

ISSN 2313-7304

ПРОЦЕССЫ УПРАВЛЕНИЯ И УСТОЙЧИВОСТЬ

ТОМ 8 (24)

**ТРУДЫ LI МЕЖДУНАРОДНОЙ НАУЧНОЙ
КОНФЕРЕНЦИИ АСПИРАНТОВ И СТУДЕНТОВ**

Санкт-Петербург

5 – 9 апреля 2021 года

ISSN 2313-7304

***CONTROL PROCESSES
AND STABILITY***

Vol. 8 (24)

**PROCEEDINGS OF THE LII
INTERNATIONAL CONFERENCE**

**St. Petersburg
April 5 – 9, 2021**

PUBLISHING HOUSE
FEDOROVA G.V.
2021

ББК 22.1

П84

Редакционная коллегия:

д-р физ.-мат. наук, проф. Л. А. Петросян (отв. редактор) (СПбГУ),
д-р физ.-мат. наук, проф. Н. В. Смирнов (зам. отв. редактора) (СПбГУ)
д-р физ.-мат. наук, проф. О. И. Дривотин (СПбГУ)
д-р физ.-мат. наук, проф. А. П. Жабко (СПбГУ)
д-р физ.-мат. наук, проф. Е. Д. Котина (СПбГУ)
д-р физ.-мат. наук, проф. А. Ю. Крылатов (СПбГУ)

Процессы управления и устойчивость. Том 8(24). № 1 / науч.
П84 ред. тома Н. В. Смирнов. СПб.: Издательский Дом Федоровой Г.В.,
2021. 538 с.
ISSN 2313-7304

«Процессы управления и устойчивость» – ежегодное научное издание. Издается на факультете прикладной математики – процессов управления СПбГУ с 1998 года. Сборник традиционно формируется по результатам работы ежегодной Международной научной конференции аспирантов и студентов «Процессы управления и устойчивость» и включает статьи студентов, аспирантов, сотрудников СПбГУ и других высших учебных заведений, в том числе зарубежных, по математической теории процессов управления, математическим методам в механике и физике, математическому моделированию в медико-биологических системах, информационным и компьютерным технологиям, теории управления социально-экономическими системами.

В Том 8(24) включены статьи участников 52-й Международной научной конференции аспирантов и студентов «Процессы управления и устойчивость» (5 – 9 апреля 2021 года), прошедшие рецензирование и рекомендованные к публикации. Сборник предназначен для студентов старших курсов физико-математических факультетов университетов, аспирантов и научных работников.

ББК 22.1

*Сборник издается при финансовой поддержке
факультета прикладной математики – процессов управления
Санкт-Петербургского государственного университета*

ISSN 2313-7304

© Издательский Дом Федоровой Г. В., 2021
© Смирнов Н. В., 2021

Содержание

Пленарная сессия	9
Приглашенные лекторы	9
<i>Тамасян Г. Ш.</i> О конструктивном негладком анализе	13
1. Математическая теория процессов управления	37
<i>Алькин Р. В.</i> Децентрализованное управление расположением объектов на прямой с учетом их размера	37
<i>Белов А. И.</i> Критерий экспоненциальной устойчивости для линейной системы с запаздыванием	45
<i>Евтина Д. С.</i> О робастной устойчивости линейных систем с запаздыванием	50
<i>Жигалов В. С.</i> Условия точечной управляемости дифференциально-разностной системы с линейно-возрастающим запаздыванием	55
<i>Жолобов Е. В.</i> Прогнозирование места аварийной посадки квадрокоптера	61
<i>Иванов А. В., Моисеев И. А.</i> Частный случай синтеза оптимального управления в задаче перелета	66
<i>Кудряков Д. А.</i> Анализ устойчивости линейных систем с запаздыванием и неопределенными параметрами	74
<i>Миронов А. С.</i> Обобщение задачи Данскина	79
<i>Kastner A., Gröll L., Hagenmeyer V.</i> Abstractions in modeling of switching DC/AC power converters	85
<i>Wachter J., Gröll L., Hagenmeyer V.</i> Realtime testbench for inverters – a sliding mode control experiment	93
2. Математические методы в механике и физике	101
<i>Алцубеев Г. О., Бурова И. Г.</i> Газотурбинный двигатель и сплайновые приближения	101
<i>Вакаева А. Б., Шувалов Г. М., Костырко С. А.</i> Морфологическая неустойчивость нанопоры в твердом теле при поверхностной диффузии	108
<i>Виноградова Е. М., Доронин Г. Г.</i> Расчет распределения потенциала в двумерной диодной системе с лезвийным полевым катодом	113

<i>Демьянович Ю. К., Бурова И. Г., Доронина А. Г., Мирошниченко И. Д.</i> Об аппроксимации методом конечных элементов в плоской области.....	117
<i>Зубахина Т. С.</i> Анализ устойчивости методов Рунге–Кутты–Чебышёва второго порядка для уравнений с запаздыванием..	126
<i>Казанков В. К., Холодова С. Е.</i> Исследование и анализ уравнения теории идеальной несжимаемой электропроводящей жидкости	131
<i>Онегов Н. А.</i> Анализ влияния внутренних волн на процесс разлива нефти и на основные характеристики флюида в нефтеносном пласте.....	138
<i>Пегливанова М. М.</i> Коррозия пластины из упругопластического материала с SD-эффектом.....	144
<i>Рыбалкин А. А.</i> Математическое моделирование балансирующего робота на сферическом колесе	149
<i>Хоанг В. Н.</i> Дифференциально-разностное уравнение с распределенными параметрами на графе	155
<i>Шувалов Г. М., Костырко С. А.</i> Влияние поверхностной упругости на морфологическую устойчивость поверхности ультратонкого плёночного покрытия.....	161
<i>Schafberger J., Hagenmeyer V., Gröll L.</i> Modeling of a distillation column without measurement data	167
3. Математические модели медико-биологических систем.....	175
<i>Барташук А. Е.</i> Прогнозирование показателя Глисона рака предстательной железы с помощью искусственной нейронной сети	175
<i>Белавин С. А.</i> Использование методов кластерного анализа для диагностирования заболеваний мозга	182
<i>Бучина Д. А., Гончарова А. Б.</i> Обработка данных медицинских исследований с использованием псевдорандомизации (PSM).....	187
<i>Виль М. Ю., Гончарова А. Б.</i> Применение приложения MATLAB SimBiology для моделирования онкологического заболевания с учетом интерференционной конкуренции	192
<i>Ларочкин П. В.</i> Исследование МРТ изображений головного мозга.....	197

<i>Леонова Е. Б., Котина Е. Д.</i> Построение поля перемещений для последовательностей изображений с использованием дискретных систем.....	202
<i>Моисеева В. Д.</i> Анализ одной мультивирусной эпидемической модели.....	207
<i>Петрова О. Г.</i> Сегментация МРТ изображений позвоночника с использованием сверточных нейронных сетей.....	214
<i>Тихонова Е. О.</i> Статистический анализ изменения внутриглазного давления после интравитреальных инъекций.....	219
<i>Хао Ц.</i> Нахождение проекций квадратичных многообразий...	224
<i>Чубаров Д. А.</i> Математическое моделирование межпозвоночного диска с использованием численного метода.....	233
4. Информационные и компьютерные технологии.....	239
<i>Акопян А. С.</i> Улучшение методов получения выравнивания слов в задаче нейронного машинного перевода.....	239
<i>Афанасов А. К., Цой А. И.</i> Извлечение данных Android-приложения WhatsApp.....	246
<i>Гладышев Т. Ю.</i> Реконструкция трехмерных объектов из однарных изображений при помощи глубинных нейронных сетей.....	254
<i>Грашин Д. М.</i> Определение размера текстуры для реконструированной модели в приложении Agisoft MetaShape.....	259
<i>Жереб В. В.</i> Реализация алгоритмов RAID-вычислений для процессора ARM.....	265
<i>Ильченко И. И.</i> Реализация сетевого пространства имен в ОС Embox.....	274
<i>Кванччани К. Ш.</i> Повышение качества медицинских изображений в контексте задачи сегментации опухолей.....	279
<i>Князев Н. А.</i> Определение пола и возраста людей по фотографии лица с использованием нейронных сетей.....	284
<i>Кульминский Д. П., Ефимов В. В.</i> Semi-supervised сегментация с использованием нейросетей в задаче оценки кондиционности ядра.....	289
<i>Меньшиков М. А.</i> Систематизация абстракций для поддержки различных языков программирования и сред в статических анализаторах кода.....	294
<i>Николаева И. Н.</i> Минимизация размера образов контейнеров приложений на примере Docker.....	303

<i>Павлова Е. Д., Чеботарева А. А., Цепелева Р. В.</i> Решение задачи автоматизированной разметки подводных животных с помощью каскада нейронных сетей.....	308
<i>Руценок В. В.</i> Разработка алгоритма упорядочивания наложенных контуров.....	313
<i>Савченко А. В.</i> Сравнение системы прав доступа на основе промежуточных групп пользователей со стандартной системой прав доступа без промежуточных групп.....	318
<i>Сермягин Н. В.</i> Детектирование людей в видеопотоке с применением детектора движения и сверточной нейронной сети..	321
<i>Сорокина С. О., Михайлова Е. Г.</i> Оптимизация гиперпараметров алгоритмов градиентного бустинга.....	326
<i>Тереженко Д. В.</i> Программный интерфейс для отказоустойчивых распределённых вычислений на Python.....	332
<i>Фараджов А. М.</i> Сравнительный анализ DLT решений на базе алгоритма PBFT.....	337
<i>Цепелева Р. В., Павлова Е. Д., Чеботарева А. А.</i> Имплементация протокола по кроссчейн-взаимодействию.....	342
<i>Bai M., Li S.</i> Comparative study of various edge detection operator.....	347
<i>Kimura K., Tu Y., Tanji R., Mozgovoy M.</i> Identifying winning actions in a 3D tennis game with Monte Carlo tree search.....	352
<i>Li Y., Fan F.</i> Design and implementation of online education system based on B2C mode.....	358
<i>Liu Y., Chang G.</i> Implementation of facial mask detection in tensorflow based on improved SSD algorithm.....	363
<i>Qiao W., Zhang J.</i> Image fusion based on transform domain.....	368
<i>Su S., Blekanov I.</i> Neural network methods for messages analysis in user discussions on YouTube.....	372
<i>Tang Z., Pyshkin E.</i> Managing notifications within the context of soft planning: visual approach to requirement analysis.....	377
<i>Yang W., Li P.</i> Research on sentiment analysis of network public opinion based on BI-LSTM.....	387
<i>Yuda K., Otomo K., Ito R., Wakana I., Tanji R., Kamei S., Mozgovoy M.</i> Behavior similarity between human players and built-in AI characters in universal fighting engine.....	392
<i>Zhao C., Blekanov I.</i> Two towers collaborative filtering algorithm for movie recommendation.....	397

5. Управление социально-экономическими системами..403

<i>Берестюкова В. М.</i> Байесовское равновесие в дуополии Бертраана.....	403
<i>Булгакова М. А.</i> Вектор τ для сетевых игр с попарным взаимодействием	408
<i>Бусел В. Д., Лазарева А. В.</i> Использование различных мер центральности в задаче оптимизации передачи информации в самоорганизующихся сетях	413
<i>Кочеров И. П.</i> Анализ данных по ежедневному заражению КОВИД-19	418
<i>Магнитская Н. Г.</i> Супераддитивное расширение характеристической функции в игре управления инвестициями	424
<i>Маркелова А. Ю., Лаисцева М. Ю., Петросян О. Л.</i> Управление микрогрид с системой хранения энергии и генерацией возобновляемой энергии	430
<i>Нечипорук А. А.</i> Эквивалентность моделей динамики мнений в социальных сетях	435
<i>Никитина А. А.</i> Оценка параметров модели SQAIRD с учетом временных интервалов	440
<i>Папернюк А. О.</i> Применение ALS в задаче коллаборативной фильтрации	450
<i>Савин К. А., Громова Е. В.</i> О свойствах характеристических функций в игре с многосторонними внешними влияниями ...	455
<i>Сачков А. В., Щербаков Г. А.</i> Обсуждение потенциального применения аксиоматической теории выбора в бизнесе	460
<i>Соколова И. С.</i> Реализация алгоритмов решения задач многономенклатурной оптимизации управления запасами с дефицитом и скидками.....	466
<i>Сташков Е. А.</i> Анализ задачи достижения консенсуса для ориентированных графов в дискретном времени	473
<i>Третьякова С. С.</i> О структуре решения задачи оптимального управления объемами производства	478
<i>Чеботарева А. А., Павлова Е. Д., Цепелева Р. В.</i> Линейно-квадратичная дифференциальная игра управления объемами вредных выбросов с функционалом в форме Больца	484
<i>Dan W.</i> Analysis of static game and repeated game in tax collection and management	489

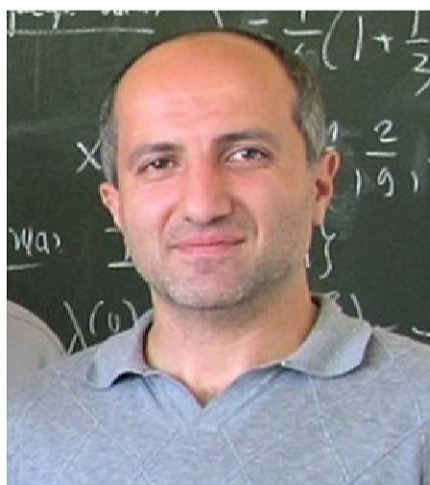
<i>Fernández-Arias E., López-Barrientos J. D., Moreno-Ruíz-Esparza A.</i> A contingent claims approach for computing the premium to ensure an investment in highways	494
<i>Gao J., Parilina E.</i> Average-oriented opinion dynamics with the last moment of observation	505
<i>He Y.</i> Analysis of the most important factors of China's air quality index	510
<i>Qi D.</i> Research on factors affecting differences in investment attractiveness based on HLM model.....	515
<i>Zhu W., Zhou G., Han M.</i> The correlation study of average prices of commodity housing in regional cities in China based on archimedean copula function	520
6. Прикладная математика и процессы управления на факультете ПМ–ПУ СПбГУ	525
<i>Погожев С. В., Смирнов Н. В.</i> Памяти профессора Евгения Игоревича Веремея	525
Contents	528
Именной указатель	534

*Пленарная
сессия*



Тамасян Григорий Шаликович

*кандидат физико-математических наук, доцент,
кафедра математической теории моделирования систем управления,
факультет прикладной математики – процессов управления,
Санкт-Петербургский государственный университет*



О конструктивном негладком анализе

Аннотация. В докладе обсуждается один из разделов современной математики – конструктивный негладкий анализ, в котором изучаются недифференцируемые функции. Одна из первых негладких задач была поставлена и решена П. Л. Чебышевым, где речь шла о поиске алгебраического многочлена наименее уклоняющегося от нуля. Результатом её решения стали широко известные и используемые многочлены Чебышева. В связи с почти двухвековой историей негладкого анализа дана краткая историческая справка создания математического аппарата для исследования недифференцируемых функций конструктивными методами. Описаны элементы исчисления, приведены основные понятия и утверждения негладкого анализа. В докладе представлен обзор некоторых решенных задач, а также приводятся новые актуальные задачи, к решению которых приглашаются все желающие.

Tamasyan Grigoriy Shalikhovich

*PhD, Associate Professor,
Department of Mathematical Modelling of Control System,
Faculty of Applied Mathematics and Control Processes,
Saint Petersburg State University*

About constructive nonsmooth analysis

Abstract. The talk focus on constructive nonsmooth analysis which is one of the branches of modern mathematics that studies nondifferentiable functions. One of the first nonsmooth problems was formulated and solved by P. L. Chebyshev. He was looking for an algebraic polynomial of least deviation from zero. The result of the solution of the problem was famous Chebyshev polynomials which have many applications. The history of nonsmooth analysis goes back for almost two centuries. We give a brief historical overview on the creation of a mathematical apparatus for the study of nondifferentiable functions by constructive methods. We describe the elements of calculus as well as the basic concepts and statements of nonsmooth analysis. The report provides a quick look on some of the solved problems. It also presents new actual ones. We invite everyone to solve them with us.

Балыкина Юлия Ефимовна

*кандидат физико-математических наук, доцент,
кафедра математического моделирования энергетических систем,
факультет прикладной математики – процессов управления,
Санкт-Петербургский государственный университет*



Распространение эпидемий: в поисках новых моделей для практики прогнозирования

Аннотация. В докладе рассматривается проблема построения моделей прогнозирования распространения новых инфекционных заболеваний. Анализ и математическое моделирование вспышек таких заболеваний играют важную роль в планировании ответных мер органов здравоохранения. Базовые математические модели позволяют выявлять основные тенденции при прогнозировании распространения заболеваний. Используя более сложные модели, можно отвечать на вопросы о том, как более эффективно действовать в условиях ограниченных ресурсов системы здравоохранения и выявлять последствия внедрения ограничительных мер на распространение эпидемии. В докладе представлены различные модели прогнозирования распространения эпидемий новых вирусных инфекций, включая как классические модели на основе дифференциальных уравнений, так и современные модели на основе методов машинного обучения, агентного подхода и интеллектуального анализа данных. Приводится классификация моделей и используемых методов. Оценивается влияние входных данных на качество прогнозов. Рассматриваются способы уточнения неизвестных значений параметров математических моделей распространения инфекций. Также в докладе обсуждаются некоторые актуальные задачи данной области исследований.

Balykina Yulia Efimovna

*PhD, Associate Professor,
Department of Mathematical Modelling of Energetic Systems,
Faculty of Applied Mathematics and Control Processes,
Saint Petersburg State University*

**Spread of epidemics: in search
of new models for forecasting practice**

Abstract. We consider the problem of construction and sustainability of cooperative agreements among participants of a conflict-controlled system in the presence of random factors. A method of construction of the cooperative agreements (decisions made in cooperation) is proposed. We also examine the dynamic properties of the agreements allowing to keep cooperation over time. The construction of a mathematical model of the conflict system takes into account random factors affecting the dynamics and payoffs of the participants in the conflict. In the paper, the principles of supporting cooperative agreements are formulated. The implementation of such principles allows to guarantee (in some sense) the presence of cooperation over time. The conditions sufficient for the stability of cooperative solutions in stochastic conflict-controlled systems are obtained. The paper also describes some actual problems in this research area.

УДК 519.6

Тамасян Г. Ш.

О конструктивном негладком анализе

Введение. В работе обсуждается один из разделов современной математики – конструктивный негладкий анализ, в котором изучаются недифференцируемые функции. Приводятся основные понятия и утверждения негладкого анализа. Представлены решения некоторых негладких задач, а также формулируются новые актуальные задачи, к решению которых приглашаются все желающие.

1. Градиент – театр одного актера. Во второй половине XVII века И. Ньютон и Г. Лейбниц независимо друг от друга, ввели понятие *производной*. Они же заложили основы дифференциального и интегрального исчисления, что обеспечивает бурное и неугасаемое развитие науки по сей день.

Напомним, как определяется дифференцируемая функция. Введем следующие обозначения: \mathbb{R}^n – n -мерное евклидово пространство, $\mathbf{0} = (0, \dots, 0) \in \mathbb{R}^n$, $\langle x, y \rangle = x_1 y_1 + \dots + x_n y_n$, $\|x\| = \sqrt{\langle x, x \rangle}$.

Пусть на некоторой (открытой) области $U \subset \mathbb{R}^n$ задана функция f и $x_0 \in U$.

Определение 1. Если существует такой вектор $A \in \mathbb{R}^n$, что

$$f(x_0 + h) = f(x_0) + \langle A, h \rangle + o(\|h\|), \quad \frac{o(\|h\|)}{\|h\|} \rightarrow 0 \text{ при } h \rightarrow \mathbf{0}, \quad (1)$$

то функция f называется *дифференцируемой* в точке x_0 .

Вектор $A = f'(x_0)$ называется *градиентом* функции f в точке x_0 . С помощью градиента можно сформулировать необходимое условие экстремума, которое для полиномов было получено еще П. Ферма.

Теорема 1. *Для того чтобы функция f достигала своего наименьшего или наибольшего значения на множестве U в точке $x_* \in U$ необходимо, чтобы $f'(x_*) = \mathbf{0}$.*

Точка x_* называется *стационарной*.

Тамасян Григорий Шаликович – доцент, Санкт-Петербургский государственный университет, Институт проблем машиноведения РАН; e-mail: g.tamasyan@spbu.ru, тел.: +7(950)028-67-12

Работа выполнена в Институте проблем Машиноведения РАН при финансовой поддержке Российского научного фонда, проект № 20-71-10032

2. Негладкие задачи. Негладкая задача – это задача, которая описывается с помощью недифференцируемых функций. В этом разделе рассмотрим несколько негладких задач, для которых разработаны специфические методы решения, пригодные только для них или узкого класса им подобных.

2.1. Задача Чебышева. Одна из первых существенно негладких задач была поставлена и решена основателем петербургской математической школы Пафнутием Львовичем Чебышевым.

Пусть $n \geq 1$. Обозначим через \mathbb{P}_n множество алгебраических полиномов вида $P(A, t) = t^n + a_1 t^{n-1} + \dots + a_n$. Здесь $A = (a_1, \dots, a_n)$ – вектор коэффициентов из \mathbb{R}^n .

Простейшая задача о наименьшем уклонении от нуля ставится так (см. [1–3]):

$$\varphi(A) := \max_{t \in [-1, 1]} |P(A, t)| \rightarrow \min_{A \in \mathbb{R}^n}. \quad (2)$$

Решение этой задачи существует и единственно. Обозначим его A^* , и пусть $L = \varphi(A^*)$. Полином

$$P_n(t) := P(A^*, t) = \frac{1}{2^{n-1}} \cos(n \arccos(t))$$

называется *полиномом, наименее уклоняющимся от нуля на отрезке* $[-1, 1]$. Заметим, что П. Л. Чебышев решение задачи (2) свел к интегрированию дифференциального уравнения.

Полином $P_n(t)$ характеризуется следующим важным свойством: существуют $(n + 1)$ точки

$$-1 = t_0 < t_1 < \dots < t_n = 1, \quad (3)$$

в которых

$$\begin{aligned} |P_n(t_i)| &= L, \quad i \in 0, \dots, n; \\ P_n(t_i) &= -P_n(t_{i-1}), \quad i \in 1, \dots, n. \end{aligned} \quad (4)$$

Говорят, что точки (3) со свойством (4) образуют полный альтернанс.

Полиномы Чебышева $T_n(t) = \cos(n \arccos(t))$ обладают многими специфическими свойствами и нашли обширные применения как в теории, так и на практике.

2.2. Быстрый поиск нулей ломаной. При решении ряда задач (как гладких, так и негладких) возникает необходимость поиска

нулей у негладких функций. В качестве иллюстрации приведем две гладкие задачи, которые в итоге свелись к поиску нулей у ломаной.

2.2.1. Первая задача. Проблема поиска проекции точки на множество является актуальной и по сей день (см. [4–18]).

Рассмотрим задачу поиска проекции точки $c = (c_1, \dots, c_n)$ на стандартный симплекс $\Lambda \subset \mathbb{R}^n$, где

$$\Lambda = \left\{ x \in \mathbb{R}^n \mid \sum_{i=1}^n x_i = 1; \quad x_i \geq 0, \quad i \in 1, \dots, n \right\}.$$

Задача формализуется так

$$Q(x) := \frac{1}{2} \sum_{i=1}^n (x_i - c_i)^2 \longrightarrow \min_{x \in \Lambda}. \quad (5)$$

Опуская детали (см. [19]), поиск единственного решения сводится к решению скалярного уравнения

$$\sum_{i=1}^n \max\{0, c_i - t\} = 1.$$

Вопросу трудоемкости задачи (5) посвящена работа [20].

2.2.2. Вторая задача. Так называемая *квадратичная задача о ранце* исследовалась во многих работах [21–23]. Однако наиболее существенный прорыв был сделан А. В. Плоткиным [24]. Им предложен, вероятнее всего, самый быстрый алгоритм (из известных на сегодняшний день) решения задачи квадратичного программирования со строго выпуклой сепарабельной целевой функцией, единственным линейным ограничением (типа равенства) и двусторонними ограничениями на переменные. А именно

$$\frac{1}{2} \langle Hx, x \rangle - \langle c, x \rangle \rightarrow \min_{x \in \Omega},$$

$$\Omega : \begin{cases} \sum_{i=1}^n a_i x_i = b, \\ \ell_i \leq x_i \leq r_i, \quad i \in 1, \dots, n. \end{cases}$$

Здесь H – диагональная матрица порядка n с положительными диагональными элементами h_1, \dots, h_n ; $b, c_i, a_i \in \mathbb{R}$, $\ell_i \in \mathbb{R} \cup \{-\infty\}$, $r_i \in \mathbb{R} \cup \{+\infty\}$, $i \in 1, \dots, n$.

