснований равно 5, т. е. $n=5$, максимальное основание МСС $m=32771$.

Поясним на примере операции перевода из модулярной в позиционную систему счисления методом ортогональных базисов, как производится оценка параметров в строках и столбцах табл. 2. Алгоритм метода представлен на рис. 3.

Таблица 2

Оценка сложности методов выполнения немодульных операций в МСС

Перевод из модулярной системы счисления в позиционную систему счисления			
Модулярные основания $p_1 = 32765, p_2 = 32767, p_3 = 32768, p_4 = 32769, p_5 = 32771$			
Метод ортогональных базисов			
Количество модулярных оснований n	Операции	Оценка количества	Общая сложность метода
5	Умножение	$O(n)$	$O(n^2)$
	Сложение	$O(n)$	

Окончание табл. 2

Количество модулярных оснований n	Операции	Оценка количества	Общая сложность метода
Метод перевода в обобщенную позиционную систему счисления			
5	Вычисление мультипликативной инверсии	$O(n^2)$ сложностью $O(\log^2(m))$	$O(n^2 \cdot \log^2(m))$
	Деление по модулю	$O(n^2)$	
	Умножение	$O(n^2)$	
	Сложение	$O(n^2)$	
Сравнение чисел в МСС			
Модулярные основания $p_1 = 32765, p_2 = 32767, p_3 = 32768, p_4 = 32769, p_5 = 32771$			
Метод, основанный на переводе чисел из МСС в ПСС			
5	Умножение	$O(n)$	$O(n^2)$
	Сложение	$O(n)$	
Метод с использованием позиционных характеристик			
5	Вычисление мультипликативной инверсии	$O(n)$ сложностью $O(\log^2(m))$	$O(n \cdot \log^2(m))$
	Деление по модулю	$O(n)$	
	Умножение	$O(n)$	
	Сложение	$O(n)$	
	Деление по модулю	$O(n^2)$	
	Умножение	$O(n^2)$	
	Сложение	$O(n^2)$	

Примечание: составлено автором на основании данных, полученных в исследовании.

```
def to_int(self, return_rank=False):
    res = 0
    for i, pp in enumerate(self.rns.basis):
        m = self.rns.weight[i]
        b = m * pp
        res += b * self.residues[i]
    if return_rank:
```

Рис. 3. Алгоритм перевода из МСС в ПСС методом ортогональных базисов

Примечание: составлено автором на основании данных, полученных в исследовании.

В строке for i, pp in enumerate (self.rns.basis) имеем цикл размера n .

В строке $b = m * pp$ имеем одно умножение.

В строке $res += b * self.residues[i]$ имеем одно умножение и сложение, но поскольку res в общем случае большое число, то сложность этого сложения тоже составляет n . В итоге получаем общую сложность метода $O(n^2)$.

сравнения чисел в модулярной системе счисления. Алгоритмы используют произвольные наборы модулей, обеспечивающих вычисления в большом динамическом диапазоне. В качестве примера набора оснований МСС были выбраны числа $p_1 = 32765, p_2 = 32767, p_3 = 32768, p_4 = 32769, p_5 = 32771$ с мощностью диапазона:

$$P = 37778931511113441116160.$$

Вычислительный диапазон близок к значению:

ЗАКЛЮЧЕНИЕ

Рассмотрены алгоритмы перевода чисел из модулярной системы счисления в позиционную систему счисления, а также алгоритмы

$$2^{75} = 37778931862957161709568.$$

Для интерпретации результатов исследования на практике взяты числа

$$A = 2^{32} = 4294967296 = (36, 4, 0, 4, 36),$$

$$B = 2^{30} = 1073741824 = (9, 1, 0, 1, 9).$$

Анализируя операцию перевода из МСС в ПСС и сравнивая методы ортогональных базисов и перевода в ОПСС, отметим, что недостатком первого метода является то, что необходимо выполнять операции с большими числами B_i . Операции сложения и умножения нужно выполнять в ПСС, и если полученный результат выходит за пределы вычислительного диапазона МСС, то необходимо вычис-

лять ранг числа. Преимуществом второго метода является выполнение большинства параллельных вычислений в МСС. Оценивая сложности алгоритмов, получим, что метод ортогональных базисов имеет меньшую сложность.

Рассматривая сравнения чисел в МСС, в частности метод, основанный на переводе чисел из МСС в ПСС, и метод с использованием позиционных характеристик, в первом случае учитывается вычислительная сложность получения остатка от деления на большое число P , во втором – функция *Pirlo* проигрывает КТО, так как требует дополнительных сравнений числа. Первый метод является более эффективным и имеет меньшую асимптотическую вычислительную сложность.