И вновь отсылая читателя за деталями к работе [24], укажем, что поиск единственного решения сводится к решению скалярного уравнения (относительно λ) вида

$$\sum_{i=1}^n \left(\ell_i + \frac{1}{h_i} \max\{0, \lambda - h_i \ell_i\} - \frac{1}{h_i} \max\{0, \lambda - h_i r_i\} \right) = b.$$

2.3. Быстрый алгоритм минимизации выпуклой ломаной. Пусть заданы два набора вещественных чисел a_1, \dots, a_m и b_1, \dots, b_m . Рассмотрим следующую экстремальную задачу:

$$f(t) := \sum_{k=1}^m |a_k t - b_k| \longrightarrow \min_{t \in \mathbb{R}}. \quad (6)$$

Известно, что задача (6) может быть сведена к задаче линейного программирования (см. [25, с. 13], [26]). Однако с учетом специфики задачи в [27] показывается, что решением является взвешенная медиана (см. [28]), на поиск которой уходит линейное время (см. [29]).

Действительно, пусть все a_k положительные, тогда узлами ломаной $f(t)$ будут точки b_k/a_k , $k \in 1, \dots, m$. В целевой функции сгруппируем слагаемые с одинаковыми узлами. Пусть ξ_1, \dots, ξ_s – попарно различные узлы ломаной $f(t)$. Тогда задача (6) эквивалентна следующей

$$\frac{1}{A} \sum_{k=1}^s w_k |t - \xi_k| \longrightarrow \min_{t \in \mathbb{R}}, \quad (7)$$

где $w_k = \sum_{j: b_j/a_j = \xi_k} a_j > 0$, $k \in 1, \dots, s$, $A = w_1 + \dots + w_s = a_1 + \dots + a_m$.

Напомним понятие взвешенной медианы [28, с. 255].

Определение 2. Пусть имеется конечный набор различных вещественных чисел b_1, \dots, b_s , для которых заданы положительные веса $\hat{w}_1, \dots, \hat{w}_s$, такие, что $\hat{w}_1 + \dots + \hat{w}_s = 1$. *Взвешенной (нижней) медианой* называется элемент b_ℓ , удовлетворяющий неравенствам

$$\sum_{k: b_k < b_\ell} \hat{w}_k < \frac{1}{2} \quad \text{и} \quad \sum_{k: b_k > b_\ell} \hat{w}_k \leq \frac{1}{2}.$$

Сформулируем основной результат раздела.

Теорема 2. *Взвешенная медиана всех различных узлов ξ_1, \dots, ξ_s с весами $\frac{w_1}{A}, \dots, \frac{w_s}{A}$ доставляет глобальный минимум в задаче (7).*

3. Производная по направлению. Основным понятием, связывающим классический (гладкий) анализ с конструктивным негладким анализом, о котором пойдет речь чуть позже, является производная по направлению.

Положим $S = \{g \in \mathbb{R}^n \mid \|g\| = 1\}$.

Определение 3. Говорят, что функция f дифференцируема в точке x_0 по направлению g , если существует конечный предел

$$f'(x_0, g) = \lim_{\alpha \rightarrow +0} \frac{f(x_0 + \alpha g) - f(x_0)}{\alpha}.$$

Величина $f'(x_0, g)$ называется *производной функции f в точке x_0 по направлению g* .

Определение 4. Если

$$f'(x_0, g) \geq 0 \quad \forall g \in S, \quad (8)$$

то x_0 называется *стационарной точкой* функции $f(x)$.

Определение 5. Пусть точка x_0 не является стационарной. Тогда вектор $g_0 \in S$, на котором

$$f'(x_0, g_0) = \min_{g \in S} f'(x_0, g),$$

называется *направлением наискорейшего спуска* функции f в точке x_0 .

В случае дифференцируемой функции из (1) следует, что

$$f'(x_0, g) = \langle f'(x_0), g \rangle.$$

Если точка x_0 не является стационарной, т. е. $f'(x_0) \neq \mathbf{0}$, то направлением наискорейшего спуска является вектор $g_0 = -\frac{f'(x_0)}{\|f'(x_0)\|}$.

4. Дифференцируемость по направлениям дискретного максимума. Фундаментальным результатом в теории минимаксных задач [30] является формула производной по направлению для функции максимума, которая была одновременно получена В. Ф. Демьяновым и Дж. М. Данскиным [31, 32].

4.1. Основные понятия. Пусть I – конечное индексное множество, $f_i(x)$ – функции, заданные на всем \mathbb{R}^n , $i \in I$. Рассмотрим функцию дискретного максимума

$$\varphi(x) = \max_{i \in I} f_i(x).$$

При фиксированном $x_0 \in \mathbb{R}^n$ рассмотрим индексное множество:

$$I(x_0) = \{i \in I \mid f_i(x_0) = \varphi(x_0)\}.$$

Теорема 3. *Если все функции $f_i(x)$, $i \in I$, непрерывно дифференцируемы в окрестности точки x_0 , то производная функции максимума $\varphi(x)$ в точке x_0 по любому направлению $g \in S$ существует. При этом*

$$\varphi'(x_0, g) = \max_{i \in I(x_0)} \langle f'_i(x_0), g \rangle. \quad (9)$$

При тех же условиях, что и в теореме, справедлив более общий результат, а именно разложение

$$\varphi(x_0 + h) = \varphi(x_0) + \max_{i \in I(x_0)} \langle f'_i(x_0), h \rangle + o(\|h\|),$$

где $o(\|h\|)/\|h\| \rightarrow 0$ при $h \rightarrow \mathbf{0}$.

Обозначим через $\underline{\partial}\varphi(x_0)$ выпуклую оболочку градиентов $f'_i(x_0)$, $i \in I(x_0)$. Нетрудно проверить [33], что

$$\max_{i \in I(x_0)} \langle f'_i(x_0), g \rangle = \max_{v \in \underline{\partial}\varphi(x_0)} \langle v, g \rangle.$$

Тогда формула (9) примет вид

$$\varphi'(x_0, g) = \max_{v \in \underline{\partial}\varphi(x_0)} \langle v, g \rangle. \quad (10)$$

Таким образом, имеем современное определение субдифференцируемой функции [34,35]. А именно, некоторая функция $\varphi(x)$ является субдифференцируемой в точке x_0 , если существует выпуклый компакт $\underline{\partial}\varphi(x_0)$ и производная по направлениям вычисляется по формуле (10).

Класс субдифференцируемых функций замкнут относительно операций сложения, умножения на положительную константу и взятия поточечного максимума от субдифференцируемых функций. Имеется хорошо разработанное исчисление [34,35].

Замечание 1. В действительности, на сегодняшний день известно около двух десятков различных определений субдифференциала [34,36]: Кларка, Фреше, Гато, Мишеля – Пено, Мордуховича, Иоффе, Трэймана, Джеякумара – Люка и др.

4.2. Условие оптимальности. Направление наискорейшего спуска. Пусть функция $\varphi(x)$ субдифференцируема в точке x_0 .

Утверждение 1. Условие стационарности (см. (8)), т. е.

$$\varphi'(x_0, g) = \max_{v \in \partial\varphi(x_0)} \langle v, g \rangle \geq 0 \quad \forall g \in S,$$

равносильно включению $\mathbf{0} \in \partial\varphi(x_0)$.

Следующая теорема также принадлежит В. Ф. Демьянову.

Теорема 4 [31]. Пусть x_0 нестационарная точка. Тогда направлением наискорейшего спуска функции $\varphi(x)$ в точке x_0 является вектор $g_0 = -v_*/\|v_*\|$, где v_* – точка многогранника $\partial\varphi(x_0)$, ближайшая к началу координат.

Один из наиболее популярных методов поиска направления наискорейшего спуска является МДМ-метод, предложенный В. Ф. Митчеллом, В. Ф. Демьяновым и В. Н. Малоземовым (см. [4, 14]).

5. Квазидифференцируемые функции. Пусть функция f определена на открытом множестве U в пространстве \mathbb{R}^n и имеет производную по направлениям в точке $x_0 \in U$. Наряду с субдифференцируемыми функциями можно рассматривать *супердифференцируемые функции*, т. е. функции, у которых производная по направлениям вычисляется по формуле

$$f'(x_0, g) = \min_{w \in \bar{\partial}f(x_0)} \langle w, g \rangle \quad \forall g \in S,$$

где $\bar{\partial}f(x_0) \subset \mathbb{R}^n$ – выпуклый компакт. Понятно, что функция f супердифференцируема в том и только том случае, когда функция $(-1)f$ субдифференцируема.

В общем случае, разность, частное и произведение субдифференцируемых функций уже не являются субдифференцируемыми функциями (в смысле (10)). Аналогичная ситуация и с супердифференцируемыми функциями. Таким образом, множество субдифференцируемых (супердифференцируемых) функций не образуют линейного пространства. Это обстоятельство сподвигло к рассмотрению совокупности функций, у которых производная по направлениям представима в виде

$$f'(x_0, g) = \max_{v \in \partial f(x_0)} \langle v, g \rangle + \min_{w \in \bar{\partial}f(x_0)} \langle w, g \rangle. \quad (11)$$

Определение 6. Функция f , определенная на U , называется *квазидифференцируемой* в точке $x_0 \in U$, если она дифференцируема по направлениям в этой точке и $f'(x_0, g)$ может быть представлена в виде (11). Пара выпуклых компактов $[\underline{\partial}f(x_0), \bar{\partial}f(x_0)]$ называется *квазидифференциалом* функции f в точке x_0 .

Класс квазидифференцируемых функций достаточно богат. В частности, он содержит в себе гладкие функции, выпуклые и вогнутые, суб- и супердифференцируемые функции. А также является замкнутым относительно всех алгебраических операций (сложения, вычитания, деления, умножения, возведения в степень), взятия точечного максимума и минимума и, что очень важно, любой их суперпозиции. Для квазидифференцируемых функций разработано конструктивное исчисление, которое может быть использовано и для создания программного обеспечения (см. [37, 38]).

Стоит отметить, что квазидифференцируемая функция f в точке x_0 допускает представление

$$f(x_0 + h) - f(x_0) = \max_{v \in \underline{\partial}f(x_0)} \langle v, h \rangle + \min_{w \in \bar{\partial}f(x_0)} \langle w, h \rangle + o(\|h\|). \quad (12)$$

А также, что пара выпуклых компактов $[\underline{\partial}f(x_0), \bar{\partial}f(x_0)]$, участвующая в представлении (12), определяется неоднозначно [39].

Замечание 2. Множеству квазидифференцируемых функций принадлежат и d.c. функции (difference of convex functions), т. е. имеющих представление, как разность двух выпуклых функций [40, 41].

Определение 7. Точка x_0 , в которой производные по всем направлениям неотрицательны,

$$f'(x_0, g) \geq 0 \quad \forall g \in S, \quad (13)$$

называется *inf-стационарной* точкой квазидифференцируемой функции f .

Теорема 5. Условие *inf-стационарности* (13) равносильно *вложению*

$$-\bar{\partial}f(x_0) \subset \underline{\partial}f(x_0). \quad (14)$$

Условие оптимальности в форме (14) было получено Л. Н. Поляковой [42].

Пусть точка x_0 не является inf-стационарной для квазидифференцируемой функции $f(x)$. Тогда направление наискорейшего спуска функции f из точки x_0 определяется формулой

$$g_0 = -\frac{v_0 + w_0}{\|v_0 + w_0\|},$$

где w_0 – решение экстремальной задачи

$$\min_{v \in \underline{\partial}f(x_0)} \|v + w\| \longrightarrow \max_{w \in \bar{\partial}f(x_0)}$$

и v_0 – решение экстремальной задачи

$$\|v + w_0\| \longrightarrow \min_{v \in \underline{\partial}f(x_0)}.$$

Замечание 3. Помимо основных учебников по негладкому анализу [30, 34, 35, 43–45] имеются работы, в которых более доступно излагаются базовые понятия [46–49].

6. Кодифференцируемые функции. На сегодняшний день «вершиной» конструктивного негладкого анализа и недифференцируемой оптимизации является введенное В. Ф. Демьяновым и А. М. Рубиновым понятие кодифференцируемой функции [34].

Определение 8. Функция $f(x)$ называется *кодифференцируемой* в точке x , если существуют выпуклые компакты $\underline{d}f(x)$ и $\bar{d}f(x)$ из $\mathbb{R} \times \mathbb{R}^n$ такие, что приращение функции f допускает представление

$$\begin{aligned} f(x+h) - f(x) &= \\ &= \max_{(a,v) \in \underline{d}f(x)} [a + \langle v, h \rangle] + \min_{(b,w) \in \bar{d}f(x)} [b + \langle w, h \rangle] + o(\|h\|), \end{aligned} \quad (15)$$

где $o(\|h\|)/\|h\| \longrightarrow 0$ при $h \longrightarrow 0$.

Пара множеств $[\underline{d}f(x), \bar{d}f(x)]$ называется *кодифференциалом* функции f в точке x и обозначается символом $Df(x)$. Множества $\underline{d}f(x)$ и $\bar{d}f(x)$ называются *гиподифференциалом* и *гипердифференциалом* функции f в точке x . Элементы гиподифференциала называются *гипоградиентами*, элементы гипердифференциала – *гиперградиентами*.

Утверждение 2. Множество кодифференцируемых в точке x функций совпадает с множеством квазидифференцируемых в той же точке функций.

В работе [50] приводится доказательство этого утверждения в случае, когда квазидифференциальное и кодифференциальное множества являются произвольными выпуклыми компактами.

Замечание 4. Хотя класс кодифференцируемых функций совпадает с классом квазидифференцируемых функций, из него можно выделить подмножество *непрерывно кодифференцируемых* функций, в то время как в общем случае, квазидифференциальное отображение разрывное (в метрике Хаусдорфа).

Подробнее опишем это принципиально важное свойство кодифференцируемых функций.

Пусть A и B – пара выпуклых компактов в \mathbb{R}^n . Под расстоянием между компактами A и B будем принимать величину (метрика Хаусдорфа):

$$\rho_H(A, B) = \max \left\{ \max_{a \in A} \min_{b \in B} \|a - b\|, \max_{b \in B} \min_{a \in A} \|a - b\| \right\}.$$

Пусть $\mathcal{B}_\delta(x)$ – шар в \mathbb{R}^n радиуса δ с центром в точке x .

Определение 9. Отображение ψ , определенное на множестве U и имеющее компактные образы, называется непрерывным в точке $x \in U$, если для любого $\varepsilon > 0$ найдется такое $\delta > 0$, что для всех $x' \in \mathcal{B}_\delta(x)$ выполняется неравенство $\rho_H(\psi(x), \psi(x')) < \varepsilon$.

Определение 10. Функция f называется *непрерывно кодифференцируемой* в точке x , если она кодифференцируема в некоторой окрестности точки x и если существует непрерывное (по Хаусдорфу) в этой точке кодифференциальное отображение Df .

Еще раз обратим внимание, что класс непрерывно кодифференцируемых функций включает в себя суперпозиции непрерывно дифференцируемых функций и функций взятия поточечных минимума и максимума.

Для непрерывно кодифференцируемой функции f аппроксимация ее приращения, т.е. правая часть выражения (15) является *непрерывной функцией по x и h* , что, в некотором смысле, уподобляет непрерывно кодифференцируемые функции с непрерывно дифференцируемыми, а это *существенное* свойство для численных методов.

На следующих примерах продемонстрируем элементы кодифференциального исчисления и непрерывно кодифференцируемые функции [34, 38].

Пример 1. Пусть f непрерывно дифференцируема в окрестности точки $x \in U \subset \mathbb{R}^n$. Тогда справедливо представление (см. (1))

$$f(x+h) = f(x) + \langle f'(x), h \rangle + o_x(h),$$

где $\frac{o_x(h)}{\|h\|} \xrightarrow{\|h\| \rightarrow 0} 0$, $f'(x)$ – градиент функции f в точке x . Заметим, что функция f непрерывно кодифференцируема в окрестности точки x равномерно по направлениям и в качестве кодифференциала можно взять $Df(x) = [\underline{d}f(x), \bar{d}f(x)]$, где

$$\underline{d}f(x) = \left\{ \left(\begin{smallmatrix} 0 \\ f'(x) \end{smallmatrix} \right) \right\}, \quad \bar{d}f(x) = \left\{ \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\}.$$

Отображение $Df(x)$ непрерывно.

Пример 2. Рассмотрим функцию $f(x) = |x|$, $x \in \mathbb{R}$, которую можно представить в виде $f(x) = \max\{x, -x\}$. Таким образом, функция $f(x)$ является суперпозицией непрерывно дифференцируемых функций и функции взятия поточечного максимума. При этом $f(x)$ недифференцируема в точке $x_0 = 0$.

Пусть $f(x) = \max\{\varphi_1(x), \varphi_2(x)\}$, где $\varphi_1(x) = x$ и $\varphi_2(x) = -x$. Из примера 1 следует, что

$$D\varphi_1(x) = \left\{ \left(\begin{smallmatrix} 0 \\ 1 \end{smallmatrix} \right) \right\}, \left\{ \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\}, \quad D\varphi_2(x) = \left\{ \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\}, \left\{ \left(\begin{smallmatrix} 0 \\ -1 \end{smallmatrix} \right) \right\}.$$

Кодифференциал для функции максимума $f(x) = \max_{i \in I} \{\varphi_i(x)\}$ имеет вид $Df(x) = [\underline{d}f(x), \bar{d}f(x)]$, где

$$\underline{d}f(x) = \text{co} \left\{ \underline{d}\varphi_k(x) - \sum_{i \in I \setminus \{k\}} \bar{d}\varphi_i(x) + \left\{ \left(\begin{smallmatrix} \varphi_k(x) - f(x) \\ 0 \end{smallmatrix} \right) \right\} \mid k \in I \right\},$$

$$\bar{d}f(x) = \sum_{i \in I} \bar{d}\varphi_i(x).$$

Воспользуемся приведенными формулами при $I = 1, 2$. Вычислим гипо- и гипердифференциал функции f в точке $x_0 = 0$:

$$\underline{d}f(x_0) = \text{co} \left\{ \left(\begin{smallmatrix} 0 \\ 1 \end{smallmatrix} \right) - \left(\begin{smallmatrix} 0 \\ -1 \end{smallmatrix} \right) + \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right), \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) - \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) + \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\} = \text{co} \left\{ \left(\begin{smallmatrix} 0 \\ 2 \end{smallmatrix} \right), \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\},$$

$$\bar{d}f(x_0) = \left\{ \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) + \left(\begin{smallmatrix} 0 \\ -1 \end{smallmatrix} \right) \right\} = \left\{ \left(\begin{smallmatrix} 0 \\ -1 \end{smallmatrix} \right) \right\}.$$

Напомним, что кодифференциал в точке определяется не единственным образом. Например, $Df(x_0)$ можно задать и в таком виде:

$$Df(x_0) = [\text{co} \left\{ \left(\begin{smallmatrix} 0 \\ 1 \end{smallmatrix} \right), \left(\begin{smallmatrix} 0 \\ -1 \end{smallmatrix} \right) \right\}, \left\{ \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix} \right) \right\}].$$

В связи с тем, что кодифференциал в точке определяется не единственным образом вводится понятие эквивалентных пар.

Определение 11. Упорядоченные пары выпуклых компактов $[A, B]$ и $[C, D]$ называются *эквивалентными*, если $A - D = C - B$.

В терминах метрики Хаусдорфа эквивалентность пары выпуклых компактов $[A, B]$ и $[C, D]$ выглядит так:

$$\rho_H(A - D, C - B) = 0.$$

Следующие примеры иллюстрируют разрывность квазидифференциального отображения и непрерывность кодифференциального отображения.

Пример 3. Функция $f(x) = |x|$ (из примера 2) квазидифференцируема (даже субдифференцируема) на \mathbb{R} , причем в качестве квазидифференциала можно взять $\mathcal{D}f(x) = [\underline{\partial}f(x), \bar{\partial}f(x)]$, где

$$\underline{\partial}f(x) = \begin{cases} \{1\}, & x > 0, \\ \{-1\}, & x < 0, \\ \text{co}\{-1, 1\}, & x = 0, \end{cases} \quad \bar{\partial}f(x) = \{0\}.$$

Покажем, что квазидифференциальное отображение $\mathcal{D}f$ в точке $x_0 = 0$ терпит разрыв.

Для этого достаточно рассмотреть случай, когда x стремится к 0 справа. Имеем $\underline{\partial}f(x_0) = \text{co}\{-1, 1\}$, $\underline{\partial}f(x) = \{1\}$. Вычислим

$$\rho_H(\underline{\partial}f(x_0), \underline{\partial}f(x)) = \max \left\{ \max \left\{ \min_{\beta \in \underline{\partial}f(x)} \|-1 - \beta\|, \min_{\beta \in \underline{\partial}f(x_0)} \|1 - \beta\| \right\}, \right. \\ \left. \max \left\{ \min_{\alpha \in \underline{\partial}f(x_0)} \|\alpha - 1\| \right\} \right\} = \max \{ \max\{2, 0\}, 0 \} = 2.$$

Это означает разрыв квазидифференциального отображения $\mathcal{D}f(x)$ в точке $x_0 = 0$.

Пример 4. На примере функции $f(x) = |x|$ проиллюстрируем непрерывность кодифференциального отображения.

Для любого x имеем

$$\underline{d}f(x) = \text{co} \left\{ \begin{pmatrix} x - |x| \\ 2 \end{pmatrix}, \begin{pmatrix} -x - |x| \\ 0 \end{pmatrix} \right\}, \\ \bar{d}f(x) = \left\{ \begin{pmatrix} 0 \\ -1 \end{pmatrix} \right\}.$$

Зафиксируем $\varepsilon > 0$. Пусть $x_0 = 0$, $x_k \neq 0$. Положим $A = \underline{d}f(x_0)$, $B = \overline{d}f(x_0)$, $C = \underline{d}f(x_k)$ и $D = \overline{d}f(x_k)$. Покажем, что пары $[A, B]$ и $[C, D]$ эквивалентны при $x_k \rightarrow x_0$.

Вычислим

$$W = A - D = \text{co} \left\{ \begin{pmatrix} 0 \\ 3 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\},$$

$$V = C - B = \text{co} \left\{ \begin{pmatrix} x_k \\ 3 \end{pmatrix}, \begin{pmatrix} -x_k \\ 1 \end{pmatrix} \right\}.$$

Выделим из V два множества:

$$V_+ = C - B = \text{co} \left\{ \begin{pmatrix} 0 \\ 3 \end{pmatrix}, \begin{pmatrix} -2x_k \\ 1 \end{pmatrix} \right\}, \quad \text{если } x_k > 0,$$

$$V_- = C - B = \text{co} \left\{ \begin{pmatrix} 2x_k \\ 3 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\}, \quad \text{если } x_k < 0.$$

На рис. 1 иллюстрируется геометрическая интерпретация, как V_+ стремится к W при $x_k \rightarrow +0$ и V_- стремится к W при $x_k \rightarrow -0$.

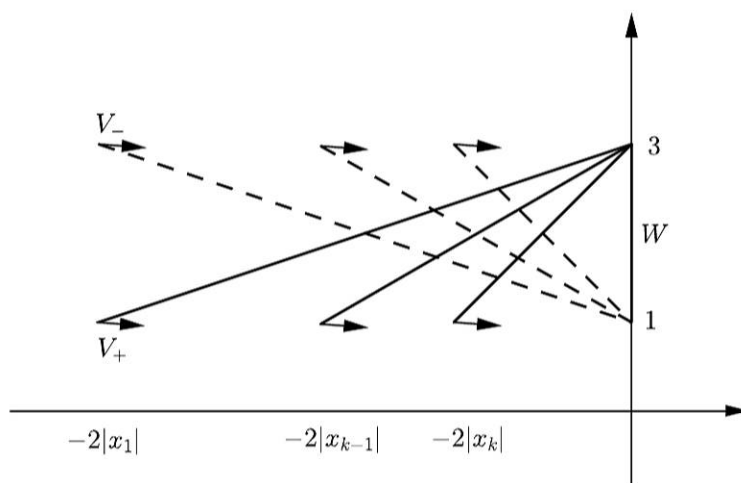


Рис. 1. Стремление V к W при $|x_k| \rightarrow 0$

Вычислим расстояние Хаусдорфа для случая $x_k > 0$:

$$\begin{aligned} \rho_H(W, V_+) &= \max \left\{ \max \left\{ \min_{\beta \in V_+} \left\| \begin{pmatrix} 0 \\ 3 \end{pmatrix} - \beta \right\|, \min_{\beta \in V_+} \left\| \begin{pmatrix} 0 \\ 1 \end{pmatrix} - \beta \right\| \right\}, \right. \\ &\quad \left. \max \left\{ \min_{\alpha \in W} \left\| \alpha - \begin{pmatrix} 0 \\ 3 \end{pmatrix} \right\|, \min_{\alpha \in W} \left\| \alpha - \begin{pmatrix} -2x_k \\ 1 \end{pmatrix} \right\| \right\} \right\} = \\ &= \max \left\{ \max \left\{ 0, \frac{2x_k}{\sqrt{x_k^2 + 1}} \right\}, \max \{0, 2x_k\} \right\} = \max \left\{ \frac{2x_k}{\sqrt{x_k^2 + 1}}, 2x_k \right\} = 2x_k. \end{aligned}$$

Аналогично вычисляем $\rho_H(W, V_-) = -2x_k$ для $x_k < 0$. Тогда для любого $0 < |x_k| < \varepsilon/2$ выполняется $\rho_H(W, V) < \varepsilon$. Непрерывность по Хаусдорфу показана.

Замечание 5. За рамками статьи остались иные инструменты по исследованию негладких задач [51–54].

7. Открытые проблемы недифференцируемой оптимизации. В этом разделе приведем постановки нескольких задач, имеющих непосредственное отношение к конечномерному конструктивному негладкому анализу. А именно, рассмотрим задачи вычислительной геометрии, глобальной оптимизации и интервального анализа.

7.1. Выпуклая оболочка. Ранее рассмотренные обобщения производной (субдифференциал, квази- и кодифференциал) имеют выпуклую структуру. В большинстве практических задач они принимают вид выпуклых многогранников, а точнее выпуклой оболочки некоторого набора точек.

Как известно, для полноценного описания выпуклого многогранника достаточно знания его крайних точек [55, 56]. В настоящее время наиболее популярный алгоритм нахождения выпуклой оболочки является QuickHull [57]. В частности, этот алгоритм можно встретить в пакетах MATLAB и Maple. Однако, у него имеются два основных недостатка – это трудоемкость и ограниченность в размерности. Точнее, для размерности более 10 авторы алгоритма не гарантируют корректность результата.

Проблема 1. Разработать новые эффективные алгоритмы по выявлению крайних точек.

7.2. Поиск проекции точки на выпуклое множество. В пункте 2 актуальность этой задачи уже упоминалась. Очевидно, что поиск проекции точки на многогранник легко сводится к задаче поиска проекции начала координат.

Проблема 2. Пусть задан политоп V , т. е. выпуклая оболочка точек V_1, \dots, V_s . Разработать новые эффективные алгоритмы по решению следующей экстремальной задачи: $\|v\| \rightarrow \min_{v \in V}$.

Обратим внимание, что существуют алгоритмы, не требующие знания крайних точек множества [14].

7.3. Сложение множеств по Минковскому. Операции сложения и вычитания в кодифференциальном (суб- и квазидифференциальном) исчислении одни из тех, которые существенно увеличивают количество элементов в результирующем кодифференциале. Поясним этот момент.

Пусть $[A, B]$ и $[C, D]$ – две пары выпуклых множеств (например, кодифференциалы двух функций). Тогда

$$[A, B] + [C, D] = [A + C, B + D],$$

где оператор «+» обозначает сложение по Минковскому, т. е.

$$A + B = \{a + b \mid a \in A, b \in B\}.$$

Отсюда, налицо колоссальный рост данных. Одна из идей борьбы с этим явлением является использование процедуры QuickHull. Однако более перспективным направлением может стать развитие алгоритмов, предложенных в работе [58].

Проблема 3. Разработка алгоритмов по выявлению крайних точек и «выброса» внутренних точек у множества, образованного в результате сложения двух многогранников (политопов).

7.4. Глобальная оптимизация кусочно-аффинной функции. Класс кусочно-аффинных функций является одним из простейших среди всего многообразия негладких функций. В тоже время они широко используются в методах исследования других функций, а также в практически важных задачах. Поэтому владение эффективными методами поиска экстремума и нулей кусочно-аффинных функции является важной задачей.

Отметим, что в кодифференциальном представлении (15) для кусочно-аффинной функции последнее слагаемое равняется нулю. Имеем так называемый *глобальный* кодифференциал [59].

Проблема 4. Разработка эффективных алгоритмов глобальной оптимизации кусочно-аффинной функции.

7.5. Распознающие функционалы. Данный пункт основан на работах [60, 61]. Рассмотрим интервальную систему линейных алгебраических уравнений

$$\mathbf{A}x = \mathbf{b}.$$

Здесь \mathbf{A} – интервальная $(m \times n)$ -матрица, $x = (x_1, \dots, x_n)^T$, \mathbf{b} – интервальный m -вектор. Данная система понимается как совокупность точечных систем линейных уравнений $Ax = b$, где $A \in \mathbf{A}$, $b \in \mathbf{b}$.

Определение 12. Допусковым множеством решений интервальной системы линейных алгебраических уравнений называется множество

$$\Xi_{tol}(\mathbf{A}, \mathbf{b}) = \{x \in \mathbb{R}^n \mid (\forall A \in \mathbf{A}) (\exists b \in \mathbf{b}) (Ax = b)\}.$$

Далее будем использовать следующие обозначения и понятия:

- интервал: $\mathbf{a} = [\underline{a}, \bar{a}] = \{a \in \mathbb{R} \mid \underline{a} \leq a \leq \bar{a}\}$;
- середина интервала: $\text{mid } \mathbf{a} = \frac{1}{2}(\bar{a} + \underline{a})$;
- радиус интервала: $\text{rad } \mathbf{a} = \frac{1}{2}(\bar{a} - \underline{a})$;
- абсолютное значение (модуль) интервала:

$$|\mathbf{a}| = \max_{a \in \mathbf{a}} |a| = \max\{|\underline{a}|, |\bar{a}|\};$$

- мигнитуда интервала:

$$\langle \mathbf{a} \rangle = \min_{a \in \mathbf{a}} |a| = \begin{cases} \min\{|\underline{a}|, |\bar{a}|\}, & \text{если } 0 \notin \mathbf{a}, \\ 0, & \text{иначе.} \end{cases}$$

Рассмотрим (так называемый *распознающий*) функционал:

$$\text{Tol}(x, \mathbf{A}, \mathbf{b}) = \min_{1 \leq i \leq m} \left\{ \text{rad } \mathbf{b}_i - \left| \text{mid } \mathbf{b}_i - \sum_{j=1}^n a_{ij} x_j \right| \right\}.$$

Теорема 6. *Справедливо*

$$\bar{x} \in \Xi_{tol}(\mathbf{A}, \mathbf{b}) \iff \text{Tol}(\bar{x}, \mathbf{A}, \mathbf{b}) \geq 0.$$

Отсюда следует, что с помощью распознающего функционала можно, в частности, диагностировать пустоту/непустоту допустового множества.

Проблема 5. Разработка эффективных алгоритмов решения максиминной задачи:

$$\text{Tol}(x, \mathbf{A}, \mathbf{b}) \longrightarrow \max_{x \in \mathbb{R}^n}.$$

Отметим, что функционал Tol является ограниченной, вогнутой, кусочно-аффинной на всем пространстве функций. Поэтому для поиска максимума может быть применен метод гипердифференциального подъема.

Сформулируем еще две экстремальных задачи, имеющих отношение к вопросу о разрешимости линейной интервальной системы.

Проблема 6. Разработка эффективных алгоритмов поиска глобального решения для многоэкстремальных функций:

$$\text{Uni}(x, \mathbf{A}, \mathbf{b}) = \min_{1 \leq k \leq m} \left\{ \text{rad } \mathbf{b}_k - \left\langle \text{mid } \mathbf{b}_k - \sum_{j=1}^n \mathbf{a}_{kj} x_j \right\rangle \right\} \longrightarrow \max_{x \in \mathbb{R}^n},$$

$$\text{Uss}(x, \mathbf{A}, \mathbf{b}) = \min_{1 \leq k \leq m} \left\{ \sum_{j=1}^n (\text{rad } \mathbf{a}_{kj}) |x_j| + \text{rad } \mathbf{b}_k - \left| \sum_{j=1}^n (\text{mid } \mathbf{a}_{kj}) x_j - \text{mid } \mathbf{b}_k \right| \right\} \longrightarrow \max_{x \in \mathbb{R}^n}.$$

Отметим, что функционалы Uni и Uss являются кусочно-аффинными на всем пространстве функциями. Поэтому для поиска максимума могут быть применены метод кодифференциального подъема и его модификации [59, 62, 63].

В заключении отметим, что за рамками статьи остались многие другие аспекты недифференцируемой оптимизации, а также вопросы приложения инструментов негладкого анализа в решении задач вариационного исчисления и теории управления [64–76].

Литература

1. Чебышев П. Л. Вопросы о наименьших величинах, связанных с приближенным представлением функций. Полн. собр. соч. Т. 2. М.–Л.: Изд. АН СССР, 1947. 520 с.
2. Научное наследие П. Л. Чебышева. Выпуск первый: Математика. М.–Л.: Изд-во АН СССР, 1945. 175 с.
3. Данилов Ю. А. Многочлены Чебышева. Минск: Выш. шк., 1984. 157 с.
4. Митчелл Б. Ф., Демьянов В. Ф., Малоземов В. Н. Нахождение ближайшей к началу координат точки многогранника // Вестник ЛГУ. 1971. № 19. С. 38–45.
5. Held M., Wolfe P., Crowder H. P. Validation of the subgradient optimization // Math. Programming. 1974. Vol. 6. P. 62–88.
6. Maculan N., de Paula Jr. G. G. A linear-time median-finding algorithm for projecting a vector on the simplex of \mathbb{R}^n // Operations Research Letters. 1989. Vol. 8. No 4. P. 219–222.
7. Lin A., Han S. P. On the distance between two ellipsoids // SIAM J. Optim. 2002. Vol. 13. P. 298–308.
8. Michelot C. A finite algorithm for finding the projection of a point onto the canonical simplex of \mathbb{R}^n // JOTA. 1986. Vol. 50. No 1. P. 195–200.
9. Нурминский Е. А. Проекция на внешне заданные полиэдры // Журн. вычисл. математики и мат. физики. 2008. Т. 48. № 3. С. 387–396.
10. Утешев А. Ю., Яшина М. В. Нахождение расстояния от эллипсоида до плоскости и квадрики в R^n // Докл. АН. 2008. Т. 419. № 4. С. 471–474.
11. Causa A., Raciti F. A purely geometric approach to the problem of computing the projection of a point on a simplex // JOTA. 2013. Vol. 156. No 2. P. 524–528.

12. Зоркальцев В. И. Проекция точки на полиэдр // Журн. вычисл. математики и мат. физики. 2013. Т. 53. № 1. С. 4–19.
13. Tamasyan G., Chumakov A. Finding the distance between the ellipsoid and the intersection of a linear manifold and ellipsoid // Proc. 2015 Int. Conf. “Stability and Control Processes” in Memory of V. I. Zubov (SCP) joined with 21st Int. Workshop on Beam Dynamics and Optimization (BDO). Saint Petersburg, Russia. IEEE, 2015. P. 357–360.
14. Малоземов В. Н. МДМ-методу – 40 лет // Вестн. Сыктывкар. ун-та. Сер. 1. 2012. Вып. 15. С. 51–62.
15. Малоземов В. Н., Певный А. Б. Быстрый алгоритм проектирования точки на симплекс // Вестник СПбГУ. Сер. 1. 1992. № 1. С. 112–113.
16. Tamasyan G., Prosolupov E. Orthogonal projection of a point onto the standard simplex algorithms analysis // Proc. 2015 Int. Conf. “Stability and Control Processes” in Memory of V. I. Zubov (SCP) joined with 21st Int. Workshop on Beam Dynamics and Optimization (BDO). Saint Petersburg, Russia. IEEE, 2015. P. 353–356.
17. Малоземов В. Н., Тамасян Г. Ш. Два быстрых алгоритма проектирования точки на стандартный симплекс // Журн. вычисл. математики и мат. физики. 2016. Т. 56. № 5. С. 742–755.
18. Чумаков А. А., Тамасян Г. Ш. Об одной модификации метода градиентного спуска в задаче проектирования начала координат на эллипсоид // Процессы управления и устойчивость. 2017. Т. 4. № 1. С. 59–63.
19. Тамасян Г. Ш., Просолупов Е. В., Ангелов Т. А. Сравнительный анализ двух быстрых алгоритмов проектирования точки на стандартный симплекс // Дискрет. анализ и исслед. операций. 2016. Т. 23. № 2. С. 100–123.
20. Просолупов Е. В., Тамасян Г. Ш. Оценка трудоемкости алгоритма по поиску нуля одной выпуклой кусочно-линейной функции // Дискрет. анализ и исслед. операций. 2018. Т. 25. № 2. С. 82–100.

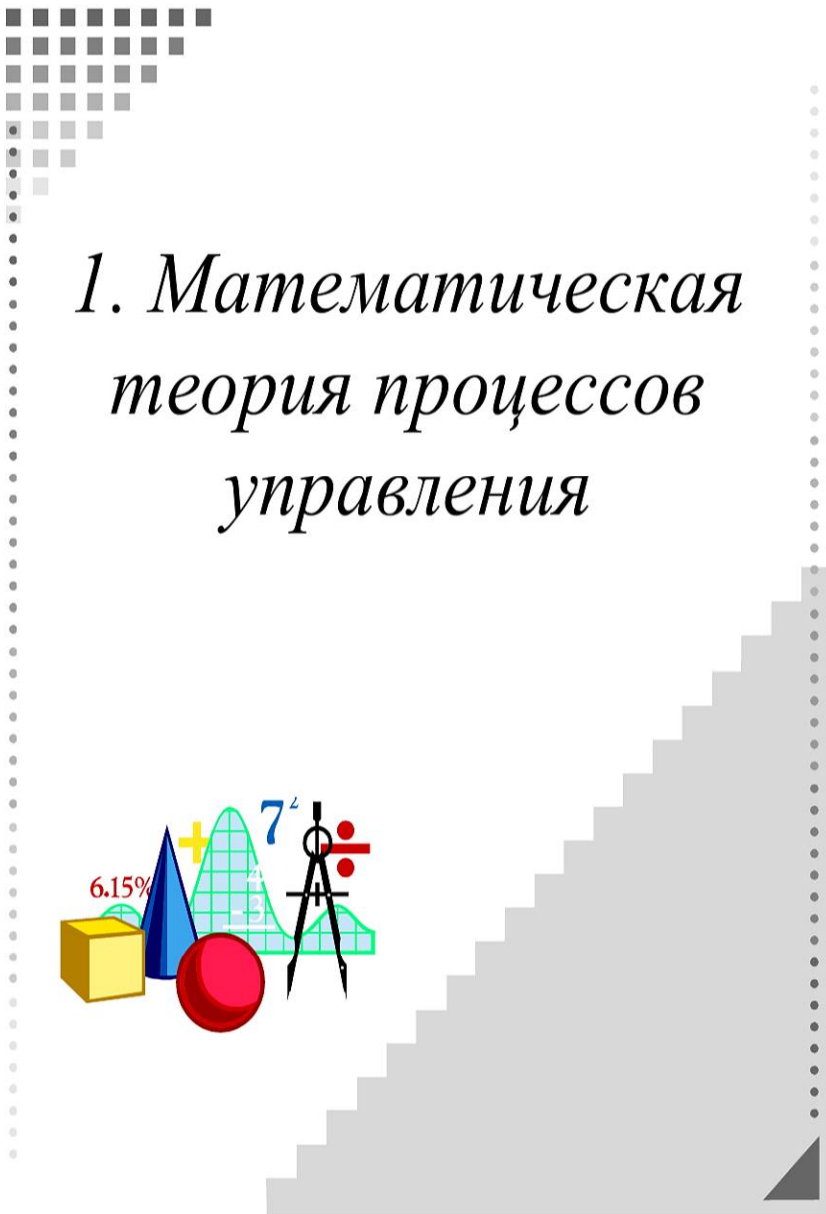
21. Helgason R. V., Kennington J. L., Lall H. A polynomially bounded algorithm for a singly constrained quadratic program // *Mathematical Programming*. 1980. Vol. 18. No 3. P. 338–343.
22. Brucker P. An $O(n)$ algorithm for quadratic knapsack problems // *Operations Research Letters*. 1984. Vol. 3. No 3. P. 163–166.
23. Patriksson M. A survey on the continuous nonlinear resource allocation problem // *Eur. J. Oper. Res.* Feb., 2008. Vol. 185. No 1. P. 1–46.
24. Плоткин А. В. Быстрый алгоритм решения квадратичной задачи о ранце // Семинар “CNSA & NDO”. Избранные доклады. 24 февраля 2021 г. [Электронный ресурс]: URL:<http://www.apmath.spbu.ru/cnsa/rep21.shtml#0224> (дата обращения: 05.05.2021).
25. Гавурин М. К., Малоземов В. Н. Экстремальные задачи с линейными ограничениями. Л.: Изд-во ЛГУ, 1984. 176 с.
26. Шульга Г. С., Удот М. В. Тамасян Г. Ш. О задаче минимизации суммы модулей аффинных функций // *Процессы управления и устойчивость*. 2019. Т. 6. № 1. С. 471–475.
27. Тамасян Г. Ш., Шульга Г. С. Быстрый алгоритм минимизация выпуклой ломаной // Семинар “CNSA & NDO”. Избранные доклады. 24 марта 2021 г. [Электронный ресурс]: URL:<http://www.apmath.spbu.ru/cnsa/rep21.shtml#0324> (дата обращения: 05.05.2021).
28. Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Алгоритмы. Построение и анализ. М.: Вильямс, 2013. 1328 с.
29. Малистов А. С. О поиске медианы массива за линейное время // *Матем. просв. Сер. 3. Т. 21*, М.: МЦНМО, 2017. С. 265–270.
30. Демьянов В. Ф., Малоземов В. Н. Введение в минимакс. М.: Наука, 1972. 368 с.
31. Демьянов В. Ф. К минимизации максимального уклонения // *Вестник Ленинградского университета*. 1966. № 7. С. 21–28.
32. Danskin J. M. The theory of min-max, with applikations // *SIAM J. on Applied Math.* 1966. Vol. 14. P. 641–664.

33. Малоземов В. Н. Некоторые свойства дискретного максимума // Семинар “CNSA & NDO”. Избранные доклады. 14 мая 2015 г. [Электронный ресурс]: URL:<http://www.apmath.spbu.ru/cnsa/refs15.shtml#0514> (дата обращения: 05.05.2021).
34. Демьянов В. Ф., Рубинов А. М. Основы негладкого анализа и квазидифференциальное исчисление. М.: Наука, 1990. 432 с.
35. Демьянов В. Ф., Васильев Л. В. Недифференцируемая оптимизация. М.: Наука, 1981. 384 с.
36. Демьянов В. Ф., Рощина В. А. Обобщенные субдифференциалы и экзостеры // Владикавказский матем. журн. 2006. Т. 8. Вып. 4. С. 19–31.
37. Андрамонов М. Ю., Тамасян Г. Ш. Реализация аналитического кодифференцирования в пакете MATLAB // Вычислительные методы и программирование. 2007. Т. 8. № 2. С. 1–5.
38. Ангелов Т. А. О вычислении кодифференциалов // Вычислительные методы и программирование. 2013. Т. 14. № 1. С. 133–122.
39. Тамасян Г. Ш. Об эквивалентности классов квази- и кодифференцируемых функций // Семинар “CNSA & NDO”. Избранные доклады. 11 апреля 2019 г. [Электронный ресурс]: URL:<http://www.apmath.spbu.ru/cnsa/refs19.shtml#0411> (дата обращения: 05.05.2021).
40. Стрекаловский А. С. Элементы невыпуклой оптимизации. Новосибирск: Наука, 2003. 356 с.
41. Dolgopolik M. V. New global optimality conditions for nonsmooth DC optimization problems // Journal of Global Optimization. 2020. Vol. 76. No 1. P. 25–55.
42. Демьянов В. Ф., Полякова Л. Н. Условия минимума квазидифференцируемой функции на квазидифференцируемом множестве // Журн. вычисл. математики и мат. физики. 1980. Т. 20. № 4. С. 849–846.
43. Рокафеллар Р. Выпуклый анализ. М.: Мир, 1973. 470 с.

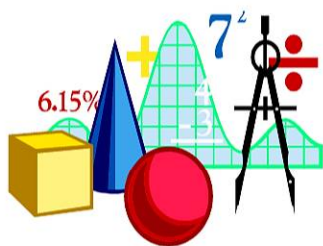
44. Пшеничный Б. Н. Выпуклый анализ и экстремальные задачи. М.: Наука, 1980. 320 с.
45. Шор Н. З. Методы минимизации недифференцируемых функций и их приложения. Киев: Наукова думка, 1979. 200 с.
46. Демьянов В. Ф. Обобщение понятия производной в негладком анализе // Соросовский образовательный журнал. 1996. № 5. С. 121–127.
47. Демьянов В. Ф. Негладкий анализ на плоскости. Часть I // Соросовский образовательный журнал. 1997. № 8. С. 122–128.
48. Демьянов В. Ф. Негладкий анализ на плоскости. Часть II // Соросовский образовательный журнал. 1997. № 9. С. 123–127.
49. Малоземов В. Н., Тамасян Г. Ш. О направлении наискорейшего спуска // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 4. С. 489–501.
50. Dolgopolik M. V. A convergence analysis of the method of codifferential descent // Computational Optimization and Applications. 2018. Vol. 71. No 3. P. 879–913.
51. Аббасов М. Э. Исчисление коэжзостеров второго порядка // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2018. Т. 14. № 4. С. 276–285.
52. Избранные лекции по экстремальным задачам. Часть 1 / под ред. проф. В. Н. Малоземова. СПб.: Изд-во ВВМ, 2017. 470 с.
53. Избранные лекции по экстремальным задачам. Часть 2 / под ред. проф. В. Н. Малоземова. СПб.: Изд-во ВВМ, 2017. 410 с.
54. Малоземов В. Н., Машарский С. М. Элементарные методы в экстремальных задачах. СПб.: Изд-во ВВМ, 2020. 160 с.
55. Ангелов Т. А. Представление кусочно-аффинных функций в виде разности полиэдральных // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2016. Т. 12. № 1. С. 4–18.
56. Препарата Ф., Шеймос М. Вычислительная геометрия: Введение. М.: Мир, 1989. 478 с.

57. Barber C. B., Dobkin D. P., Huhdanpaa H. T. The Quickhull algorithm for convex hulls // ACM Trans. on Mathematical Software. 1996. Vol. 22. P. 469–483. [Электронный ресурс]: URL: <http://www.qhull.org> (дата обращения: 29.05.21).
58. Ангелов Т. А. Нахождение крайних точек суммы двух политопов // Вестник Волгоградского государственного университета. Серия 1: Математика. Физика. 2016. № 6(37). С. 7–17.
59. Dolgopolik M. V. The method of codifferential descent for convex and global piecewise affine optimization // Optimization Methods and Software. 2019. DOI: 10.1080/10556788.2019.1571590.
60. Шарый С. П. Сильная согласованность в задаче восстановления зависимостей при интервальной неопределенности данных // Вычислительные технологии. 2017. Т. 22. № 2. С. 150–172.
61. Шарый С. П., Шарая И. А. Распознавание разрешимости интервальных уравнений и его приложения к анализу данных // Вычислительные технологии. 2013. Т. 18. № 3. С. 80–109.
62. Bagirov A., Karmita N., Makela M. M. Introduction to Nonsmooth Optimization. Theory, Practice and Software. Springer, 2014. 377 p.
63. Bagirov A. M., Gaudioso M., Karmita N., Makela M. M., Taheri S. Numerical Nonsmooth Optimization. State of the Art Algorithms. Springer, 2020. 698 p.
64. Демьянов В. Ф. Условия экстремума и вариационное исчисление. М.: Высшая школа, 2005. 335 с.
65. Долгополик М. В., Тамасян Г. Ш. Об эквивалентности методов наискорейшего и гиподифференциального спусков в некоторых задачах условной оптимизации // Известия Саратовского университета. Новая серия. Серия: Математика. Механика. Информатика. 2014. Т. 14. № 4-2. С. 532–542.
66. Тамасян Г. Ш. Численные методы в задачах вариационного исчисления для функционалов, зависящих от производных высшего порядка // Проблемы матем. анализа. Вып. 67. Новосибирск: Изд-во «Тамара Рожковская», 2012. С. 113–132.
67. Dem'yanov V. F., Tamasyan G. Sh. Exact penalty functions in isoperimetric problems // Optimization: A Journal of Mathematical

- Programming and Operations Research. 2011. Vol. 60. No 1–2. P. 153–177.
68. Демьянов В. Ф., Тамасян Г. Ш. О прямых методах решения вариационных задач // Труды института математики и механики УрО РАН. 2010. Т. 16. № 5. С. 36–47.
 69. Фоминых А. В. Точные штрафы в вариационной задаче с ограничениями в виде дифференциальных неравенств // Процессы управления и устойчивость. 2014. Т. 1. № 1. С. 50–55.
 70. Фоминых А. В., Карелин В. В., Полякова Л. Н., Мышков С. К., Трегубов В. П. Метод кодифференциального спуска в задаче нахождения глобального минимума кусочно-аффинного целевого функционала в линейных системах управления // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2021. Т. 17. № 1. С. 47–58.
 71. Фоминых А. В. Применение метода наискорейшего спуска к решению дифференциальных неравенств // Процессы управления и устойчивость. 2016. Т. 3. № 1. С. 133–137.
 72. Фоминых А. В. О методах субдифференциального и гиподифференциального спуска в задаче построения программного управления с интегральным ограничением на управление // Автоматика и телемеханика. 2017. № 4. С. 37–48.
 73. Фоминых А. В. Задача программного управления объектом, описываемым системой с негладкой правой частью // Журнал вычислительной математики и математической физики. 2019. Т. 59. № 10. С. 1695–1705.
 74. Dolgopolik M. V., Fominyh A. V. Exact penalty functions for optimal control problems I: Main theorem and free-endpoint problems // Optimal Control Applications and Methods. 2019. Vol. 40. No 6. P. 1018–1044.
 75. Abbasov M. E. Comparison between quasidifferentials and exhausters // Journal of Optimization Theory and Applications. Springer US, 2017. Vol. 175. P. 59–75.
 76. Аббасов М. Э. Условия экстремума с ограничениями в терминах собственных и несобственных коэкзостеров // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 2. С. 160–172.