Список источников

1. Лобес М. В. Разработка методов и алгоритмов модулярных вычислений для задач большой алгоритмической сложности : автореф. дис. ... канд. физ.-мат. наук. Ставрополь, 2009. 20 с.
2. Лавриненко А. Н., Червяков Н. И. Исследование немодульных операций в системе остаточных классов // Науч. ведомости. Компьютер. моделирование 2012. № 1 (120), Вып. 21/1. С. 110–122.
3. Акушский И. Я., Юдицкий Д. И. Машинная арифметика в остаточных классах. М. : Сов. Радио, 1968. 439 с.
4. Исупов К. С. Методика выполнения базовых немодульных операций в модулярной арифметике с применением интервальных позиционных характеристик // Изв. высш. учеб. заведений. Поволжский регион. Технич. науки. 2013. № 3 (27). С. 26–39.
5. Червяков Н. И., Авербух В. М., Бабенко М. Г. и др. Приближенный метод выполнения немодульных операций в системе остаточных классов // Фундамент. исслед. 2012. № 6–1. С. 189–193. URL: <https://fundamental-research.ru/ru/article/view?id=29963> (дата обращения: 20.10.2022).
6. Ляхов П. А., Грובה Т. А., Карасев И. В. Моделирование алгоритмов обратного преобразования чисел из системы остаточных классов в позиционную систему счисления // Актуал. направления науч. исслед. XXI века: теория и практика. 2015. Т. 3, № 7–4 (18–4). С. 430–433.
7. Ляхов П. А., Голошубова Ю. В., Попова Е. А. Сравнительный анализ методов перевода чисел из системы остаточных классов в позиционную систему счисления // Молодой ученый. 2017. № 22 (156). С. 1–6. URL: <https://moluch.ru/archive/156/44137/> (дата обращения: 19.10.2022).

References

1. Lobes M. V. Razrabotka metodov i algoritmov modularnykh vychislenii dlia zadach bolshoi algoritmicheskoi slozhnosti : Extended abstract of Cand. Sci. Dissertation (Physics and Mathematics). Stavropol, 2009. 20 p. (In Russian).
2. Lavrinenko A. N., Chervyakov N. I. Nomodal Operations Research in System of Residual Classes // Belgorod State University. Scientific Bulletin. Series: Economics. Information Technologies. 2012. No. 1 (120), Is. 21/1. P. 110–122. (In Russian).
3. Akushsky I. Ya., Yuditsky D. I. Mashinnaiia arifmetika v ostatochnykh klassakh. Moscow : Sov. Radio, 1968. 439 p. (In Russian).
4. Isupov K. S. Methods of Basic Non-Modular Operations in Modular Arithmetic Using Interval Positional Characteristics // University Proceedings. Volga Region. Technical Sciences. 2013. No. 3 (27). P. 26–39. (In Russian).
5. Chervyakov N. I., Averbukh V. M., Babenko M. G. et al. Approximate Method of Implementation Non-Modular Operations in the Residue Number System // Fundamental Research. 2012. No. 6–1. P. 189–193. URL: <https://fundamental-research.ru/ru/article/view?id=29963> (accessed: 20.10.2022). (In Russian).
6. Lyakhov P. A., Grobova T. A., Karasev I. V. Modelirovanie algoritmov obratnogo preobrazovaniia chisel iz sistemy ostatochnykh klassov v pozitsionnuu sistemu schisleniia // Aktual. napravleniia nauch. Issled. XXI veka: teoriia i praktika. 2015. Vol. 3, No. 7–4 (18–4). P. 430–434. (In Russian).
7. Lyakhov P. A., Goloshubova Yu. V., Popova E. A. Sravnitelnyi analiz metodov perevoda chisel iz sistemy ostatochnykh klassov v pozitsionnuu sistemu schisleniia // Molodoi uchenyi. 2017. No. 22 (156). P. 1–6. URL: <https://moluch.ru/archive/156/44137/> (accessed: 19.10.2022). (In Russian).