1. Математическая теория процессов управления



УДК 510

Алькин Р. В.

Децентрализованное управление расположением объектов на прямой с учетом их размера

Рекомендовано к публикации доцентом Кирилловым А. Н.

1. Введение. Для решения различных задач, связанных с мобильными роботами, спасательно-розыскными и военными операциями, транспортными сетями, социальными и биологическими системами, часто требуется применение множества однотипных объектов, которыми необходимо децентрализованно управлять. Эти объекты называются точками или агентами. Ключевой работой по разработке алгоритмов группового децентрализованного движения точек с целью их размещения вдоль заданной траектории является статья [1], а также работы [2–5], развивающие модель [1]. В ней рассматривается задача формообразования (спрямления и окруживания) применительно к эмбриологии. Подход, предложенный в [1], в работах [6–9] применяется для решения задачи равномерного размещения агентов с распределенным управлением. Данная задача решается в [9] для точек (агентов), не имеющих размера. При этом допускается совпадение координат объектов в один и тот же момент времени, что на практике невозможно для неточечных объектов, имеющих некоторые размеры. Цель работы состоит в исследовании применимости алгоритма равномерного расположения объектов на отрезке, описанного в [9], с учетом размера объектов, т. е. с учетом их нестолкновения. В данной работе в качестве объектов управления рассматриваются круги одного радиуса. При этом возникает вопрос о нумерации объектов, что не имело значения в алгоритме [9]. Это связано с важностью взаимного расположения «соседних» объектов, т. е. объектов с номерами $i - 1, i, i + 1$. Развитие модели без учета столкновений представлено, например, в [10–12].

2. Постановка задачи. В работе [9] предложен линейный алгоритм, решающий задачу расположения n точек, принадлежащих

Алькин Руслан Валерьевич – аспирант, ФИЦ Карельский научный центр РАН, Институт прикладных математических исследований Карельского НЦ РАН (Петрозаводск); e-mail: ruslan5.ru@yandex.ru, тел.: +7(8142)76-63-12

плоскости, на прямой, на равных расстояниях друг от друга. При этом показано, что если динамика точек задается системой вида

$$\begin{cases} \dot{z}_1 = \dot{z}_n = 0, \\ \dot{z}_i = \frac{z_{i-1} + z_{i+1}}{2} - z_i, \quad i = 2, \dots, n-1, \end{cases} \quad (1)$$

где $z_j = z_j(t) = (x_j(t), y_j(t))$, то задача расположения точек на отрезке $[z_1, z_n]$ будет решена асимптотически. Систему (1) будем называть алгоритмом 1, поскольку расстояния между точками, двигающимися согласно (1), стремятся к равным величинам при $t \rightarrow +\infty$, что дает решение задачи равномерного расположения. При этом дана оценка скорости сходимости.

Суть алгоритма 1 заключается в том, что точка с номером i движется по направлению к середине отрезка, соединяющего точки $i-1$ и $i+1$, $i = 2, \dots, n-1$. Далее, не умаляя общности, будем полагать $z_1(0) = (0, 0)$, $z_n(0) = (L, 0)$, где $0 < L$ – заданная постоянная.

В реальных задачах необходимо учитывать форму и размер объектов. В настоящей работе в качестве объектов рассматриваются круги (на плоскости) радиуса R . Пусть $z_i(t) = (x_i(t), y_i(t))$ – положение центра i -го круга в момент времени t . Будем говорить, что в момент времени t произошло столкновение, если $\rho(z_i(t), z_j(t)) = 2R$, где $\rho^2(z_i, z_j) = (x_i - x_j)^2 + (y_i - y_j)^2$.

Задача состоит в том, чтобы сформулировать необходимые и достаточные условия для множества начальных положений объектов $z_i(0)$, $i = 1, \dots, n$, при которых задача расположения объектов на некотором отрезке, решаемая на основе алгоритма 1, разрешима при условии

$$\rho_{i,j} = \rho(z_i(t), z_j(t)) \geq \varepsilon, \quad (2)$$

где $2R \leq \varepsilon$ – заданная постоянная.

Помимо этого, требуется оценить меру начальных положений (конфигураций объектов), при которых задача разрешима. В такой общей постановке аналитическое решение затруднительно, из-за наличия условия (2). В представленной работе получены некоторые необходимые и достаточные условия на начальные положения, при выполнении которых задача разрешима в одномерном случае, т. е. при $z_i = x_i$. Также произведен численный анализ поставленной задачи. При этом исследуется влияние на ее разрешимость следующих параметров:

- начальное положение объектов;
- нумерация.

Следует отметить, что выбор нумерации объектов – это выбор «соседей», на основании которых строится алгоритм 1, где соседи фиксированы.

3. Решение задачи. Сформулируем условия, при которых задача равномерного расположения объектов разрешима на прямой, а также оценим ограниченность этих условий.

3.1. Условия разрешимости задачи. Пусть объекты расположены на некоторой прямой, т. е. объекты – это отрезки длины $2R$ и $z_i = x_i(t)$, $i = 1, \dots, n$, $x_i(t)$ – середина отрезка, $\forall t \geq 0$. Для разрешимости задачи равномерного расположения необходимо ввести следующую нумерацию объектов:

$$i < j \Leftrightarrow z_i(t) < z_j(t), \quad \forall t \geq 0. \quad (3)$$

Лемма. Движение объектов (т. е. середин z_i отрезков) согласно системе (1) и нумерации (3) в одномерном случае таково, что минимальное расстояние между каждой парой соседних агентов не убывает, если $t_1 \leq t_2$, то $\min_i \rho_{i,i+1}(t_1) \leq \min_i \rho_{i,i+1}(t_2)$.

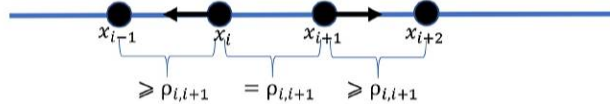


Рис. 1. Отталкивание точек на минимальном расстоянии в любой момент времени t_1

Доказательство. Рассмотрим функцию $\delta(t) = \min_i \rho_{i,i+1}(t)$, $i = 1, \dots, n - 1$. Эта функция дифференцируема по t на интервале $(0, \infty)$ в каждой точке, кроме точек пересечения графиков функций $\rho_{i,i+1}(t)$.

Пусть в момент времени $t = t_1$ производная $\delta(t_1)$ существует. Покажем, что векторы скоростей объектов, на которых реализуется

минимум в момент времени t_1 , направлены в противоположные стороны (см. рис. 1). Пусть $\delta(t_1) = \rho_{i,i+1}(t_1)$, причем i , возможно, не единственен, т. е. $\dot{x}_i \leq 0$. Тогда при $t = t_1$, точка i движется в левую сторону или стоит на месте, так как $x_i(t) - x_{i-1}(t) \geq x_{i+1}(t) - x_i(t)$, что в силу динамики (1) означает отрицательную скорость и движение влево. Аналогично при $t = t_1$ точка $i + 1$ движется вправо или стоит на месте в этот момент, т. е. $\dot{x}_{i+1} \geq 0$. Это значит, что производная $\rho'_{i,i+1}(t_1) \geq 0$. Поскольку выбор момента времени t_1 произволен, то лемма доказана для всех таких t_1 , для которых производная $\delta(t_1)$ существует.

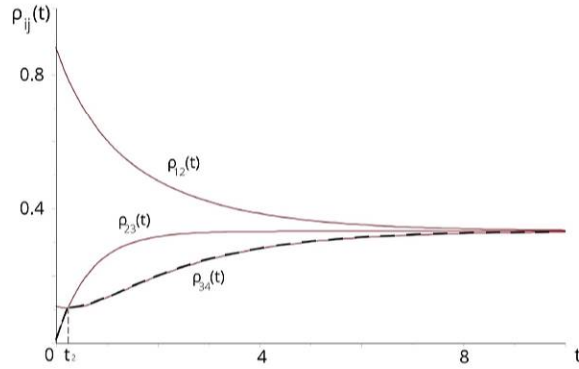


Рис. 2. Графики функций расстояний между соседними точками для случая $n = 4$. Пунктиром обозначен график функции $\delta(t)$

Пусть при $t = t_2$ производная $\delta'(t_2)$ не существует. Это означает, что в момент времени t_2 в одной точке пересекаются несколько графиков функций $\rho_{i,i+1}(t)$ при различных i (см. рис. 2), т. е. между k агентами реализовалось минимальное расстояние. Покажем, что в этом случае ни одна из функций $\rho_{i,i+1}(t)$ для всех таких i , что $\rho_{i,i+1}(t_2) = \delta(t_2)$ не может убывать на достаточно малом промежутке $[t_2, t_2 + \varepsilon]$, $\varepsilon > 0$. Если $k = 2$, то точки расходятся так же, как и в первом случае, так как векторы скорости не могут быть нулевыми (см. рис. 2). Если $k > 2$, то некоторые точки стоят на месте и производная $\rho'_{i,i+1}(t_1) = 0$. Покажем, что ни одна из этих функций не может убывать на достаточно малом промежутке времени

$[t_2, t_2 + \varepsilon], \varepsilon > 0$. Рассмотрим последовательность подряд идущих k точек, между которыми реализовано минимальное расстояние при $t = t_2$. Если точки x_i и x_{i+k-1} стоят на месте, то это крайние точки и, следовательно, все точки стоят на месте (на равных расстояниях). Иначе движение точек x_i и x_{i+k-1} противоположно в силу минимальности расстояния $\rho_{i,i+1}(t_2)$ и движения (1). Рассмотрим промежуток времени, на котором направления движения этих точек неизменны $\delta > 0$. На этом промежутке все точки с индексами меньше i имеют $\dot{x}_i \leq 0$, а с индексами больше $i + 1 - \dot{x}_i \geq 0$. При этом расстояния между каждой парой соседей не уменьшаются, так как расстояние между точками ближе к центру не может быть больше, чем расстояние между парой точек дальше от центра в силу движения (1).

Лемма доказана.

Сформулируем необходимые и достаточные условия на начальное расположение объектов для разрешимости основной задачи в одномерном случае.

Теорема 1. Пусть динамика середин отрезков $x_i, i = 1, \dots, n$, длины $2R$ задана системой (1). Для того чтобы задача равномерного при условии (2) расположения объектов-отрезков на отрезке $[0, L]$ была разрешима, необходимо и достаточно выполнение следующих условий:

1. Объекты должны быть занумерованы согласно условию (3).
2. Расстояние $\rho_{i,i+1}(0) \geq \varepsilon$.

Доказательство. Необходимость. Если условие (3) не выполнено, то объекты, движение которых задается системой (1), очевидно, столкнутся. Необходимость условия (2) очевидна.

Достаточность. Согласно лемме минимальное расстояние между соседними объектами не убывает. Значит, если в момент времени $t = 0$ положение объектов удовлетворяет ограничению (2), то и для всех $t > 0$ это верно.

Теорема 1 доказана.

3.2. Ограниченность условий разрешимости задачи.

Рассмотрим, насколько ограниченно первое условие теоремы 1. Обозначим через $P(A)$ вероятность события A .

Теорема 2. Пусть $x(0) = x_i$ – случайная величина, равномерно распределенная на отрезке $[0, 1]$. Вероятность

$$P(\min_{i,j} |x_i - x_j| > \varepsilon \cap \varepsilon \leq x_i \leq 1 - \varepsilon, i, j = 1, \dots, n, i \neq j) = \\ = (1 - (n + 1)\varepsilon)^n.$$

Доказательство. Пусть $P(n, \varepsilon) = (1 - (n + 1)\varepsilon)^n$. Вычислим $D(n, \varepsilon) = \frac{P(n, \varepsilon)}{n!}$, т. е. вероятность из условия теоремы для n упорядоченных точек. Данную вероятность можно вычислять геометрически. Событие – точка с n координатами, следовательно, необходимо найти n -мерный объем множества точек, соответствующих благоприятному событию [13]. При упорядочивании исходных точек получаем упорядочивание значений по осям, т. е. координата по оси x_1 должна принимать значения на отрезке $[0, 1 - n\varepsilon]$: $x_1 \in [0, 1 - n\varepsilon]$, $x_2 \in [x_1 + \varepsilon, 1 - (n - 1)\varepsilon]$ и т. д. Таким образом,

$$D(1, \varepsilon) = \int_{x_1=\varepsilon}^{1-\varepsilon} dx_1 = 1 - 2\varepsilon, \\ D(2, \varepsilon) = \int_{x_1=\varepsilon}^{1-2\varepsilon} \int_{x_2=x_1+\varepsilon}^{1-\varepsilon} dx_2 dx_1, \\ D(n, \varepsilon) = \int_{x_1=\varepsilon}^{1-n\varepsilon} \int_{x_2=x_1+\varepsilon}^{1-(n-1)\varepsilon} \dots \int_{x_n=x_{n-1}+\varepsilon}^{1-\varepsilon} dx_n \dots dx_1.$$

Это нетрудно показать с помощью математической индукции. Введем следующее обозначение:

$$Q(n, w) = \int_{z=w+\varepsilon}^{1-n\varepsilon} Q(n-1, z) dz, \quad (4)$$

где w – формальный параметр. Несложно убедиться в равенстве $D(n, \varepsilon) = Q(n, 0)$. Покажем, используя метод математической индукции, что

$$Q(n, w) = \frac{(1 - (n + 1)\varepsilon + w)^n}{n!}.$$

При $n = 0$ очевидно, что, пользуясь рекуррентным соотношением (4) и учитывая индукционное предположение, получаем

$$Q(n, w) = \int_{z=w+\varepsilon}^{1-n\varepsilon} \frac{(1 - n\varepsilon + z)^{n-1}}{(n-1)!} dz.$$

В результате интегрирования имеем

$$Q(n, w) = \frac{(1 - (n + 1)\varepsilon + w)^n}{n!}.$$

Таким образом, $D(n, \varepsilon) = Q(n, 0) = \frac{(1 - (n + 1)\varepsilon)^n}{n!}$.

Теорема 2 доказана.

Аналогичная задача решена в [13].

Из теоремы 2 вероятность подходящих начальных данных $P(n) = (1 - (n + 1)\varepsilon)^n$. При $\varepsilon = \frac{k}{n+1}$ имеем вероятность, при условии того, что точки могут быть на расстоянии не более чем $\frac{1}{k}$ от расстояния после построения, т. е. $\frac{1}{n+1}$. При увеличении числа объектов данная вероятность стремится к нулю.

При $\varepsilon = \frac{k}{(n+1)n}$ получаем вероятность, при условии того, что точки могут быть на расстоянии не более чем $\frac{1}{kn}$ от расстояния после построения, т. е. при увеличении числа объектов данная вероятность стремится к e^{-k} .

4. Выводы и перспективы. В статье рассмотрен алгоритм движения объектов для выстраивания на отрезке через равные расстояния. При этом использована динамика из [5] и добавлены ограничения на столкновения объектов. Получены необходимые и достаточные условия на начальные данные для одномерного случая, оценена вероятность таких начальных данных. Вопрос об условиях на начальные данные в двумерном случае исследуется, имеются гипотезы, аналогичные лемме и теореме 1, где помимо расстояний учитываются углы между «соседними» отрезками, как это сделано, например, в [14].

Литература

1. Леонтович А. М., Пятецкий-Шапиро И. И., Ставская О. Н. Некоторые математические задачи, связанные с формообразованием // Автоматика и телемеханика. 1970. Вып. 4. С. 94–107.
2. Квинто Я. И., Парсегов С. Э. Равноудаленное расположение агентов на отрезке: анализ алгоритма и его обобщения // Автоматика и телемеханика. 2012. Вып. 11. С. 30–41.
3. Леонтович А. М. Скорость сходимости в задаче «окруживания» // Математический сборник. 1972. Т. 88(130). № 1(5). С. 88–117.

4. Леонтович А. М., Пятецкий-Шапиро И. И., Ставская О. Н. Задача окруживания в математической модели формообразования // Автоматика и телемеханика. 1971. Вып. 2. С. 100–110.
5. Парсегов С. Э. Сверхфинитная стабилизация мультиагентной системы // Проблемы управления. 2012. Вып. 6. С. 7–12.
6. Parsegov S. E. Allocation of agents on a line: simple algorithm and generalizations // Proc. 14th WOAC. 2011. P. 119–125.
7. Olfati-Saber R., Fax J. A., Murray R. M. Consensus and cooperation in networked multi-agent systems // Proc. IEEE. 2007. Vol. 95. No 1. P. 215–233.
8. Wagner I. A., Bruckstein A. M. Row straightening via local interactions // Circuits, Systems and Signal Processing. 1997. Vol. 16. No 3. P. 287–305.
9. Петрикевич Я. И. Линейные алгоритмы управления геометрическим расположением объектов в многоагентной системе // Управление большими системами. Спец. выпуск 30.1 «Сетевые модели в управлении». М.: ИПУ РАН, 2010. С. 665–680.
10. Парсегов С. Э. Сцепление координат и иерархические алгоритмы в задаче равноудаленного расположения агентов на отрезке // Управление большими системами. 2012. Вып. 39. С. 264–287.
11. Проскурников А. В., Парсегов С. Э. Задача равномерного размещения на отрезке для агентов с моделью второго порядка // Автоматика и телемеханика. 2016. Вып. 7. С. 152–165.
12. Проскурников А. В. Консенсус в нелинейных стационарных сетях с идентичными агентами // Автоматика и телемеханика. 2015. Вып. 9. С. 44–63.
13. Васильев Н. Б., Леонтович А. М. Оценка случайности перекрытия фигур // Проблемы передачи информации. 1998. Т. 34. Вып. 1. С. 87–111.
14. Ставская О. Н. Исследование сходимости в задаче о «спрямлении» // Математический сборник. 1972. Т. 88(130). № 1(5). С. 118–136.

УДК 517.929.4

Белов А. И.

Критерий экспоненциальной устойчивости для линейной системы с запаздыванием

Рекомендовано к публикации доцентом Александровой И. В.

1. Введение. Одним из подходов к исследованию экспоненциальной устойчивости линейных систем с запаздыванием является подход Ляпунова – Красовского. Он заключается в построении функционала с заданной отрицательно-определенной производной вдоль решений системы и последующем исследовании его на положительную определенность [1]. Основной проблемой, возникающей при применении этого подхода является конструктивная проверка положительной определенности функционала. В работе [2] задача сводится к проверке положительной определенности некоторой матрицы, однако размерность получаемой матрицы оказывается большой. В данной статье получен новый критерий проверки устойчивости, в котором задача сводится к проверке неотрицательной определенности двухпараметрического семейства матриц размерности $2n \times 2n$, где n – порядок системы.

Введем обозначения. Пусть $H := PC([-h, 0], \mathbb{R}^n)$ – пространство кусочно-непрерывных функций, определенных на $[-h, 0]$ и действующих в \mathbb{R}^n . Нотация $\|x\|$ обозначает евклидову норму вектора $x \in \mathbb{R}^n$. Под записью $\|\varphi\|_h$ будем понимать $\sup \|\varphi(\theta)\|$, $\theta \in [-h, 0]$. Нотация 0_n обозначает нулевой вектор размерности n .

2. Основные понятия. Рассмотрим систему

$$\dot{x}(t) = Ax(t) + Bx(t-h), \quad t \geq 0, \quad (1)$$

где $h \geq 0$ – запаздывание, $x(t) \in \mathbb{R}^n$, A, B – вещественные матрицы. Пусть $\varphi \in H$. Обозначим через $x(t, \varphi)$ решение системы (1), удовлетворяющее начальному условию $x(\theta, \varphi) = \varphi(\theta)$, $\theta \in [-h, 0]$.

Определение 1 [1]. Система (1) экспоненциально устойчива, если существуют числа $\gamma \geq 1, \sigma > 0$ такие, что любое решение системы $x(t, \varphi)$ удовлетворяет неравенству $\|x(t, \varphi)\| \leq \gamma e^{-\sigma t} \|\varphi\|_h$, $t \geq 0$.

Белов Александр Иванович – студент, Санкт-Петербургский государственный университет; e-mail: famesyasd@gmail.com, тел.: +7(911)704-23-97

Определение 2 [1]. Число $s \in \mathbb{C}$ называется собственным числом системы (1), если $\det(sE - A - Be^{-sh}) = 0$.

Определение 3 [1]. Пусть $s \in \mathbb{C}$ – собственное число системы (1). Вектор $C \in \mathbb{C}^n$ – собственный вектор, соответствующий числу s , если $C \neq 0_n$ и $(sE - A - Be^{-sh})C = 0_n$.

Теорема 1 [1]. Система (1) является экспоненциально устойчивой тогда и только тогда, когда все ее собственные числа лежат в открытой левой полуплоскости: $\{z \in \mathbb{C} : \operatorname{Re}(z) < 0\}$.

Определение 4 [1]. Матрица $U(\tau)$ называется матрицей Ляпунова системы (1), если она непрерывна и удовлетворяет свойствам:

1. $U'(\tau) = U(\tau)A + U(\tau - h)B, \quad \tau > 0,$
2. $U(-\tau) = U^T(\tau),$
3. $U(0)A + A^T U(0) + U(-h)B + B^T U(h) = -W,$

где W – симметричная матрица.

Определение 5 [1]. Система (1) удовлетворяет условию Ляпунова, если не существует числа $s \in \mathbb{C}$ такого, что s и $-s$ – собственные числа системы.

Теорема 2 [1]. Если система (1) экспоненциально устойчива, то она удовлетворяет условию Ляпунова.

Теорема 3 [1]. Если система (1) удовлетворяет условию Ляпунова, то существует матрица Ляпунова.

Пусть у системы (1) существует матрица Ляпунова. Определим квадратичный функционал $v_0 : H \rightarrow \mathbb{R}$

$$v_0(\varphi) := \varphi^T(0)U(0)\varphi(0) + 2\varphi^T(0) \int_{-h}^0 U(-h - \theta)B\varphi(\theta)d\theta + \int_{-h}^0 \int_{-h}^0 \varphi^T(\theta_1)B^T U(\theta_1 - \theta_2)B\varphi(\theta_2)d\theta_1 d\theta_2. \quad (2)$$

Введем функцию $\bar{\varphi}(\theta) = e^{\alpha\theta} (\cos(\beta\theta)C_1 - \sin(\beta\theta)C_2), \theta \in [-h, 0]$. Из доказательства теоремы 1 [3] следует следующая

Лемма 1. Пусть система (1) удовлетворяет условию Ляпунова. Тогда для любого собственного числа $s = \alpha + i\beta$ системы (1) такого, что $\alpha > 0$ и любого соответствующего ему собственного вектора $C = C_1 + iC_2, C_1, C_2 \in \mathbb{R}^n, C_1 \neq 0_n$ следует $v_0(\bar{\varphi}) < 0$.

3. Критерий проверки устойчивости линейной системы с запаздыванием. Введем обозначение $M := \|A\| + \|B\|$.

Теорема 4. Система (1) экспоненциально устойчива тогда и только тогда, когда она удовлетворяет условию Ляпунова и

$$v_0(\bar{\varphi}) \geq 0, \quad \forall C_1, C_2 \in \mathbb{R}^n, \quad \forall \alpha \in (0, M], \quad \forall \beta \in [0, M]. \quad (3)$$

Доказательство. Необходимость. Пусть система (1) экспоненциально устойчива. Тогда по теореме 2 она удовлетворяет условию Ляпунова и по теореме 3 существует квадратичный функционал v_0 . Причем известно [1], что в случае экспоненциальной устойчивости системы

$$v_0(\varphi) = \int_0^\infty x^T(t, \varphi) W x(t, \varphi) dt \geq 0$$

для любых функций φ .

Достаточность. Предположим, что система неустойчива. Тогда по теореме 1 существует собственное число $s = \alpha + i\beta$, лежащее в открытой правой полуплоскости либо на мнимой оси. На мнимой оси оно лежать не может, потому что выполнено условие Ляпунова. Таким образом, имеем $\alpha > 0$, $\beta \geq 0$. Если $\beta < 0$, то в качестве s можно взять сопряженное собственное число \bar{s} . Известно [2], что для s существует собственный вектор $C = C_1 + iC_2$, $C_1, C_2 \in \mathbb{R}^n$, $C_1 \neq 0_n$. По определению 3 имеем $(sE - A - Be^{-sh})C = 0_n$ или $sC = (A + Be^{-sh})C$, откуда $|s| \leq \|A\| + |e^{-sh}| \|B\| \leq \|A\| + \|B\|$, так как $|e^{-sh}| = e^{-\alpha h} \leq 1$. Поскольку $\alpha \leq |s|$ и $\beta \leq |s|$, окончательно получаем $\alpha \in (0, M]$, $\beta \in [0, M]$. По лемме 1 $v_0(\bar{\varphi}) < 0$, с другой стороны по условию $v_0(\bar{\varphi}) \geq 0$. Противоречие.

Теорема 4 доказана.

4. Конструктивный критерий экспоненциальной устойчивости. Непосредственной подстановкой функции $\bar{\varphi}$ в выражение для функционала (2) получим следующую лемму.

Лемма 2. Значение функционала v_0 на функции $\bar{\varphi}$ равно

$$v_0(\bar{\varphi}) = (C_1^T \ C_2^T) \begin{bmatrix} S_{1,1}(\alpha, \beta) & S_{1,2}(\alpha, \beta) \\ S_{1,2}^T(\alpha, \beta) & S_{2,2}(\alpha, \beta) \end{bmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}, \quad \text{где}$$

$$S_{1,1}(\alpha, \beta) := U(0) + \int_{-h}^0 (U(-\theta - h)B + B^T U(\theta + h)) e^{\alpha\theta} \cos(\beta\theta) d\theta +$$

$$+ \int_{-h}^0 \int_{-h}^0 B^T U(\theta_1 - \theta_2) B e^{\alpha\theta_1} e^{\alpha\theta_2} \cos(\beta\theta_1) \cos(\beta\theta_2) d\theta_1 d\theta_2,$$

$$S_{1,2}(\alpha, \beta) := - \int_{-h}^0 U(-\theta - h) B e^{\alpha\theta} \sin(\beta\theta) d\theta -$$

$$- \int_{-h}^0 \int_{-h}^0 B^T U(\theta_1 - \theta_2) B e^{\alpha\theta_1} e^{\alpha\theta_2} \cos(\beta\theta_1) \sin(\beta\theta_2) d\theta_1 d\theta_2,$$

$$S_{2,2}(\alpha, \beta) := \int_{-h}^0 \int_{-h}^0 B^T U(\theta_1 - \theta_2) B e^{\alpha\theta_1} e^{\alpha\theta_2} \sin(\beta\theta_1) \sin(\beta\theta_2) d\theta_1 d\theta_2.$$

Теорема 5. Система (1) экспоненциально устойчива тогда и только тогда, когда она удовлетворяет условию Ляпунова и $\forall \alpha \in (0, M], \beta \in [0, M]$ симметричная матрица

$$S(\alpha, \beta) := \begin{bmatrix} S_{1,1}(\alpha, \beta) & S_{1,2}(\alpha, \beta) \\ S_{1,2}^T(\alpha, \beta) & S_{2,2}(\alpha, \beta) \end{bmatrix}$$

неотрицательно определена.

Доказательство. Условие (3) в формулировке теоремы 4 с учетом леммы 2 эквивалентно условию

$$\forall \alpha \in (0, M], \forall \beta \in [0, M], (C_1^T \ C_2^T) \begin{bmatrix} S_{1,1}(\alpha, \beta) & S_{1,2}(\alpha, \beta) \\ S_{1,2}^T(\alpha, \beta) & S_{2,2}(\alpha, \beta) \end{bmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} \geq 0,$$

другими словами, неотрицательной определенности матрицы $S(\alpha, \beta)$ при любых $\alpha \in (0, M], \beta \in [0, M]$.

Теорема 5 доказана.

Таким образом, в теореме 5 задача анализа экспоненциальной устойчивости системы (1) сводится к проверке неотрицательной определенности двухпараметрического семейства матриц $S(\alpha, \beta)$, в котором параметры α, β изменяются на отрезке конечной длины.

5. Заключение. В данной работе получен критерий устойчивости для линейной системы с запаздыванием, в котором задача анализа устойчивости сводится к проверке неотрицательной определенности семейства матриц, определяемых матрицей Ляпунова.

Литература

1. Kharitonov V.L. Time-delay systems. Lyapunov functionals and matrices. Basel: Birkhäuser, 2013. 311 p.
2. Gomez M. A., Egorov A. V., Mondie S. Lyapunov matrix based necessary and sufficient stability condition by finite number of mathematical operations for retarded type systems // Automatica. 2019. Vol. 108. Art. No 108475.
3. Medvedeva I. V., Zhabko A. P. Synthesis of Razumikhin and Lyapunov-Krasovskii approaches to stability analysis of time-delay systems // Automatica. 2015. Vol. 51. P. 372–377.

УДК 517.929.4

Евтина Д. С.

О робастной устойчивости линейных систем с запаздыванием

Рекомендовано к публикации доцентом Александровой И. В.

1. Введение. Целью данной работы является анализ робастной устойчивости линейной системы дифференциальных уравнений с одним запаздыванием и неопределённостями одновременно в запаздывании и коэффициентах на основе метода функционалов Ляпунова–Красовского. Используется подход, разработанный в статье [1] для случая возмущений в запаздывании. Данный подход основан на построении специальной интегральной оценки производной функционала вдоль решений возмущённой системы.

Введём обозначения: $PC([-h, 0], \mathbb{R}^n)$ – пространство кусочно-непрерывных функций $\phi : [-h, 0] \rightarrow \mathbb{R}^n$ с равномерной нормой; $\|x\|$ – евклидова норма вектора $x \in \mathbb{R}^n$; $\|A\|$ – евклидова норма матрицы A , $\|A\| = \max_{\|x\|=1} \|Ax\|$.

2. Постановка задачи. Предположим, что система

$$\dot{x}(t) = A_0 x(t) + A_1 x(t - h), \quad (1)$$

где $x \in \mathbb{R}^n$, A_0, A_1 – постоянные матрицы, $h \geq 0$ – запаздывание, экспоненциально устойчива. Рассмотрим возмущённую систему

$$\dot{y}(t) = (A_0 + \Delta_0)y(t) + (A_1 + \Delta_1)y(t - h - \eta_1), \quad (2)$$

где $\eta_1 \geq -h$ – возмущение в запаздывании, а Δ_0, Δ_1 – матричные возмущения. Задача заключается в нахождении ограничений на возмущения $\Delta_0, \Delta_1, \eta_1$, при которых система (2) остаётся экспоненциально устойчивой.

3. Предварительные сведения. Зададим положительно-определённую матрицу W .

Евтина Диана Сергеевна – студент, Санкт-Петербургский государственный университет; e-mail: diana.evtina@mail.ru, тел.: +7(981)883-27-89

Определение [2]. Непрерывная матрица $U(\tau)$ называется матрицей Ляпунова системы (1), ассоциированной с симметрической матрицей W , если она удовлетворяет свойствам:

- 1) $\frac{d}{d\tau}U(\tau) = U(\tau)A_0 + U(\tau - h)A_1$, $\frac{d}{d\tau}U(0) = \lim_{\tau \rightarrow +0} \frac{d}{d\tau}U(\tau)$, $\tau \geq 0$;
- 2) $U(-\tau) = U^T(\tau)$;
- 3) $U(0)A_0 + A_0^T U(0) + U(-h)A_1 + A_1^T U^T(-h) = -W$.

Введём функционал, заданный на пространстве $PC([-h, 0], \mathbb{R}^n)$ [2]:

$$v_0(\phi) = \phi^T(0)U(0)\phi(0) + 2\phi^T(0) \int_{-h}^0 U(-\theta - h)A_1\phi(\theta)d\theta + \\ + \int_{-h}^0 \phi^T(\theta_1)A_1^T \int_{-h}^0 U(\theta_1 - \theta_2)A_1\phi(\theta_2)d\theta_2 d\theta_1.$$

Тогда

$$\left. \frac{d}{dt}v_0(x_t) \right|_{(1)} = -x^T(t)Wx(t), \quad t \geq 0.$$

Запишем систему (2) в виде

$$\dot{y}(t) = A_0y(t) + A_1y(t - h) + f(y_t), \\ f(y_t) = \Delta_0y(t) + A_1(y(t - h - \eta_1) - y(t - h)) + \Delta_1y(t - h - \eta_1).$$

Имеем [3]

$$\left. \frac{d}{dt}v_0(y_t) \right|_{(2)} = -y^T(t)Wy(t) + l(y_t), \quad (3)$$

$$l(y_t) = 2f^T(y_t) \left(U(0)y(t) + \int_{t-h}^t U(t - \xi - h)A_1y(\xi)d\xi \right).$$

4. Оценка функционала $l(y_t)$. Введём обозначения:

$$v = \max_{\theta \in [0, h]} \|U(\theta)\|, \quad a_1 = \|A_1\|, \quad \alpha = 1 + a_1h, \\ \delta_j = \|\Delta_j\|, \quad \rho_j = \|A_j + \Delta_j\|, \quad j = \overline{0, 1}, \quad H = \max\{h, h + \eta_1\}, \\ \beta = \delta_0 + \delta_1 + a_1|\eta_1|(\rho_0 + \rho_1), \quad \gamma = \delta_0 + \delta_1 + \rho_1|\eta_1|(\rho_0 + \rho_1). \quad (4)$$

Оценим функционал $l(y_t)$ двумя способами, полагая $t \geq H$.

Первый способ. Применим к функционалу $f(y_t)$ формулу Ньютона – Лейбница:

$$f(y_t) = \Delta_0 y(t) + \Delta_1 y(t - h - \eta_1) + A_1 \int_{-h}^{-h-\eta_1} (A_0 + \Delta_0) y(t + s) ds + \\ + A_1 \int_{-2h-\eta_1}^{-2h-2\eta_1} (A_1 + \Delta_1) y(t + s) ds.$$

Подставим это выражение в функционал $l(y_t)$ и учтём неравенство $2ab \leq a^2 + b^2$.

Лемма 1. Функционал $l(y_t)$ удовлетворяет неравенству:

$$l(y_t) \leq v(\beta + \alpha\delta_0) \|y(t)\|^2 + v\alpha\delta_1 \|y(t - h - \eta_1)\|^2 + v\beta a_1 \int_{t-h}^t \|y(\xi)\|^2 d\xi + \\ + v\alpha\rho_0 a_1 \left| \int_{-h}^{-h-\eta_1} \|y(t + s)\|^2 ds \right| + v\alpha\rho_1 a_1 \left| \int_{-2h-\eta_1}^{-2h-2\eta_1} \|y(t + s)\|^2 ds \right|. \quad (5)$$

Второй способ. Снова применим формулу Ньютона – Лейбница и представим функционал $f(y_t)$ в виде

$$f(y_t) = \Delta_0 y(t) + \Delta_1 y(t - h) + (A_1 + \Delta_1) \int_{-h}^{-h-\eta_1} (A_0 + \Delta_0) y(t + s) ds + \\ + (A_1 + \Delta_1) \int_{-2h-\eta_1}^{-2h-2\eta_1} (A_1 + \Delta_1) y(t + s) ds.$$

Лемма 2. Функционал $l(y_t)$ удовлетворяет неравенству:

$$\begin{aligned}
 l(y_t) \leq & v(\gamma + \alpha\delta_0)\|y(t)\|^2 + v\alpha\delta_1\|y(t-h)\|^2 + v\gamma a_1 \int_{t-h}^t \|y(\xi)\|^2 d\xi + \\
 & + v\alpha\rho_0\rho_1 \left| \int_{-h}^{-h-\eta_1} \|y(t+s)\|^2 ds \right| + v\alpha\rho_1^2 \left| \int_{-2h-\eta_1}^{-2h-2\eta_1} \|y(t+s)\|^2 ds \right|.
 \end{aligned} \tag{6}$$

5. Основной результат. Ключевым этапом работы является интегральная оценка функционала $l(y_t)$, полученная путём непосредственного интегрирования слагаемых, входящих в состав правых частей неравенств (5), (6).

Лемма 3. При $t \geq H$ справедлива оценка:

$$\int_H^t l(y_s) ds \leq L \int_H^t \|y(s)\|^2 ds + L_1 \int_{-H}^H \|y(s)\|^2 ds,$$

где $L = 2v\alpha M$, $L_1 = vM(2\alpha - 1)$, $M = \min\{\beta, \gamma\}$.

Основным результатом работы является следующая

Теорема. Пусть система (1) экспоненциально устойчива. Если

$$M < \frac{\lambda_{\min}(W)}{2v\alpha},$$

где $\lambda_{\min}(W)$ – минимальное собственное число матрицы W , а $M = \min\{\beta, \gamma\}$ определяется равенствами (4). Тогда система (2) остаётся экспоненциально устойчивой.

Доказательство. Проинтегрируем равенство (3) на промежутке $[H, t]$ и применим формулу Ньютона – Лейбница:

$$v_0(y_t) - v_0(y_H) = - \int_H^t y^T(s) W y(s) ds + \int_H^t l(y_s) ds.$$

Оценим правую часть получившегося равенства, воспользовавшись леммой 3. Получим

$$v_0(y_t) - v_0(y_H) \leq -(\lambda_{\min}(W) - L) \int_H^t \|y(s)\|^2 ds + \Psi,$$
$$\Psi = L_1 \int_{-H}^H \|y(s)\|^2 ds.$$

Условие теоремы эквивалентно тому, что $\lambda_{\min}(W) - L > 0$. Оставшаяся часть доказательства повторяет доказательство леммы 3 из работы [4].

Теорема доказана.

6. Заключение. В работе получены новые условия робастной устойчивости для линейных стационарных систем, содержащих возмущения одновременно в запаздывании и в матрицах. Полученные условия можно применить итерационно со сходимостью к точным границам области устойчивости в пространстве параметров [1]. Результат может быть обобщён на класс систем с распределённым запаздыванием и несколькими сосредоточенными запаздываниями, а также на случай нестационарных возмущений.

Литература

1. Alexandrova I. V., Zhabko A. P. A new LKF approach to stability analysis of linear systems with uncertain delays // *Automatica*. 2018. No 91. P. 173–178.
2. Kharitonov V. L. *Time-delay systems: Lyapunov functionals and matrices*. Basel: Birkhäuser, 2013. 311 p.
3. Egorov A. V., Mondié S. The delay Lyapunov matrix in robust stability analysis of time-delay systems // *IFAC-PapersOnLine*. 2015. Vol. 48. No 12. P. 245–250.
4. Alexandrova I. V. New robustness bounds for neutral type delay systems via functionals with prescribed derivative // *Applied Mathematics Letters*. 2018. No 76. P. 34–39.

УДК 517.977.1

Жигалов В. С.

Условия точечной управляемости дифференциально-разностной системы с линейно-возрастающим запаздыванием

Рекомендовано к публикации профессором Жабко А. П.

1. Введение. В работе рассматривается управляемая система дифференциально-разностных уравнений с постоянными коэффициентами вида

$$\dot{x}(t) = A_0x(t) + A_1x(\alpha t) + Bu(t), \quad (1)$$

где A_0, A_1 – $(n \times n)$ -матрицы, B – $(n \times r)$ -матрица, $\alpha \in (0, 1)$. Управление и стабилизация систем вида (1) с постоянным запаздыванием рассматривались в работах [1, 2], в которых введено понятие полной управляемости и точечной управляемости для систем с постоянным запаздыванием. Однако (1) является системой с линейно возрастающим (неограниченным) запаздыванием, и в данной работе предлагаются достаточные условия точечной управляемости.

Определение 1. Система (1) называется точно управляемой на промежутке $[t_0, T]$, где $t_0 > 0$, если для любых начальных условий

$$x(t) = \phi(t), \quad t \in [\alpha t_0, t_0]$$

и любого значения $x(T) = x_1$ существует вектор-функция $u(t)$, переводящая ее из функции $x_{t_0} = \phi$ в состояние x_1 .

Системы и уравнения с линейным запаздыванием встречаются в математических моделях радиоактивного распада [3], работы информационного сервера [4], смесительного бака [5] и в ряде других случаев.

В работе [6] построен весь класс программных управлений в случае системы дифференциальных уравнений без запаздывания. В данной работе решение задачи основано на развитии этого подхода для систем с линейно возрастающим запаздыванием.

Жигалов Валентин Сергеевич – студент, Санкт-Петербургский государственный университет; e-mail: zhigalovvs.98b@gmail.com, тел.: +7 (914) 324-31-26

2. Вспомогательные сведения. Рассмотрим систему

$$\dot{x}(t) = A_0x(t) + A_1x(\alpha t) + f(t) \quad (2)$$

с кусочно-непрерывной начальной функцией $\phi(t)$, заданной на промежутке $[\alpha t_0, t_0]$.

Определение 2. Фундаментальной матрицей системы (2) называется матричная функция $\Phi(t, t_0)$, удовлетворяющая соотношениям:

$$\begin{cases} \Phi(t, t_0) \equiv 0, & t < t_0, \\ \Phi(t, t_0) = E, & t = t_0, \\ \frac{\partial \Phi(t, t_0)}{\partial t} = A_0\Phi(t, t_0) + A_1\Phi(\alpha t, t_0), & t > t_0. \end{cases}$$

Известно, что общее решение системы (2) в форме Коши имеет вид

$$x(t) = \Phi(t, t_0)x_0 + \int_{t_0}^{\alpha^{-1}t_0} \Phi(t, \tau)A_1\phi(\alpha\tau)d\tau + \int_{t_0}^t \Phi(t, \tau)f(\tau)d\tau. \quad (3)$$

Замечание 1. В нашем случае для анализа системы (1) формула (3) принимает вид

$$x(t) = \Phi(t, t_0)x_0 + \int_{t_0}^{\alpha^{-1}t_0} \Phi(t, \tau)A_1\phi(\alpha\tau)d\tau + \int_{t_0}^t \Phi(t, \tau)Bu(\tau)d\tau. \quad (4)$$

3. Достаточные условия точечной управляемости. Поскольку система (1) является нестационарной и явных методов построения $\Phi(t, t_0)$ в общем случае не существует, рассмотрим условия точечной управляемости системы на промежутках $[t_0, \alpha^{-1}t_0]$ и $[t_0, \alpha^{-2}t_0]$.

Пусть $T \in [t_0, \alpha^{-1}t_0]$. Согласно определению

$$\begin{cases} \frac{\partial \Phi(t, t_0)}{\partial t} = A_0\Phi(t, t_0), \\ \Phi(t_0, t_0) = E. \end{cases}$$

Поэтому при $t \in [t_0, \alpha^{-1}t_0]$ выполняется

$$\Phi(t, t_0) = \exp(A_0(t - t_0)).$$

Теорема 1. Система (1) точно управляема на промежутке $[t_0, \alpha^{-1}t_0]$ тогда и только тогда, когда $\text{rang}(B, A_0B, \dots, A_0^{n-1}B) = n$.

Теперь рассмотрим случай $T \in [\alpha^{-1}t_0, \alpha^{-2}t_0]$. Для построения матрицы $\Phi(t, t_0)$ на промежутке $[\alpha^{-1}t_0, \alpha^{-2}t_0]$ найдем решение следующей системы:

$$\begin{cases} \frac{\partial \Phi(t, t_0)}{\partial t} = A_0 \Phi(t, t_0) + A_1 \exp(A_0(\alpha t - t_0)), \\ \Phi(\alpha^{-1}t_0, t_0) = \exp(A_0(\alpha^{-1}t_0 - t_0)). \end{cases}$$

При $t \in [\alpha^{-1}t_0, \alpha^{-2}t_0]$ справедливо равенство

$$\begin{aligned} \Phi(t, t_0) &= \exp(A_0(t - t_0)) + \\ &+ \int_{\alpha^{-1}t_0}^t \exp(A_0(t - \tau)) A_1 \exp(A_0(\alpha \tau - t_0)) d\tau. \end{aligned}$$

Замечание 2. Используя метод шагов, можно построить матрицу $\Phi(t, t_0)$ при $t \in [t_0, T]$.

С условием $x(T) = x_1$ из (4) получаем равенство

$$\Phi(T, t_0)x_0 + \int_{t_0}^{\alpha^{-1}t_0} \Phi(T, \tau) A_1 \phi(\alpha \tau) d\tau + \int_{t_0}^T \Phi(T, \tau) B u(\tau) d\tau = x_1. \quad (5)$$

Обозначим

$$b = \Phi(T, t_0)x_0 + \int_{t_0}^{\alpha^{-1}t_0} \Phi(T, \tau) A_1 \phi(\alpha \tau) d\tau - x_1, \quad Q(\tau) = \Phi(T, \tau)B. \quad (6)$$

Будем искать управление в виде

$$u(t) = Q^T c + v(t), \quad (7)$$

где c – постоянный вектор, а $v(t)$ – функция, суммируемая с квадратом на $[t_0, T]$ и такая, что

$$\int_{t_0}^T Q(t)v(t)dt = 0.$$

После подстановки (6), (7) в уравнение (5) получаем

$$A(T, t_0)c = b, \quad (8)$$

где $A(T, t_0) = - \int_{t_0}^T Q Q^T dt$.

Уравнение (8) имеет решение для любого b , если $\det A(T, t_0) \neq 0$. Таким образом, справедлива

Теорема 2. *Для того чтобы система (1) была точечно управляемой на промежутке $[t_0, T]$, необходимо и достаточно, чтобы матрица $A(T, t_0) = - \int_{t_0}^T Q Q^T dt$, где $Q = \Phi B$, была неособой. При этом все множество программных управлений задается формулой*

$$u = Q^T c + v,$$

где $v(t)$ – суммируемая с квадратом на $[t_0, T]$ функция и

$$\int_{t_0}^T Q v dt = 0, \quad c = A^{-1} \left(\Phi(T, t_0)x_0 + \int_{t_0}^{\alpha^{-1}t_0} \Phi(T, \tau) A_1 \phi(\alpha \tau) d\tau - x_1 \right).$$

Замечание 3. В качестве достаточного условия точечной управляемости (1) можно рассмотреть следующее:

$$\exists t \in [t_0, T] : \text{rang} \left(\Phi(t, t_0)B, \frac{\partial \Phi(t, t_0)}{\partial t} B, \dots, \frac{\partial^{n-1} \Phi(t, t_0)}{\partial t^{n-1}} B \right) = n. \quad (9)$$

При исследовании точечной управляемости более полезным может оказаться интегральный признак линейной независимости векторных функций.

Теорема 3 [6]. *Для того чтобы строки $q_1(t), \dots, q_m(t)$ матрицы $Q(t)$ были линейно независимыми на отрезке $[t_0, T]$, необходимо и достаточно, чтобы интегральная матрица*

$$A = \int_{t_0}^T Q(t) Q^T(t) dt$$

была положительно определенной.

Следствие. Для того чтобы система (1) была точно управляема на $[t_0, T]$, необходимо и достаточно, чтобы строки матрицы $\Phi(T, t)B$ были линейно независимы на отрезке $[t_0, T]$.

Пример. Пусть в системе (1)

$$A_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad A_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

и известно, что $x_0 = x(t_0), T = \alpha^{-2}t_0$. Построим для такой системы управление $u(t)$ на отрезке $[t_0, T]$, чтобы выполнялось условие $x(T) = 0$. После вычисления фундаментальной матрицы получим представление

$$\Phi(T, t)B = \begin{cases} \begin{pmatrix} e^{T-t} \\ \frac{1}{\alpha-1}(e^{\alpha T + (\alpha^{-1}-1)t} - e^T) \end{pmatrix}, & t \in [t_0, \alpha^{-1}t_0], \\ \begin{pmatrix} e^{T-t} \\ 0 \end{pmatrix}, & t \in [\alpha^{-1}t_0, \alpha^{-2}t_0]. \end{cases}$$

Очевидно, что строки матрицы $\Phi(T, t)B$ линейно независимы на промежутке $[t_0, T]$. Следовательно, согласно следствию из теоремы 3, для любого значения ϕ существует точное управление, переводящее систему в состояние $x(T) = 0$.

Теперь проверим выполнение (9). Матрица из этого условия выглядит следующим образом:

$$\begin{cases} \begin{pmatrix} e^{t-t_0} & e^{t-t_0} \\ 0 & 0 \end{pmatrix}, & t \in [t_0, \alpha^{-1}t_0], \\ \begin{pmatrix} e^{t-t_0} & e^{t-t_0} \\ \frac{1}{\alpha-1}(\beta - e^t) & \frac{1}{\alpha-1}(\alpha\beta - e^t) \end{pmatrix}, & t \in [\alpha^{-1}t_0, \alpha^{-2}t_0], \end{cases}$$

где $\beta = e^{\alpha t + (\alpha^{-1}-1)t_0}$.

Очевидно, что условие (9) в случае рассматриваемой системы выполняется.

4. Заключение. В работе предложено несколько критериев точечной управляемости систем вида (1). Однако остался вопрос о существовании достаточных условий точечной управляемости, выраженных в терминах матриц A_0, A_1, B .

Литература

1. Марченко В. М. К управляемости линейных систем с последствием // ДАН СССР. 1977. Т. 236. № 5. С. 1083–1086.
2. Метельский А. В. Полное успокоение линейной автономной дифференциально-разностной системы с регулятором того же типа // Дифференциальные уравнения. 2012. Т. 48. № 9. С. 1240–1255.
3. Беллман Р., Кук К. Л. Дифференциально-разностные уравнения. М: Мир, 1967. 548 с.
4. Жабко А. П., Чижова О. Н. Анализ устойчивости однородного дифференциально-разностного уравнения с линейным запаздыванием // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2015. Т. 11. № 3. С. 105–115.
5. Жабко А. П., Чижова О. Н. Гибридный метод анализа устойчивости линейных дифференциально-разностных систем с линейно возрастающим запаздыванием // Вестник Тамбовского университета. Серия: Естественные и технические науки. 2015. Т. 20. № 4. С. 843–850.
6. Зубов В. И. Лекции по теории управления. СПб.: Лань, 2009. 496 с.

УДК 517.935.4

Жолобов Е. В.

Прогнозирование места аварийной посадки квадрокоптера

Рекомендовано к публикации профессором Смирновым Н. В.

1. Введение. Под управлением беспилотным летательным аппаратом (БПЛА) подразумевается его контроль в соответствии с полетным заданием. Однако некоторые подходы к данной задаче могут быть дополнены алгоритмами, направленными на анализ поведения квадрокоптера. Например, нахождение БПЛА над поверхностью с водой вносит некоторые ограничения на управление аппаратом в виде отсутствия пространства для посадки. В связи с этим считается актуальной формализация метода оценки расстояния приземления квадрокоптера.