8. Полисский Ю. Д. Сравнение чисел в системе остаточных классов // 50 лет модулярной арифметики : тр. Юбилейн. Междунар. науч.-техн. конф., 23–25 ноября 2005 г., Москва, Зеленоград. М. : МИЭТ, 2005. С. 274–290.
9. Краснобаев В. А., Янко А. С., Кошман С. А. Метод арифметического сравнения данных, представленных в системе остаточных классов // Кибернетика и систем. анализ. 2016. Т. 52, № 34. С. 157–162.
10. Бабенко М. Г., Черных А. Н., Червяков Н. И. и др. Эффективное сравнение чисел в системе остаточных классов на основе позиционной характеристики // Тр. ИСП РАН. 2019. Т. 31, Вып. 2. С. 187–202.
11. Исупов К. С. Об одном алгоритме сравнения чисел в системе остаточных классов // Вестн. Астрахан. гос. техн. ун-та. Сер. Управление, вычисл. техн. информ. 2014. № 3. С. 40–49.
12. Тейтельбаум В. Н. Сравнение чисел в чешской системе счисления // Докл. АН СССР. 1958. Т. 121, № 5. С. 807–810.
13. Копыткова Л. Б. Математические модели нейросетевой реализации модулярных вычислительных структур для высокоскоростной цифровой фильтрации : дис. ... канд. физ.-мат. наук. Ставрополь, 2001. 264 с.
14. Червяков Н. И., Ляхов П. А., Копыткова Л. Б. и др. Обработка информации в системе остаточных классов (СОК). Ставрополь : Север.-Кавказ. федер. ун-т, 2016. 225 с. URL: <https://book.ru/book/928854> (дата обращения: 06.09.2022).
15. Кочеров Ю. Н. Разработка методов и алгоритмов разделения и восстановления данных в модулярных пороговых структурах для распределенных вычислительных сетей : моногр. Ставрополь : Север.-Кавказ. федер. ун-т, 2016. 239 с.
8. Polissky Yu. D. Sravnenie chisel v sisteme ostatochnykh klassov // 50 let moduliarnoi arifmetiki : Proceedings of the Anniversary International Scientific and Engineering Conference, November 23–25, 2005, Moscow, Zelenograd. Moscow : MIET, 2005. P. 274–290. (In Russian).
9. Krasnobaev V. A., Yanko A. S., Koshman S. A. Metod arifmeticheskogo sravneniia dannykh, predstavlennykh v sisteme ostatochnykh klassov // Kibernetika i system. analiz. 2016. Vol. 52, No. 34. P. 157–162. (In Russian).
10. Babenko M. G., Chernykh A. N., Chervyakov N. I. et al. Efficient Number Comparison in the Residue Number System Based on Positional Characteristics // Proceedings of the Institute for System Programming of the RAS. 2019. Vol. 31, Is. 2. P. 187–202. (In Russian).
11. Isupov K. S. On an Algorithm for Number Comparison in the Residue Number System // Vestnik of Astrakhan State Technical University. Series: Management, Computer Science and Informatics. 2014. No. 3. P. 40–49. (In Russian).
12. Teitelbaum V. N. Comparison of Numbers in the Czechic System of Numbers // Doklady Akademii Nauk SSSR. 1958. Vol. 121, No. 5. P. 807–810. (In Russian).
13. Kopytkova L. B. Matematicheskie modeli neirosetevoi realizatsii moduliarnykh vychislitelnykh struktur dlia vysokoskorostnoi tsifrovoi filtratsii : Cand. Sci. Dissertation (Physics and Mathematics). Stavropol, 2001. 264 p. (In Russian).
14. Chervyakov N. I., Lyakhov P. A., Kopytkova L. B. et al. Obrabotka informatsii v sisteme ostatochnykh klassov (SOK). Stavropol : North-Caucasus Federal University, 2016. 225 p. URL: <https://book.ru/book/928854> (accessed: 06.09.2022). (In Russian).
15. Kocherov Yu. N. Razrabotka metodov i algoritmov razdeleniia i vosstanovleniia dannykh v moduliarnykh porogovykh strukturakh dlia raspredeleniia vychislitelnykh setei : Monograph. Stavropol : North-Caucasus Federal University, 2016. 239 p. (In Russian).

Информация об авторе

Н. С. Золотарева – аспирант.

Information about the author

N. S. Zolotareva – Postgraduate.

Сетевое издание. Полные тексты статей размещаются
на официальном сайте издания <https://www.vestcyber.ru> и в базе данных
Научной электронной библиотеки на сайте elibrary.ru, сведения о публикуемых материалах включаются
в Российский индекс научного цитирования (РИНЦ)

Адрес учредителя и издателя:
бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры
«Сургутский государственный университет»,
628412, Ханты-Мансийский автономный округ – Югра, Сургут, пр. Ленина, 1. Тел. (3462) 76-30-67

Дата опубликования 30.12.2022. Формат 60 × 84/8
Уч.-изд. л. 9,1. Цена свободная