2. Постановка задачи. Математическая модель квадрокоптера. Перемещение квадрокоптера может быть описано следующей системой [1]:

$$\begin{aligned} \frac{dx}{dt} &= V_x, & \frac{dy}{dt} &= V_y, & \frac{dz}{dt} &= V_z, \\ m \frac{dV_x}{dt} &= (\sin \psi \sin \varphi + \cos \psi \sin \theta \cos \varphi) U_1, \\ m \frac{dV_y}{dt} &= (-\cos \psi \sin \varphi + \sin \psi \sin \theta \cos \varphi) U_1, \\ m \frac{dV_z}{dt} &= U_1 \cos \theta \cos \varphi - mg, \\ \frac{d\varphi}{dt} &= \omega_\varphi, & \frac{d\theta}{dt} &= \omega_\theta, & \frac{d\psi}{dt} &= \omega_\psi, \\ I_{xx} \frac{d\omega_\varphi}{dt} &= (I_{yy} - I_{zz}) \omega_\theta \omega_\psi - J_{TP} \omega_\theta \Omega + U_2, \\ I_{yy} \frac{d\omega_\theta}{dt} &= (I_{zz} - I_{xx}) \omega_\psi \omega_\varphi + J_{TP} \omega_\varphi \Omega + U_3, \\ I_{zz} \frac{d\omega_\psi}{dt} &= (I_{xx} - I_{yy}) \omega_\psi \omega_\theta + U_4, \end{aligned} \tag{1}$$

Жолобов Ефим Витальевич – студент, Санкт-Петербургский государственный университет; e-mail: st062550@student.spbu.ru, тел.: +7(958)587-73-71

где x, y, z – координаты центра масс, V_x, V_y, V_z – проекции вектора линейной скорости, φ, θ, ψ – угол крена, тангажа и рыскания соответственно, ω_φ – угловая скорость крена, ω_θ – угловая скорость тангажа, ω_ψ – угловая скорость рыскания, m – масса квадрокоптера, I_{xx}, I_{yy}, I_{zz} – моменты инерции вокруг осей x, y и z соответственно, U_1, U_2, U_3, U_4 – каналы управления, Ω – общая скорость четырёх винтов, J_{TP} – общий вращательный момент инерции вокруг оси винта.

Уравнения связи каналов управления U_1, U_2, U_3, U_4 со скоростями вращения винтов $\Omega_1, \Omega_2, \Omega_3, \Omega_4$ имеют вид

$$\begin{aligned} U_1 &= b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2), & U_2 &= lb(-\Omega_2^2 + \Omega_4^2), \\ U_3 &= lb(-\Omega_1^2 + \Omega_3^2), & U_4 &= d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2), \\ \Omega &= -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4, \end{aligned} \quad (2)$$

где l – расстояние между центрами квадрокоптера и пропеллера, b и d – аэродинамические составляющие тяги и коэффициента сопротивления соответственно.

Для стандартных типов движений квадрокоптера (взлет, посадка, движение в горизонтальной плоскости) могут быть использованы управляющие сигналы вида

$$\Omega_i = \begin{cases} C_i, & \text{если } t \in T_1, \\ C_i + \frac{a_0}{2} + \sum_{k=1}^n (a_k \cos(kt) + b_k \sin(kt)), & \text{если } t \in T_2. \end{cases}$$

Здесь C_i – угловая скорость, необходимая для компенсации силы тяжести, a_k и b_k – параметры тригонометрического многочлена, задающие отклонения, необходимые для гладкого выхода на стационарный режим, T_1 – множество отрезков времени, при которых аппарат поддерживает висение, T_2 – множество отрезков времени, при которых аппарат маневрирует. Такая форма управляющих сигналов близка к оптимальной по расходу энергии [2] и при этом является гладкой функцией, что применимо в практическом смысле.

С точки зрения модели нарушение работоспособности винта может быть представлено как мгновенное изменение величин

$$\Omega_i^{em} = \Omega_i - \varepsilon_i, \quad \varepsilon_i \in [0, \Omega_i], \quad i = \overline{1, 4}.$$

Случай, когда $\varepsilon_i = \Omega_i$ и, следовательно, $\Omega_i^{em} = 0$, будет описывать ситуацию с полной неработоспособностью соответствующего пропеллера

лера. Подстановка аварийных значений Ω_i^{em} в выражения для элементов управления (2) и далее в систему (1) определит математическую модель аварийной ситуации.

Основной целью данной работы является определение потенциальной зоны посадки аппарата в аварийной ситуации. В этой связи проводится ряд численных экспериментов, основанных на изучении динамики перемещения аппарата в зависимости от объёма повреждения (величины ε_i) одного из винтов. Таким образом, в силу теоремы о непрерывности решений систем дифференциальных уравнений от правых частей совокупность полученных точек приземления аппарата будет определять ожидаемую зону падения квадрокоптера в аварийной ситуации. Также рассматривается случай использования алгоритма для отказоустойчивого приземления [3] с переменным значением начала действия после идентификации аварии.

3. Численные эксперименты. Рассмотрим квадрокоптер с параметрами согласно (1):

$$\begin{aligned} m &= 1 \text{ кг}, \quad l = 0,175 \text{ м}, \quad b = 26,5 \cdot 10^{-6} \text{ Н} \cdot \text{м} \cdot \text{с}^2, \\ d &= 0,6 \cdot 10^{-6} \text{ Н} \cdot \text{м} \cdot \text{с}^2, \quad I_{xx} = I_{yy} = I_{zz} = 0,1 \text{ Н} \cdot \text{м} \cdot \text{с}^2, \\ J_{TP} &= 0,005 \text{ Н} \cdot \text{м} \cdot \text{с}^2. \end{aligned}$$

Проведем ряд моделирований аварийных ситуаций, выбрав в качестве начального положения аппарата $(x_0, y_0, z_0) = (0, 0, 50)$. Будем рассматривать следующие значения угловой скорости повреждённого винта Ω_2 :

$$\Omega_2^{em_i} = 0 + 3i, \quad i = \overline{0, 100}.$$

Значение $\Omega_2^{em_{100}} = 300$ об/с соответствует слабому повреждению пропеллера, так как необходимое значение угловой скорости каждого винта для поддержания состояния покоя $\Omega_{hovering} = 304,1691$ об/с.

Таким образом, получим график, представленный на рис. 1. Каждая точка является координатами падения аппарата в i -й аварийной ситуации. Можно заметить, что малые по величине угловые скорости не сильно влияют на расстояние между точками приземления аппарата и начальным положением, в то время как между точками со значениями $\Omega_2^{em_i}$, соответствующими небольшому повреждению аппарата, изменение расстояния заметно больше. Это наблюдение является следствием того, что при слабом нарушении работоспособности винта квадрокоптер длительное время не достигает критического наклона, приводящего к неконтрольному падению. Иными

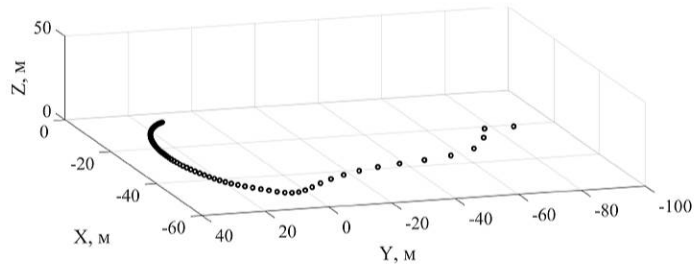


Рис. 1. Множество точек приземления аппарата при разных значениях Ω_2^{em}

словами, чем меньше повреждён винт, тем дальше пролетит аппарат, прежде чем упадёт на землю.

Далее рассмотрим воздействие алгоритма отказоустойчивого приземления квадрокоптера [3] в тех же условиях моделирования, но с учётом введения дополнительного, измеряемого в секундах, параметра $t_{reaction} = 0,5 + 0,1i$, где $i = \overline{0, 5}$.

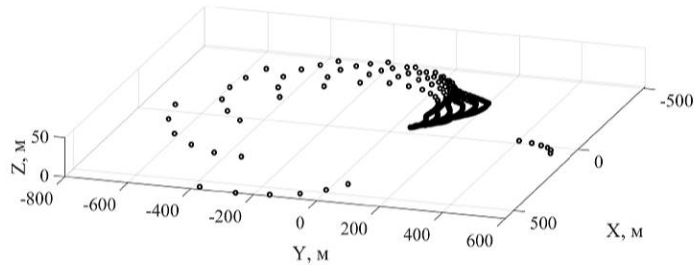


Рис. 2. Множество точек приземления аппарата при разных $\Omega_2^{em}, t_{reaction}$

Важно заметить, что в данной работе метод аварийной посадки не полностью обнуляет скорости повреждённого и симметричного ему винтов, а сохраняет угловые скорости обоих пропеллеров одинаковыми. В данном случае на время действия отказоустойчивого алгоритма $\Omega_4 = \Omega_2^{em_i}$, что устраняет наклон аппарата и позволяет сохранить некоторую тягу, облегчая при посадке работу полностью исправной паре винтов. Таким образом, можно исследовать зависимость перемещения квадрокоптера от времени начала действия спа-

сательного алгоритма после возникновения аварии. На рис. 2 видно, что при работе отказоустойчивого метода аппарат летит тем дальше, чем больше значение Ω_2^{emi} . Время начала работы алгоритма посадки после возникновения аварии $t_{reaction}$ влияет на направление посадки квадрокоптера – множество точек приземления образует «витки», в которых каждая точка соответствует определенному времени переключения на спасательный алгоритм.

4. Заключение. В данной работе продемонстрирован подход, позволяющий определить ожидаемую зону посадки квадрокоптера в определенной аварийной ситуации. Основываясь на результатах моделирования, при возникновении аварийной ситуации можно определить область падения аппарата или, в случае подключения отказоустойчивого алгоритма, определить примерную дальность и направление посадки, что позволяет избежать перемещение в нерекомендуемую для посадки зону, варьируя время запуска программы после потери тяги на одном из винтов. Таким образом, полученные данные потенциально могут быть использованы для усовершенствования работы алгоритма посадки [3, 4].

Литература

1. Bresciani T. Modeling, identification and control of a quadrotor helicopter. Master thesis. Sweden, Lund: Lund University, 2008. 184 p.
2. Попков А. С., Баранов О. В. Об оптимальном управлении вращательным движением вала электродвигателя // Процессы управления и устойчивость. 2014. Т. 1. № 1. С. 31–36.
3. Baranov O. V., Smirnov N. V., Smirnova T. E., Zholobov Y. V. Design of a quadcopter with PID-controlled fail-safe algorithm // Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2020. Vol. 11. № 2. P. 23–33.
4. Baranov O. V., Smirnov N. V., Smirnova T. E., Zholobov Y. V. Design of fail-safe quadcopter configuration // Studies in Computational Intelligence. 2020. Vol. 868. P. 13–22.

УДК 517.977.5

Иванов А. В., Моисеев И. А.

Частный случай синтеза оптимального управления в задаче перелета

1. Введение. Рассматривается задача быстрогодействия для объекта, движение которого описывается нелинейной системой обыкновенных дифференциальных уравнений. Достаточно часто ее формулируют как дифференциальную игру преследования «шофер-убийца». Впервые ее сформулировал и рассмотрел Р. Айзекс [1, 2]. В последующие годы рассматривались более сложные системы, описывающие исходное движение [3–7], различные численные методы решения задачи [8, 9]. Достаточно широко для подобных задач используются методы теории оптимального управления [10–15].

2. Постановка задачи. Итак, движение объекта удовлетворяет следующей системе обыкновенных дифференциальных уравнений:

$$\begin{aligned}\dot{x} &= V \sin \varphi + W_x, \\ \dot{y} &= V \cos \varphi - W_y, \\ \dot{\varphi} &= \omega u_1(t), \\ \dot{V} &= \omega u_2(t).\end{aligned}\tag{1}$$

Объект переводится из начальной точки $(x_0, y_0, \varphi_0, V_0)$ в область $(0, 0, \cos \varphi(T) = 1, V_0)$, где T – конечный момент времени, за минимальное время. Вектор управления $(u_1(t), u_2(t))^*$, удовлетворяющий ограничениям $|u_1(t)| \leq 1, |u_2(t)| \leq 1$ выбирается из класса кусочно-непрерывных функций. Курсовой угол удовлетворяет ограничению $|\Delta \varphi(t)| \leq 2\pi$. Скорость объекта лежит в интервале $V(t) \in [V_0, V_{max}]$, $V_{max} = V_0 + a \frac{2\pi}{\omega}$. Угловая скорость ω , ускорение a , возмущения W_x, W_y – положительные постоянные, причем W_x, W_y не равны нулю одновременно. Следует отметить, что отличие данной

Иванов Артемий Валерьевич – студент, Санкт-Петербургский государственный университет; e-mail: st062236@student.spbu.ru, тел.: +7(981)981-06-99

Моисеев Игорь Анатольевич – старший преподаватель, Санкт-Петербургский государственный университет; e-mail: i.a.moiseev@spbu.ru, тел.: +7(921)906-60-44

постановки задачи от рассматриваемых ранее заключается в ограничениях, наложенных на курсовой угол и скорость объекта. В силу того, что для данной постановки задачи количество точек переключения управлений для оптимальной траектории не определено, то рассмотрим частный случай, когда существует допустимое и удовлетворяющее необходимому условию оптимальности движение с одним переключением курсового угла. Отметим, что в точке переключения угла не происходит переключения управления скоростью. При рассмотрении задачи воспользуемся принципом максимума [13].

Итак, гамильтониан системы (1)

$$H(x, y, \varphi, V, \psi, u) = \psi_1(V \sin \varphi + W_x) + \psi_2(V \cos \varphi - W_y) + \psi_3 \omega u_1 + \psi_4 \omega u_2.$$

Система, сопряженная (1), имеет вид

$$\begin{aligned} \dot{\psi}_1 &= 0, \\ \dot{\psi}_2 &= 0, \\ \dot{\psi}_3 &= -(\psi_1 V \cos \varphi - \psi_2 V \sin \varphi), \\ \dot{\psi}_4 &= -(\psi_1 \sin \varphi + \psi_2 V \cos \varphi). \end{aligned} \tag{2}$$

Интегрируя первые три уравнения системы (2), получим

$$\begin{aligned} \psi_1 &= C_1, \quad \psi_2 = C_2, \\ \psi_3(t) &= \int (C_2 V \sin \varphi - C_1 V \cos \varphi) dt = \\ &= C_2(x - W_x t) - C_1(y + W_y t) + C_3 \end{aligned}$$

Пусть $[\tau, t]$ – интервал постоянства управлений u_1 и u_2 . Интегрируя

первые два уравнения из (1), получим

$$\begin{aligned}
x(t) &= \int_{\tau}^t (V \sin \varphi + W_x) dt = x(\tau) + W_x(t - \tau) - \frac{V(t) \cos \varphi(t)}{\omega u_1(\tau)} + \\
&\quad + \frac{V(\tau) \cos \varphi(\tau)}{\omega u_1(\tau)} + \frac{a u_2(\tau) \sin \varphi(t)}{\omega^2 u_1^2(\tau)} - \frac{a u_2(\tau) \sin \varphi(\tau)}{\omega^2 u_1^2(\tau)}, \\
y(t) &= \int_{\tau}^t (V \cos \varphi - W_y) dt = y(\tau) - W_y(t - \tau) + \frac{V(t) \sin \varphi(t)}{\omega u_1(\tau)} - \\
&\quad - \frac{V(\tau) \sin \varphi(\tau)}{\omega u_1(\tau)} + \frac{a u_2(\tau) \cos \varphi(t)}{\omega^2 u_1^2(\tau)} - \frac{a u_2(\tau) \cos \varphi(\tau)}{\omega^2 u_1^2(\tau)}.
\end{aligned} \tag{3}$$

Имеет место следующая теорема.

Теорема. Пусть для движения управляемого объекта, описываемого системой (1), существуют траектории с одной точкой переключения управления курсовым углом, являющиеся решением поставленной задачи. Тогда эти траектории имеют одну или же три точки переключения управления скоростью.

Доказательство. Пусть время начала движения равно нулю и $x_i = x(t_i)$, $y_i = y(t_i)$, $\varphi_i = \varphi(t_i)$, $V_i = V(t_i)$, $i = \overline{1, n}$, – значения фазовых координат в моменты переключения управления $u_2(t)$. Рассмотрим временной интервал $[0, t_p]$, где t_p – момент переключения $u_1(t)$. Выясним, сколько может быть точек переключения управления скоростью в данном интервале. Пусть их больше чем одна. С учетом постоянства гамильтониана как функции времени [13], имеем

$$\begin{aligned}
H(t_2) - H(t_1) &= C_1(V_2 \sin \phi_2 + W_x) + C_2(V_2 \cos \phi_2 - W_y) + \\
&\quad + u_1^0 \omega (C_2(x_2 - W_x t_2) - C_1(y_2 + W_y t_2) + C_3) - \\
&\quad - C_1(V_1 \sin \varphi_1 + W_x) - C_2(V_1 \cos \varphi_1 - W_y) - \\
&\quad - u_1^0 \omega (C_2(x_1 - W_x t_1) - C_1(y_1 + W_y t_1) + C_3) = \\
&= C_1(V_2 \sin \varphi_2 - V_1 \sin \varphi_1) + C_2(V_2 \cos \varphi_2 - V_1 \cos \varphi_1) + \\
&\quad + u_1^0 \omega (C_2((x_2 - x_1) - W_x(t_2 - t_1)) - \\
&\quad - C_1((y_2 - y_1) + W_y(t_2 - t_1))).
\end{aligned} \tag{4}$$

С учетом (3) получим, что

$$\begin{aligned} u_1^0 \omega C_2 (x_2 - x_1 - W_x(t_2 - t_1)) &= C_2 \left(V_1 \cos \varphi_1 - V_2 \cos \varphi_2 + \right. \\ &\quad \left. + \frac{au_2^1 \sin \varphi_2}{u_1^0 \omega} - \frac{au_2^1 \sin \varphi_1}{u_1^0 \omega} \right), \\ u_1^0 \omega C_1 (y_2 - y_1 + W_y(t_2 - t_1)) &= C_1 \left(-V_1 \sin \varphi_1 + V_2 \sin \varphi_2 + \right. \\ &\quad \left. + \frac{au_2^1 \cos \varphi_2}{u_1^0 \omega} - \frac{au_2^1 \cos \varphi_1}{u_1^0 \omega} \right) \end{aligned}$$

и (4) примет следующий вид:

$$\begin{aligned} H(t_2) - H(t_1) &= C_1 V_2 \sin \varphi_2 - C_1 V_1 \sin \varphi_1 + C_2 V_2 \cos \varphi_2 - \\ &\quad - C_2 V_1 \cos \varphi_1 + C_2 V_1 \cos \varphi_1 - C_2 V_2 \cos \varphi_2 + C_2 \frac{au_2^1 \sin \varphi_2}{u_1^0 \omega} - \\ &\quad - C_2 \frac{au_2^1 \sin \varphi_1}{u_1^0 \omega} + C_1 V_1 \sin \varphi_1 - C_1 V_2 \sin \varphi_2 - C_1 \frac{au_2^1 \cos \varphi_2}{u_1^0 \omega} + \\ &\quad + C_1 \frac{au_2^1 \cos \varphi_1}{u_1^0 \omega} = \frac{au_2^1}{u_1^0 \omega} (\sin \varphi_2 C_2 - \cos \varphi_2 C_1 - \sin \varphi_1 C_2 + \\ &\quad + \cos \varphi_1 C_1) = \frac{au_2^1}{u_1^0 \omega} (\sin(\varphi_2 - \alpha) - \sin(\varphi_1 - \alpha)) = 0, \end{aligned}$$

где $\alpha = \arctg(C_1/C_2)$. Отсюда следует, что

$$\sin(\varphi_2 - \alpha) - \sin(\varphi_1 - \alpha) = 2 \sin \frac{\varphi_2 - \varphi_1}{2} \cos \frac{\varphi_2 + \varphi_1 - 2\alpha}{2} = 0$$

и $\sin \frac{\varphi_2 - \varphi_1}{2} = 0$, $\varphi_2 = \varphi_1 + 2\pi k$, $k = 0, \pm 1$, что противоречит ограничению, наложенному на курсовой угол. Тогда нулю равен второй множитель и $\varphi_2 = 2\alpha - \varphi_1 + \pi + 2\pi k$, $k = 0, \pm 1$. Допустим, что существует третья точка переключения управления скоростью. Тогда применив те же рассуждения, получим, что $\varphi_3 = 2\alpha - \varphi_2 + \pi + 2\pi m$, $k = 0, \pm 1$, $\varphi_3 = 2\alpha - 2\alpha + \varphi_1 - \pi - 2\pi k + \pi + 2\pi m = \varphi_1 + 2\pi(m - k)$, что противоречит ограничениям, наложенным на курсовой угол. Таким образом, на интервале $[0, t_p]$, не может быть более двух точек переключения $u_2(t)$. Далее рассмотрим интервал $[t_p, T]$. Применяя те же рассуждения, что и для первого интервала, получим, что и в этом

случае не может быть больше двух точек переключения управления $u_2(t)$. Таким образом получим, что общее количество переключений не более четырех. Теперь воспользуемся тем фактом, что минимальная скорость объекта равна начальной скорости. Это означает, что в начальный момент времени $u_2^0 = +1$, а в конечный момент $u_2^T = -1$, т. е. $u_2^n = -1$, где t_n – последняя точка переключения управления. Тогда возможны следующие варианты:

1. Пусть в интервале $[0, t_p]$ нет переключений, тогда первая точка переключения $t_1 \in [t_p, T]$. После переключения $u_2^1 = -1$. Далее движение с данным управлением будет решением задачи (1). Если предположить, что это не так, тогда значения управлений в следующих точках переключения равны $u_2^2 = +1, u_2^3 = -1$, т. е. в интервале три точки переключения, а это противоречит доказанному выше результату.

2. В интервале $[0, t_p]$ одна точка переключения. В данном случае происходит переключение с управления $u_2^0 = +1$ на управление $u_2^1 = -1$. Тогда возможны следующие случаи: больше точек переключения нет и $u_2^n = u_2^1 = -1$; во втором случае в интервале $[t_p, T]$ может быть два переключения $u_2^2 = +1, u_2^3 = -1$.

3. В интервале $[0, t_p]$ две точки переключения, т. е. $u_2^1 = -1, u_2^2 = +1$. В интервале $[t_p, T]$ имеется одна точка переключения $u_2^3 = -1$ (аналогично пункту 1).

Таким образом, мы убедились, что искомые траектории имеют одну или же три точки переключения управления $u_2(t)$.

Теорема доказана.

Следствие. Для движения управляемого объекта, описываемого системой (1), искомые траектории содержат следующие наборы управлений:

$$1. \quad u_1^0 = +1, t \in [0, t_p], \quad u_1^1 = -1, t \in [t_p, T].$$

$$1.1. \quad u_2^0 = +1, t \in [0, t_p] \cup [t_p, t_1], \quad u_2^1 = -1, t \in [t_1, T].$$

$$1.2. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_p] \cup [t_p, T].$$

$$1.3. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_p] \cup [t_p, t_2],$$

$$u_2^2 = +1, t \in [t_2, t_3], \quad u_2^3 = -1, t \in [t_3, T].$$

$$1.4. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_2],$$

$$u_2^2 = +1, t \in [t_2, t_p] \cup [t_p, t_3], \quad u_2^3 = -1, t \in [t_3, T].$$

$$2. \quad u_1^0 = -1, t \in [0, t_p], \quad u_1^1 = +1, t \in [t_p, T].$$

$$2.1. \quad u_2^0 = +1, t \in [0, t_p] \cup [t_p, t_1], \quad u_2^1 = -1, t \in [t_1, T].$$

$$2.2. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_p] \cup [t_p, T].$$

$$2.3. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_p] \cup [t_p, t_2],$$

$$u_2^2 = +1, t \in [t_2, t_3], \quad u_2^3 = -1, t \in [t_3, T].$$

$$2.4. \quad u_2^0 = +1, t \in [0, t_1], \quad u_2^1 = -1, t \in [t_1, t_2],$$

$$u_2^2 = +1, t \in [t_2, t_p] \cup [t_p, t_3], \quad u_2^3 = -1, t \in [t_3, T].$$

Заключение. Таким образом, используя результаты теоремы, можно заранее найти траектории движения при заданных начальных данных, удовлетворяющие необходимым условиям оптимальности. Следует также проверить возможность существования при тех же начальных данных траекторий с другим количеством точек переключения управления курсовым углом, удовлетворяющим принципу максимума, и выбрать из них оптимальную.

Литература

1. Isaacs R. Games of pursuit. Scientific report of the RAND Corporation. Santa Monica: RAND Corporation, 1951. 16 p.
2. Айзекс Р. Дифференциальные игры / пер. с англ. В. И. Аркина, Э. Н. Симаковой; под ред. М. И. Зеликина. М.: Мир, 1967. 384 с.

3. Двуреченский П. Е. Алгоритмы построения эpsilon-оптимальных стратегий в нелинейных дифференциальных играх на плоскости: автореф. дис. канд. физ.-мат. наук. Долгопрудный: МФТИ, 2013. 24 с.
4. Пацко В. С., Турова В. Л. Игра «шофер-убийца» и ее модификации // Вестник Удмурт. ун-та. Вып. 2. Ижевск. 2008. С. 105–110.
5. Пацко В. С., Турова В. Л. Игра «шофер-убийца»: история и современные исследования / Научные доклады / Рос. акад. наук, Урал. отд-ние, Ин-т математики и механики. УрО РАН, Екатеринбург. 2009. 43 с.
6. Reeds J. A., Shepp L. A. Optimal paths for a car that goes both forwards and backwards // Pacific J. Math. 1990. Vol. 145. No 2. P. 367–393.
7. Patsko V. S., Turova V. L. Level sets of the value function in differential games with the homicidal chauffeur dynamics // Game Theory and Applications. 2007. Vol. 12. P. 123–152.
8. Кулешов А. В. Некоторые способы численного решения дифференциальной игры «шофер-убийца» // Известия Института математики и информатики УдГУ. 2012. Вып. 1 (39). С. 73.
9. Patsko V. S., Turova V. L. Numerical study of the homicidal chauffeur differential game with the reinforced pursuer // Game Theory and Applications. 2007. Vol. 12. P. 12–152.
10. Розов Н. Х. Постановка задачи оптимального управления. Математика на службе инженера // Основы теории оптимального управления: сб. статей. М.: Знание, 1973. С. 6–27.
11. Бердышев Ю. И. Синтез оптимального по быстродействию управления для одной нелинейной системы четвертого порядка // Прикладная математика и механика. 1975. Т. 39. Вып. 6. С. 985–994.
12. Бердышев Ю. И. Синтез оптимального по быстродействию управления движением материальной точки в среде с сопротивлением. автореф. дис. канд. физ.-мат. наук. Свердловск: Урал. науч. центр, 1978. 18 с.

13. Понтрягин Л. С., Болтянский В. Г., Гамкрелидзе Р. В., Мищенко Е. Ф. Математическая теория оптимальных процессов. М.: Наука, 1969. 384 с.
14. Гриднев Л. В., Моисеев И. А. Частный случай синтеза траекторий нелинейной системы четвертого порядка, удовлетворяющих необходимым условиям оптимальности // Процессы управления и устойчивость. 2016. Т. 3. № 1. С. 85–89.
15. Гриднев Л. В., Моисеев И. А. Синтез траекторий движения объекта, удовлетворяющих необходимым условиям оптимальности в задаче быстрогодействия // Современные методы прикладной математики, теории управления и компьютерных технологий (ПМТУКТ-2017). Сборник трудов IX международной конференции. 2017. С. 126–129.

УДК 517.929.4

Кудряков Д. А.

Анализ устойчивости линейных систем с запаздыванием и неопределенными параметрами

Рекомендовано к публикации доцентом Александровой И. В.

1. Введение. Целью настоящей работы является разработка новых методов исследования устойчивости линейных стационарных дифференциально-разностных систем с параметрами методом функционалов Ляпунова–Красовского. В работе [1] показано, что для анализа устойчивости таких систем положительную определенность функционалов с заданной производной достаточно проверять только на множестве функций, удовлетворяющих условию $\|\varphi(\theta)\| \leq \|\varphi(0)\|$, $\theta \in [-h, 0]$. В данной работе показано, что это условие может быть применено и для проверки отрицательной определенности производной функционалов вдоль решений систем, зависящих от параметров. При этом используются функционалы, определенные на комплексных функциях.

Введем обозначения: $\|x\|$ – евклидова норма вектора $x \in \mathbb{C}^n$; $PC([-h, 0], \mathbb{C}^n)$ – пространство кусочно-непрерывных функций, действующих из $[-h, 0]$ в \mathbb{C}^n , с нормой $\|\varphi\|_h = \sup_{\theta \in [-h, 0]} \|\varphi(\theta)\|$; $A^* = (\overline{A})^T$ – эрмитово сопряжение матрицы A ; i – мнимая единица; $\Re(z)$ – действительная часть $z \in \mathbb{C}$; E – единичная матрица.

2. Постановка задачи. Рассмотрим систему дифференциальных уравнений с одним запаздыванием вида

$$\frac{dx(t)}{dt} = Ax(t) + Bx(t-h), \quad t \geq 0, \quad (1)$$

где $x(t) \in \mathbb{C}^n$; $A, B \in \mathbb{R}^{n \times n}$; $h > 0$. Пусть $\varphi \in PC([-h, 0], \mathbb{C}^n)$ – начальная функция: $x(\theta, \varphi) = \varphi(\theta)$, $\theta \in [-h, 0]$; $x_t(\varphi) : \theta \mapsto x(t+\theta, \varphi)$, $\theta \in [-h, 0]$, – состояние системы.

Кудряков Дмитрий Александрович – студент, Санкт-Петербургский государственный университет; e-mail: st068497@student.spbu.ru, тел.: +7(922)218-75-90

Предположим, что система (1) экспоненциально устойчива. Рассмотрим возмущенную систему

$$\frac{dy(t)}{dt} = \mathcal{A}(p)y(t) + \mathcal{B}(p)y(t - \mathfrak{h}(p)), \quad t \geq 0, \quad (2)$$

где $y(t) \in \mathbb{C}^n$, матрицы \mathcal{A} , \mathcal{B} и запаздывание \mathfrak{h} непрерывно зависят от набора параметров $p \in \mathbb{R}^m$, $\mathcal{A}(0) = A$, $\mathcal{B}(0) = B$, $\mathfrak{h}(0) = h$.

Задача состоит в нахождении условий, при которых система (2) остается экспоненциально устойчивой.

Для решения задачи будем использовать множество

$$S = \{\varphi \in PC([-h(p), 0], \mathbb{C}^n) : \|\varphi(\theta)\| \leq \|\varphi(0)\|, \theta \in [-h(p), 0]\}.$$

3. Предварительные сведения. Введем базовые определения.

Определение 1 [2]. Система (1) называется *экспоненциально устойчивой*, если существуют $\gamma \geq 1$ и $\sigma > 0$ такие, что для любых начальных функций $\varphi \in PC([-h, 0], \mathbb{R}^n)$ выполняется неравенство $\|x(t, \varphi)\| \leq \gamma e^{-\sigma t} \|\varphi\|_h$, $t \geq 0$.

Определение 2 [2]. Число $s \in \mathbb{C}$ называется *собственным числом* системы (1), если оно является корнем уравнения $\det(sE - A - e^{-sh}B) = 0$.

Теорема 1 [2]. Система (1) экспоненциально устойчива тогда и только тогда, когда действительные части всех ее собственных чисел отрицательны.

Определение 3 [3]. Матрица $U(\tau)$, удовлетворяющая

$$\frac{d}{d\tau}U(\tau) = U(\tau)A + U(\tau - h)B, \quad U(-\tau) = U^T(\tau), \quad \tau \geq 0,$$

и $U(0)A + U(-h)B + A^T U(0) + B^T U(h) = -W$, называется *матрицей Ляпунова* системы (1), ассоциированной с матрицей W . Здесь при $\tau = 0$ имеется в виду правая производная $U(\tau)$.

Замечание 1. Матрица Ляпунова вещественна и непрерывна, если система (1) экспоненциально устойчива [3].

4. Функционал с заданной производной, определенный на комплексных функциях. Пусть $W \in \mathbb{R}^{n \times n}$ положительно определена. Функционал $v_0(\varphi)$ такой, что

$$\frac{d}{dt}v_0(x_t) = -x^*(t)Wx(t), \quad t \geq 0,$$

определенный на комплексных функциях, имеет вид

$$\begin{aligned} v_0(\varphi) &= \varphi^*(0)U(0)\varphi(0) + \varphi^*(0) \int_{-h}^0 U(-h-\theta)B\varphi(\theta)d\theta + \\ &+ \int_{-h}^0 \varphi^*(\theta)B^T U(h+\theta)d\theta\varphi(0) + \\ &+ \int_{-h}^0 \varphi^*(\theta_1)B^T \int_{-h}^0 U(\theta_1-\theta_2)B\varphi(\theta_2)d\theta_2d\theta_1 \end{aligned}$$

и является непосредственным обобщением выражения, приведенного в [3] для случая вещественных начальных функций.

Замечание 2. Функционал $v_0(\varphi)$ и его производная, определенные на комплексных функциях, принимают только вещественные значения.

5. Основной результат. Получено следующее утверждение.

Лемма. Если система (1) экспоненциально устойчива, а производная функционала v_0 вдоль решений системы (2) удовлетворяет условию

$$\frac{d}{dt}v_0(y_t) = -w(y_t),$$

где функционал $w(\varphi) \geq w_0\|\varphi(0)\|^2$ для любых $\varphi \in S$, $w_0 > 0$, то система (2) не имеет собственных чисел на мнимой оси.

Доказательство. Предположим, что система (2) имеет собственное число $s = i\beta$, $\beta \in \mathbb{R}$. Следовательно, система (2) имеет решение $y(t) = e^{i\beta t}C$, где $C \in \mathbb{C}^n$, $\|C\| \neq 0$. Заметим, что $y_t \in S$, так как $\|y(t)\| = \|C\|$ для любого $t \geq 0$.

Рассмотрим функционал на этом решении:

$$\begin{aligned} v_0(y_t) &= e^{-i\beta t}C^*U(0)e^{i\beta t}C + e^{-i\beta t}C^* \int_{-h}^0 U(-h-\theta)Be^{i\beta(t+\theta)}Cd\theta + \\ &+ \int_{-h}^0 e^{-i\beta(t+\theta)}C^*B^T U(h+\theta)d\theta e^{i\beta t}C + \\ &+ \int_{-h}^0 e^{-i\beta(t+\theta_1)}C^*B^T \int_{-h}^0 U(\theta_1-\theta_2)Be^{i\beta(t+\theta_2)}Cd\theta_2d\theta_1. \end{aligned}$$

Очевидно, что $v_0(y_t)$ не зависит от t . А значит,

$$-w(y_t) = \frac{d}{dt}v_0(y_t) = 0.$$

Поэтому $w(y_t) = 0 \geq w_0 \|y(t)\|^2$, тогда $\|y(t)\| = \|C\| = 0$, что противоречит определению собственного числа s .

Лемма доказана.

В силу непрерывной зависимости нулей квазиполиномов от параметров из леммы вытекает следующая теорема.

Теорема 2. *Если функционал v_0 удовлетворяет условиям леммы, то система (2) экспоненциально устойчива.*

Замечание 3. Теорема 2 означает, что для проверки устойчивости системы (2) достаточно исследовать отрицательную определенность производной функционала только на множестве S .

Нетрудно заметить, что в силу линейности системы (2) в доказательстве леммы дополнительно можно считать, что $\|C\| = 1$. Поэтому справедливо

Следствие. *Лемма и теорема 2 остаются справедливыми, если в них заменить множество S на множество*

$$\tilde{S} = \left\{ \varphi \in PC([-h(p), 0], \mathbb{C}^n) : \|\varphi(\theta)\| = 1, \quad \theta \in [-h(p), 0] \right\}.$$

6. Применение теоремы 2 к анализу робастной устойчивости. Рассмотрим частный случай системы (2):

$$\frac{dy(t)}{dt} = (A + \Delta_0)y(t) + (B + \Delta_1)y(t-h), \quad t \geq 0, \quad (3)$$

где $\Delta_0, \Delta_1 \in \mathbb{R}^{n \times n}$ – неизвестные матрицы.

Рассмотрим производную функционала $v_0(\varphi)$ вдоль решений системы (3):

$$\begin{aligned} \frac{dv_0(y_t)}{dt} &= -y^*(t)W y(t) + l(y_t), \\ l(y_t) &= y^*(t) [\Delta_0^T U(0) + U(0)\Delta_0] y(t) + \\ &+ 2\Re \left[y^*(t)U(0)\Delta_1 y(t-h) + \right. \\ &\left. + (y^*(t)\Delta_0^T + y^*(t-h)\Delta_1^T) \int_{t-h}^t U(-h-\xi+t)B y(\xi) d\xi \right]. \quad (4) \end{aligned}$$

Выражение (4) является прямым обобщением выражения, приведенного в [3], на случай комплексных аргументов.

Применим теорему 2 к анализу устойчивости системы (3). Пусть $w(\varphi) = \varphi^*(0)W\varphi(0) - l(\varphi)$, тогда

$$w(\varphi) \geq \left(\lambda_{\min}(W) - 2(\|\Delta_0\| + \|\Delta_1\|)M(1 + h\|B\|) \right) \|\varphi(0)\|^2, \quad \varphi \in S,$$

где $M = \max_{\tau \in [0, h]} \|U(\tau)\| = \|U(0)\|$ (см. [4]), $\lambda_{\min}(W)$ – минимальное собственное число матрицы W . Следовательно, по теореме 2, если

$$\lambda_{\min}(W) - 2(\|\Delta_0\| + \|\Delta_1\|)M(1 + h\|B\|) > 0,$$

то система (3) экспоненциально устойчива. Этот вывод совпадает с полученным ранее в работе [5].

7. Заключение. В статье получены новые условия устойчивости линейных стационарных дифференциально-разностных систем, зависящих от параметров, которые обобщают известные ранее условия робастной устойчивости [5]. Их преимуществом является свобода в выборе зависимостей параметров правой части системы. Поэтому основной результат можно применять не только к системам вида (3), но и к системам с более сложной зависимостью от параметров.

Литература

1. Medvedeva I. V., Zhabko A. P. Synthesis of Razumikhin and Lyapunov – Krasovskii approaches to stability analysis of time-delay systems // Automatica. 2015. No 51. P. 372–377.
2. Bellman R. E., Cooke K. L. Differential-Difference Equations. Santa Monica: RAND Corporation, 1963. 474 p.
3. Kharitonov V. L. Time-delay systems. Lyapunov functionals and matrices. Basel: Birkhauser, 2013. 311 p.
4. Egorov A. V., Mondie S. The delay Lyapunov matrix in robust stability analysis of time-delay systems // IFAC-PapersOnLine. 2015. No 48. P. 245–250.
5. Medvedeva I. V., Zhabko A. P. A novel approach to robust stability analysis of linear time-delay systems // IFAC-PapersOnLine. 2015. No 48. P. 233–238.

УДК 519.853

Миронов А. С.

Обобщение задачи Данскина

Рекомендовано к публикации профессором Малозёмовым В. Н.

1. Введение. Рассмотрим следующую сепарабельную экстремальную задачу:

$$\begin{aligned} F(x) &= \sum_{i=1}^n f_i(x_i) \rightarrow \max, \\ x_i &\geq 0, \quad i = 1, \dots, n, \\ \sum_{i=1}^n x_i &= A. \end{aligned} \tag{1}$$

Здесь f_i – непрерывно дифференцируемые, строго вогнутые, возрастающие на $[0, A]$ функции, A – положительная константа. В задаче, рассматриваемой в [1], функции

$$f_i(x_i) = a_i(1 - e^{-b_i x_i}),$$

где $a_i, b_i > 0$. Эта задача является частным случаем задачи (1).

Для ее решения мы воспользовались леммой, известной как *лемма Гиббса* [2]. Приведем здесь ее усиленный вариант, достаточный для дальнейших рассуждений.

Лемма 1. Пусть в задаче (1) функции f_i вогнутые. Тогда для того чтобы план $x^* = (x_1^*, \dots, x_n^*)$ был оптимальным для задачи (1), необходимо и достаточно существование числа λ такого, что

$$f'_i(x_i^*) = \lambda, \quad x_i^* > 0, \tag{2}$$

$$f'_i(x_i^*) \leq \lambda, \quad x_i^* = 0. \tag{3}$$

2. Сведение к скалярному уравнению. Переформулируем лемму 1 в виде, более удобном для дальнейших рассуждений.

Заметим, что в силу дифференцируемости и строгой вогнутости функций $f_i(t)$ их производные $g_i(t) := f'_i(t)$ – непрерывные, строго убывающие функции.

Миронов Артем Сергеевич – студент, Санкт-Петербургский государственный университет; e-mail: st063750@student.spbu.ru, тел.: +7(929)110-28-27

Критерии (2) и (3) для плана $x = (x_1, \dots, x_n)$ можно переписать в следующем виде:

$$x_i = 0, \quad g_i(0) \leq \lambda, \quad (4)$$

$$x_i > 0, \quad g_i(0) > \lambda. \quad (5)$$

Действительно, при $g_i(0) \leq \lambda$ необходимо $x_i = 0$, так как иначе в силу (2) имеем $\lambda = g_i(x_i) < g_i(0)$, что противоречит предположению. При $g_i(0) > \lambda$ имеем $x_i > 0$, так как иначе $g_i(x_i) = g_i(0) \leq \lambda$ в силу (3), что снова приводит к противоречию.

Поскольку g_i строго монотонны и непрерывны, у них существуют обратные функции g_i^{-1} . Поэтому на основании критерия (2) формулу (5) можно записать как

$$x_i = g_i^{-1}(\lambda), \quad g_i(0) > \lambda. \quad (6)$$

Формулы (6) и (4) можно объединить в одну:

$$x_i = h_i(\lambda), \quad (7)$$

где h_i – расширение g_i^{-1} , определяемое следующим образом:

$$h_i(y) = \begin{cases} g_i^{-1}(y), & \text{если } g_i(A) \leq y \leq g_i(0), \\ 0, & \text{если } y > g_i(0). \end{cases}$$

На рис. 1 представлен график, поясняющий определение h_i . При этом график $x = h_i(y)$ следует понимать как объединение графика $y = g_i(x)$ и луча, идущего вертикально вверх из точки $(M_i, 0)$.

Напомним, что в задаче (1) присутствует ограничение

$$\sum_{i=1}^n x_i = A.$$

Чтобы быть планом задачи (1), вектор x , задаваемый формулой (7), обязан удовлетворять и этому ограничению.

Таким образом, можно сформулировать следующую лемму.

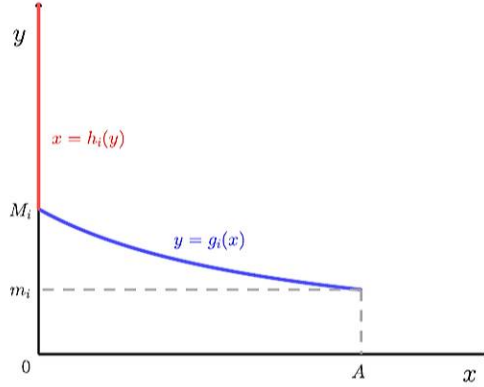


Рис. 1. Графики g_i и h_i

Лемма 2. Если число λ является решением уравнения

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

то план, задаваемый формулой (7), удовлетворяет критериям (2) и (3), а значит является оптимальным.

3. Исследование уравнения. Для удобства определим функцию

$$h(y) = \sum_{i=1}^n h_i(y). \quad (9)$$

Перенумеруем индексы таким образом, чтобы выполнялось условие $g_1(0) \geq \dots \geq g_n(0)$. Введем обозначения

$$M_i := g_i(0), \quad m_i := g_i(A), \quad m = \max_{i=1, \dots, n} m_i = m_{i_0}.$$

На рис. 2 представлен график $x = h(y)$, где виден ее «кусочно-гладкий» вид.

Функция $h(y)$ определена только при $y \geq m$. Причем для любого $y \in [m, M_1]$ можно записать формулу (9) несколько иначе, отбросив заведомо нулевые слагаемые:

$$h(y) = \sum_{i=1}^k g_i^{-1}(y), \quad y \in [M_{k+1}, M_k]. \quad (10)$$

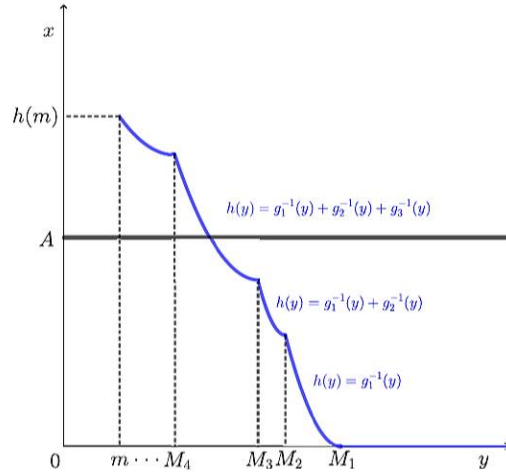


Рис. 2. График функции h

Действительно, если $i > k$, то $y > M_i$, а значит $h_i(y) = 0$; больше членов отбросить не удастся. Из последнего неравенства, неотрицательности значений h_i и подстановки $y = m$ в формулу (10) имеем

$$h(m) = \sum_{i=1}^k g_i^{-1}(m) > g_{i_0}^{-1}(m) = A.$$

В силу определения h_i ясно, что функция h непрерывна, строго убывает на $[m, M_1]$ и $h(M_1) = 0$. Также известно, что $h(m) > A$. Из этого следует, что уравнение (8) имеет единственное решение на $[m, M_1]$.

4. Решение уравнения. Теперь перейдем к нахождению λ . Вычислим

$$h(M_k) = \sum_{i=1}^{k-1} g_i^{-1}(M_k), \quad k = 2, \dots, n, \quad h(M_1) = 0.$$

Обозначим за k^* наибольший индекс, для которого выполняется $h(M_{k^*}) \leq A$. Тогда искомое значение λ найдется по формуле (10) на промежутке $[M_{k^*+1}, M_{k^*}]$, если $M_{k^*+1} > m$, или на $[m, M_{k^*}]$ в противном случае. После нахождения λ^* оптимальный план задачи (1) найдется по формуле (7).

Заметим, что если $M_i \leq m$, то $x_i^* = 0$ вне зависимости от значения λ ; это следует из формул (7) и (10).

5. Пример. Применим вышеизложенную схему к следующей задаче:

$$\begin{aligned} \sum_{k=1}^5 \ln(k + x_k) &\rightarrow \max, \\ x_k &\geq 0, \quad k = 1, \dots, 5, \\ \sum_{i=1}^5 x_i &= 7. \end{aligned}$$

Иначе говоря, положим в задаче (1) $n = 5$, $A = 7$ и

$$f_k(t) = \ln(k + t), \quad k = 1, \dots, 5.$$

Вычислим производные функций-компонент и значения производных на концах промежутка $[0, A]$:

$$\begin{aligned} g_k(t) &= \frac{1}{k + t}, \\ M_k = g_k(0) &= \frac{1}{k}, \quad m_k = g_k(A) = \frac{1}{k + 7}, \\ &k = 1, \dots, 5. \end{aligned}$$

Поскольку M_k уже располагаются в убывающем порядке, перенумерации не требуется. Нижняя граница области определения h равна $m_1 = 1/8$. Выпишем обратные функции к g_k , а также функцию h :

$$\begin{aligned} g_k^{-1}(y) &= \frac{1}{y} - k, \quad k = 1, \dots, 5; \\ h(y) &= \begin{cases} 0, & y \geq 1, \\ y^{-1} - 1, & 1/2 \leq y < 1, \\ 2y^{-1} - 3, & 1/3 \leq y < 1/2, \\ 3y^{-1} - 6, & 1/4 \leq y < 1/3, \\ 4y^{-1} - 10, & 1/5 \leq y < 1/4, \\ 5y^{-1} - 15, & 1/8 \leq y < 1/5. \end{cases} \end{aligned}$$

Значения $h(M_2) = 1$, $h(M_3) = 3$, $h(M_4) = 6$, $h(M_5) = 10$. Поскольку $A \in [h(M_4), h(M_5)]$, то $k^* = 4$ и $\lambda \in [M_5, M_4]$. Из этого следует, что

$x_5^* = 0$ и $x_k^* > 0$ для $k = 1, 2, 3, 4$. Решая уравнение $4\lambda^{-1} - 10 = A$, получаем $\lambda = 4/17$. Используя формулу (7), по значению λ вычислим компоненты оптимального плана:

$$x_1^* = \frac{13}{4}, \quad x_2^* = \frac{9}{4}, \quad x_3^* = \frac{5}{4}, \quad x_4^* = \frac{1}{4}, \quad x_5^* = 0.$$

На рис. 3 представлен график функции h .

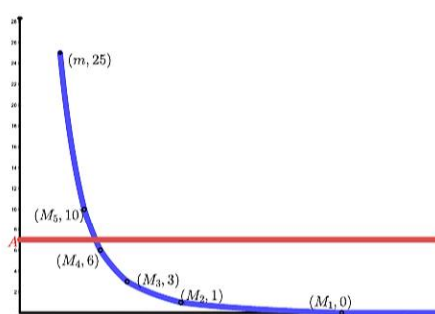


Рис. 3. График функции h

6. Заключение. С помощью леммы Гиббса задача (1) сведена к решению скалярного уравнения. Представлен алгоритм решения этого уравнения и формула для перехода к оптимальному плану задачи (1).

Литература

1. Данскин Дж. М. Теория максимина и ее приложение к задачам распределения вооружения. М.: Изд-во «Советское радио», 1970. 200 с.
2. Малозёмов В. Н., Тамасян Г. Ш. Лемма Гиббса и ее приложения [Электронный ресурс]: URL:<http://arpmath.spbu.ru/cnsa/gerp17.shtml\#1010> (дата обращения 22.04.21).

UDC 681.5

Kastner A., Gröll L., Hagenmeyer V.

Abstractions in modeling of switching DC/AC power converters

1. Introduction. The introduction of power generation from renewables and new storage and transmission technologies is leading to increasing usage of DC/AC power converters. Switching converters offer control potential for both active and reactive power flow, but the high harmonic content of currents causes additional power dissipation, and stability issues can arise from interactions with the grid impedance and between converters. Control techniques need to be developed based on models with an appropriate level of abstraction.

The present paper presents an overview of the relationships between mathematical models of power converters with different levels of detail from the viewpoint of systems theory and demonstrates their different dynamic behavior in simulations.

2. System Description. The circuit diagram of a three-phase two-level voltage source converter (VSC) with three-wire grid connection is depicted in Fig. 1. The AC side voltages and currents are subject to the conditions

$$i_1 + i_2 + i_3 = 0, \quad v_{12} + v_{23} + v_{31} = 0 \quad (1)$$

and typically the Clarke transform is used to obtain the two-dimensional representation

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 \\ 0 & \sqrt{3} & -\sqrt{3} \end{bmatrix} \begin{bmatrix} i_1 \\ i_2 \\ i_3 \end{bmatrix}, \quad \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \frac{1}{9} \begin{bmatrix} 3 & 0 & -3 \\ -\sqrt{3} & 2\sqrt{3} & -\sqrt{3} \end{bmatrix} \begin{bmatrix} v_{12} \\ v_{23} \\ v_{31} \end{bmatrix}. \quad (2)$$

Kastner Adam – graduate student, Karlsruhe Institute of Technology; e-mail: adam.kastner@kit.edu, phone: +49(721)6082-57-40

Gröll Lutz – professor, Karlsruhe Institute of Technology; e-mail: lutz.groell@kit.edu, phone: +49(721)6082-57-36

Hagenmeyer Veit – professor, Karlsruhe Institute of Technology; e-mail: veit.hagenmeyer@kit.edu, phone: +49(721)6082-92-00

This work was supported by the Helmholtz Association under the Innovation Pool Project “Energy Transition and Circular Economy”.

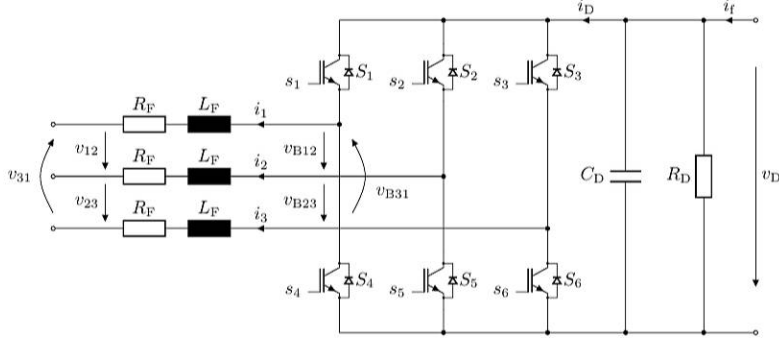


Figure 1. Circuit diagram of the voltage source converter

The switches S_l , $l = 1, \dots, 6$ are controlled by binary signals $s_l: \mathbb{R}^+ \rightarrow \{0, 1\}$, where the lower switches are operated in opposition to the upper switches, i.e.

$$s_{l+3} = 1 - s_l, \quad l = 1, 2, 3. \quad (3)$$

With 3 free switches, a total of $2^3 = 8$ configurations is possible. The latter are characterized by their configuration vectors \mathbf{c}_i , $i = 1, \dots, 8$ in the (α, β) -plane, obtained by applying the Clarke transform (2) to the three binary switch signals. The two configurations \mathbf{c}_7 and \mathbf{c}_8 where the three free switches are in the same position lead to the same system behavior. The instantaneous switching vector $\mathbf{s}: \mathbb{R}^+ \rightarrow \{\mathbf{c}_1, \dots, \mathbf{c}_8\}$ is chosen from all configuration vectors. Figure 2 shows the convex hull \mathbb{S} of the configuration vectors in grey, which is the area in which averages $\bar{\mathbf{s}} = [\bar{s}_\alpha \quad \bar{s}_\beta]^\top$ of switching vectors may lie. If only one switch configuration is active over the averaging period, the average equals the respective configuration vector, shown in blue.

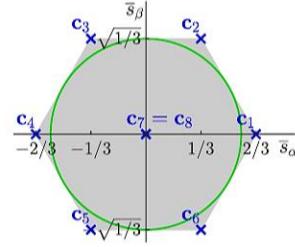


Figure 2. Switching in (α, β) -plane

A pulse-width modulation (PWM) system switches between different configurations such that the switching vector average $\bar{\mathbf{s}}$ approximate the PWM command vector $\mathbf{d} \in \mathbb{S}$. Harmonic oscillations with varying

amplitudes for \mathbf{d} are represented by circles of varying radius, the largest possible one within \mathbb{S} is depicted in green.

3. Models. The models are based on the assumptions that

- all passive components are linear and time-invariant with exactly known parameters,
- the switching elements have either infinite or zero resistance and can switch between these instantaneously,
- currents $[i_f \ i_\alpha \ i_\beta]^\top \in \mathbb{R}^3$ are not subject to physical limitations,
- the grid voltage vector stays within a circle $[v_\alpha \ v_\beta]^\top \in \mathcal{B}_{v_m}$, and the DC side voltage stays above a minimum value $v_D \geq \sqrt{3}v_m$; this ensures that the converter does not enter line-commutated operation.

3.1. Bilinear Model. A state-space model is derived with the DC side capacitor voltage and the AC side inductor currents as system state \mathbf{x} . The control input \mathbf{u} consists of the Clarke-transformed switching signals, and the uncontrolled input \mathbf{w} is the DC side input current and the grid voltages:

$$\mathbf{x} := [v_D \ i_\alpha \ i_\beta]^\top, \quad \mathbf{u} := [s_\alpha \ s_\beta]^\top, \quad \mathbf{w} := [i_f \ v_\alpha \ v_\beta]^\top. \quad (4)$$

Depending on the use of the model, \mathbf{w} may represent a known or unknown exogenous signal or a disturbance.

The dynamics can be described by the bilinear system

$$\dot{\mathbf{x}} = \mathbf{A}_0 \mathbf{x} + u_1 \mathbf{M}_1 \mathbf{x} + u_2 \mathbf{M}_2 \mathbf{x} + \mathbf{E} \mathbf{w}, \quad (5)$$

where the only bilinear interaction is between state and input. For the VSC, the matrices in (5) are

$$\mathbf{A}_0 = \begin{bmatrix} -\frac{1}{R_D C_D} & 0 & 0 \\ 0 & -\frac{R_F}{L_F} & 0 \\ 0 & 0 & -\frac{R_F}{L_F} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} \frac{1}{C_D} & 0 & 0 \\ 0 & -\frac{1}{L_F} & 0 \\ 0 & 0 & -\frac{1}{L_F} \end{bmatrix}, \quad (6a)$$

$$\mathbf{M}_1 = \begin{bmatrix} 0 & -\frac{3}{2C_D} & 0 \\ \frac{1}{L_F} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \mathbf{M}_2 = \begin{bmatrix} 0 & 0 & -\frac{3}{2C_D} \\ 0 & 0 & 0 \\ \frac{1}{L_F} & 0 & 0 \end{bmatrix}. \quad (6b)$$

3.2. Switched Linear Model. A switched model can be used as an alternative to the bilinear model. For each switch configuration

the system is described by linear dynamics. The active mode at each point of time is determined by the mode selection function $\sigma: \mathbb{R}^+ \rightarrow \Sigma$, $\Sigma = \{1, \dots, 8\}$. With state and uncontrolled input defined as in (4), the dynamics in each mode are described by

$$\dot{\mathbf{x}} = \mathbf{A}_{\sigma(t)} \mathbf{x} + \mathbf{E} \mathbf{w}. \quad (7)$$

For each configuration vector \mathbf{c}_j , $j = 1, \dots, 8$, the system matrix is obtained from (6) via $\mathbf{A}_j = \mathbf{A}_0 + c_{\alpha,j} \mathbf{M}_1 + c_{\beta,j} \mathbf{M}_2$. The control input can be defined to either directly represent the mode selection function

$$\mathbf{u} := \sigma \in \Sigma, \quad (8)$$

or to represent the command input to the static time-invariant PWM system

$$\sigma = \text{PWM}(t, \mathbf{u}), \quad \mathbf{u} := \mathbf{d} \in \mathbb{S}. \quad (9)$$

3.3. Averaged Model. State-space averaging for switched linear systems [1] is performed by averaging the right-hand side of the dynamics equation over a period T_S . For digitally controlled converters, T_S is the sampling period which is typically much smaller than the period of the grid fundamental. The state and uncontrolled input are replaced by their sliding averages

$$\bar{\mathbf{x}}(t) = \frac{1}{T_S} \int_{t-T_S}^t \mathbf{x}(\tau) d\tau. \quad (10)$$

When averaging the right-hand side of (7), we use the *small-ripple assumption* $\bar{\mathbf{x}} \approx \mathbf{x}$ [2, Ch. 4.2] and approximate the average of a product by the product of averages. This leads to

$$\dot{\bar{\mathbf{x}}} = \sum_{j=1}^8 \delta_j \mathbf{A}_j \bar{\mathbf{x}} + \mathbf{E} \bar{\mathbf{w}}, \quad (11)$$

where $\delta_j: \mathbb{R}^+ \rightarrow [0, 1]$, $j = 1, \dots, 8$ are the relative durations during which the respective mode $\sigma = j$ is active within the period T_S . They are subject to $\sum_{j=1}^8 \delta_j = 1$.

When using PWM, the average of the switching vector approximates the duty ratio command. Assuming that the continuous-time sliding average of the switching vector itself is the control input

$$\mathbf{u} := [\bar{\mathbf{s}}_{\alpha} \quad \bar{\mathbf{s}}_{\beta}]^{\top} \approx [d_{\alpha} \quad d_{\beta}]^{\top} \in \mathbb{S}, \quad (12)$$

and using

$$\sum_{j=1}^8 \delta_j \mathbf{A}_j = \sum_{j=1}^8 \delta_j (\mathbf{A}_0 + c_{\alpha,j} \mathbf{M}_1 + c_{\beta,j} \mathbf{M}_2) = \mathbf{A}_0 + \bar{\mathbf{s}}_{\alpha} \mathbf{M}_1 + \bar{\mathbf{s}}_{\beta} \mathbf{M}_2, \quad (13)$$

a bilinear system with

$$\mathbf{x} := [\bar{v}_D \quad \bar{i}_\alpha \quad \bar{i}_\beta]^\top, \quad \mathbf{w} := [\bar{i}_f \quad \bar{v}_\alpha \quad \bar{v}_\beta]^\top, \quad (14)$$

and otherwise identical to the bilinear system with value-discrete inputs in (5) and (6) is obtained.

3.4. Rotating Coordinate Frame Model. The desired operating condition for a three-phase electrical system is to have balanced sets of voltages and currents. An ideal balanced set

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = v_g \begin{bmatrix} \cos(\omega t + \phi) \\ \sin(\omega t + \phi) \end{bmatrix} \quad (15)$$

can be transformed to stationary quantities using a time-varying two-dimensional rotation

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \begin{bmatrix} \cos(\omega t) & \sin(\omega t) \\ -\sin(\omega t) & \cos(\omega t) \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = v_g \begin{bmatrix} \cos \phi \\ \sin \phi \end{bmatrix}. \quad (16)$$

The transformation is applied to the AC side currents and voltages, and the duty ratio command in the averaged bilinear system with matrices from (6). The new state and input vectors are

$$\mathbf{x} := [\bar{v}_D \quad \bar{i}_d \quad \bar{i}_q]^\top, \quad \mathbf{u} := [d_d \quad d_q]^\top, \quad \mathbf{w} := [\bar{i}_f \quad \bar{v}_d \quad \bar{v}_q]^\top, \quad (17)$$

based on the assumption that $\mathbf{d} \approx \mathbf{s}$. The system is identical to (5) and (6), but with a changed matrix

$$\mathbf{A}_0 = \begin{bmatrix} -\frac{1}{R_D C_D} & 0 & 0 \\ 0 & -\frac{R_F}{L_F} & \omega \\ 0 & -\omega & -\frac{R_F}{L_F} \end{bmatrix}. \quad (18)$$

3.5. Linearized Model. Taylor linearization of the bilinear model in rotating coordinates about an equilibrium $\mathbf{x}_e, \mathbf{u}_e, \mathbf{w}_e$ yields a linear description

$$\Delta \dot{\mathbf{x}} = \mathbf{A} \Delta \mathbf{x} + \mathbf{B} \Delta \mathbf{u} + \mathbf{E} \Delta \mathbf{w} \quad (19)$$

with

$$\mathbf{A} = \mathbf{A}_0 + u_{1,e} \mathbf{M}_1 + u_{2,e} \mathbf{M}_2, \quad \mathbf{B} = [\mathbf{M}_1 \mathbf{x}_e \quad \mathbf{M}_2 \mathbf{x}_e]. \quad (20)$$

Such models are used for design of linear controllers and for small-signal stability analysis using the impedance-based stability criterion [3]. Note

that in contrast to Taylor linearization, linear models can also be obtained by assuming that v_D is constant. This is justified if C_D is large and an external source regulates the voltage.

3.6. Discrete-Time Model. A discrete-time model, also referred to as a sampled-data model [4], is obtained from the switched linear model in (7). Over a period T_S , an exact solution for a concrete switching pattern is calculated. The uncontrolled input is assumed to follow linear dynamics [5]

$$\dot{\mathbf{w}} = \mathbf{\Omega}\mathbf{w}, \quad \mathbf{\Omega} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -\omega \\ 0 & \omega & 0 \end{bmatrix}, \quad \mathbf{w}(0) = \begin{bmatrix} i_{f,0} \\ v_g \cos \phi \\ v_g \sin \phi \end{bmatrix} \quad (21)$$

such that an extended state space system

$$\frac{d}{dt} \begin{bmatrix} \mathbf{x} \\ \mathbf{w} \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{A}_{\sigma(t)} & \mathbf{E} \\ \mathbf{0} & \mathbf{\Omega} \end{bmatrix}}_{\tilde{\mathbf{A}}_{\sigma(t)}} \begin{bmatrix} \mathbf{x} \\ \mathbf{w} \end{bmatrix}, \quad \tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{x} \\ \mathbf{w} \end{bmatrix} \quad (22)$$

can be employed for each mode. The extended state after a sequence of modes $\sigma[l], l = 1, \dots, L$ with durations $T[l] \geq 0$ with $\sum_{l=1}^L T[l] = T_S$ is calculated as

$$\tilde{\mathbf{x}}[k+1] = e^{\tilde{\mathbf{A}}_{\sigma[L]}T[L]} e^{\tilde{\mathbf{A}}_{\sigma[L-1]}T[L-1]} \dots e^{\tilde{\mathbf{A}}_{\sigma[1]}T[1]} \tilde{\mathbf{x}}[k], \quad (23)$$

with $\tilde{\mathbf{x}}[k] := \tilde{\mathbf{x}}(kT_S)$. This is a nonlinear discrete-time system which matches the switched system in the sampling instants.

4. Simulation Results. Three models of the VSC system with the parameters in Table 1 are simulated, where linearization was performed about the first operating point in Table 2.

Table 1. System Parameters

L_F	1 mH	R_F	15 m Ω	C_D	800 μ F	R_D	1 k Ω
v_g	326.6 V	T_S	0.1 ms	ω	100 π s ⁻¹	i_f	0 A

Table 2. Operating points

Operating point	OP 1	OP 2	OP 3
Starting time	0 ms	30 ms	50 ms
v_D	700 V	700 V	720 V
i_q	5 A	20 A	20 A

A feedforward control is implemented using a flatness-based approach [6] based on [7] for the bilinear rotating coordinate frame model presented in Section 2.4. Two operating point changes are performed with the parameters given in Tab. 2. The polynomial signals assigned to the flat outputs perform the transition in 4 ms.

Fig. 3 presents the simulation results, with zoomed-in versions of the marked areas to show details.

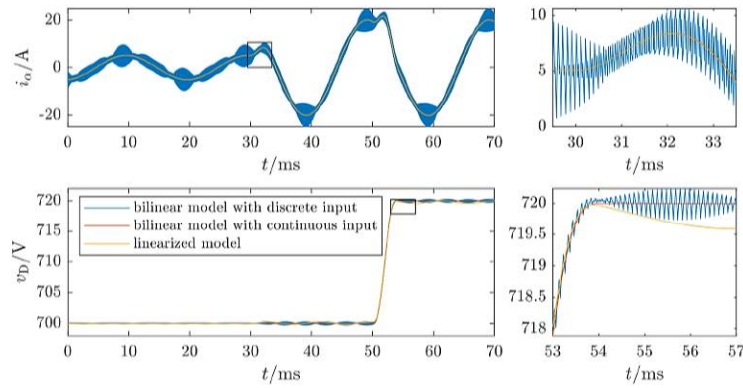


Figure 3. AC side current and DC side voltage in simulation

The bilinear model with discrete input provides information about high-frequency harmonics not available in other models; the switched linear model is equivalent and results in the same state trajectory. A ripple-free approximation of the state trajectory is provided by the averaged model in either the stationary or the rotating coordinate system. The linear model is only valid for small deviations from the operating point about which linearization is performed, large deviations result in errors, clearly visible in the detail of the voltage in Fig. 3.

A didactic example of the time- and value-continuous PWM command as input to averaged models as in (12), the value-discrete switching vector signal as input to the bilinear system as in (4), and the value-discrete mode selection function as input to the switched linear system as in (8) is presented in Fig. 4. The employed space-vector PWM [8] uses configurations 2, 3, 7 and 8 in the first three PWM periods shown, and configurations 3, 4, 7 and 8 in the following three periods.

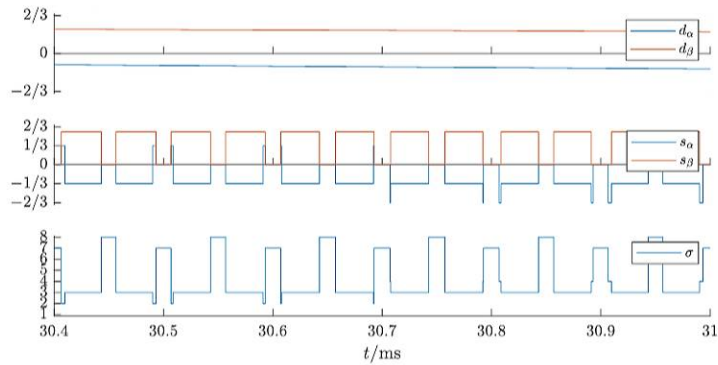


Figure 4. Input signals for different models

References

1. Middlebrook R. D., Čuk S. A general unified approach to modelling switching-converter power stages // *Int. J. Electronics*. 1997. Vol. 42. No 6. P. 521–550.
2. Bacha S., Munteanu I., Bratcu A. I. *Power electronic converters modeling and control*. London: Springer-Verlag, 2014. 477 p.
3. Messo T., Luhtala R., Aapro A., Roinila T. Accurate impedance model of grid-connected inverter for small-signal stability assessment in high-impedance grids // *2018 Int. Power Electronics Conf.* P. 3156–3163.
4. Verghese G. C., Elbuluk M. E., Kassakian J. G. A general approach to sampled-data modeling for power electronic circuits // *IEEE Trans. Power Electron.* 1986. Vol. PE-1. No 2. P. 76–89.
5. Lehn P. W. Exact modeling of the voltage source converter // *IEEE Trans. Power Del.* 2002. Vol. 17. No 1. P. 217–222.
6. Hagenmeyer V., Delaleau E. Exact feedforward linearization based on differential flatness // *Int. J. Control*. Vol. 76. No 6. P. 537–556.
7. Song E., Lynch A. F., Dinavahi V. Experimental validation of a flatness-based control for a voltage source converter // *2007 American Control Conf.* P. 6049–6054.
8. Bowes S. R., Yen-Shin L. The relationship between space-vector modulation and regular-sampled PWM // *IEEE Trans. Ind. Electron.* 1997. Vol. 44. No 5. P. 670–679.

Realtime testbench for inverters – a sliding mode control experiment

1. Introduction. The rising number of inverter interfaced devices in electric grids calls for the development of novel control methods. At the Energy Lab 2.0 of the Karlsruhe Institute of Technology, a realtime experimental testbench for such inverter systems is available. We highlight the capabilities of the testbench with the application of a super-twisting sliding mode controller (STSMC) for a neutral point clamped (NPC) inverter. The advantages of this control method lie in its robustness and the need for only one measurement per phase, since the voltage derivative is obtained via a Levant differentiator. The controller is tested in an experimental setup consisting of an NPC inverter, a passive resistive load and an active load. The experimental results show robust tracking of the reference voltage under step changes of the load current.

2. Description of the Testbench. A generic grid tied inverter, see Figure 1, consists of: a DC source, a Voltage Source Inverter (VSI), a grid filter and the Point of Common Coupling (PCC) as a grid interface.



Figure 1. Schematic overview of a grid tied three phase VSI

In the presented study these components are realized as following:

- **DC source:** A high bandwidth voltage source is used to emulate the DC source, supporting the assumption that the DC source dynamics can be neglected.

Wachter Jan – graduate student, Karlsruhe Institute of Technology; e-mail: jan.wachter@kit.edu, phone: +49(721)6082-57-60

Gröll Lutz – professor, Karlsruhe Institute of Technology; e-mail: lutz.groell@kit.edu, phone: +49(721)6082-57-36

Hagenmeyer Veit – professor, Karlsruhe Institute of Technology; e-mail: veit.hagenmeyer@kit.edu, phone: +49(721)6082-92-00

This work was supported by the Helmholtz Association under the Innovation Pool Project Energy Transition and Circular Economy.

- **VSI:** A three phase NPC inverter topology [1] is considered. The inverter is modelled as an ideal three-level inverter. Further the DC bus capacitors are considered to be sufficiently large, supporting the assumption that the DC bus dynamics can be neglected.
- **Grid filter:** An inductive grid filter in series with a line filter is used, which is modelled as an ideal LC element.

The modelling assumptions lead to the system shown in Figure 2.

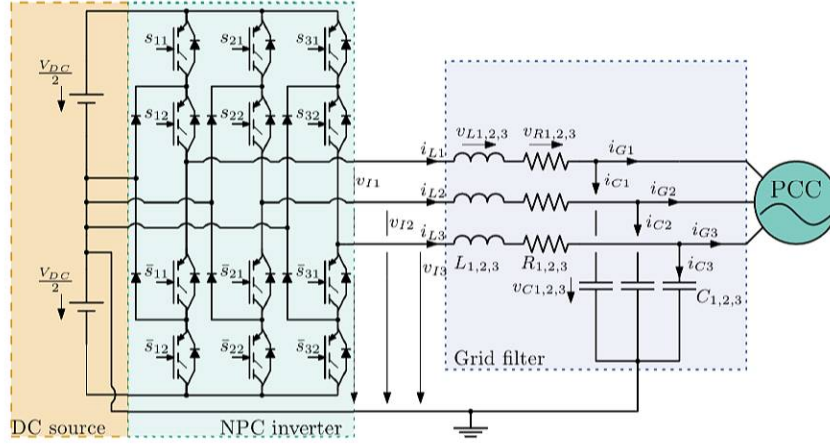


Figure 2. Overview of the grid tied NPC inverter with LC grid filter

2.1. Derivation of the System Model. The index $j = 1, 2, 3$ denotes the phases, v represent voltages and i currents. Passive elements are denoted as L, R, C and represent the respective inductances, resistances and capacities. Considering the setup in Figure 2, the following equations are derived using Kirchoff's laws:

$$L_j \frac{di_{Lj}}{dt} = -R_j i_{Lj} - v_{Cj} - v_{Ij}, \quad (1a)$$

$$C_j \frac{dv_{Cj}}{dt} = i_{Lj} - i_{Gj}. \quad (1b)$$

By eliminating i_{Lj} the following model equations are obtained

$$\frac{d^2 v_{Cj}}{dt^2} = -\frac{R_j}{L_j} \left(\frac{dv_{Cj}}{dt} + \frac{i_{Gj}}{C_j} \right) - \frac{v_{Cj}}{L_j C_j} - \frac{v_{Ij}}{L_j C_j} - \frac{1}{C_j} \frac{di_{Gj}}{dt}. \quad (2)$$

In the following we interpret the physical signals in (2) as averaged over one period of the PWM frequency [2], such that the inverter output voltage can be expressed as $v_{Ij} = \frac{V_{DC}}{2} d_j$, where d_j corresponds to the duty cycle. For the mapping to the averaged gate signals s_{j1} , s_{j2} , \bar{s}_{j1} and \bar{s}_{j2} , we refer to [1].

The averaged signals and parameters are mapped to control nomenclature as shown in Table 1. Further a normalization with suitable SI units is undertaken in order to make all following calculations unit free.

Table 1. Mapping to control nomenclature

Name	Physical Notation and Value	Control Notation
Measured states	v_{Cj} with $j = 1, 2, 3$	$x_1, x_2, x_3,$
Unmeasured states	\dot{v}_{Cj}	x_4, x_5, x_6
Inputs	d_j	u_1, u_2, u_3
Outputs	v_{Cj}	y_1, y_2, y_3
Disturbances	i_{Gj} di_{Gj}/dt	$z_1, z_2, z_3,$ z_4, z_5, z_6
Parameters	$V_{DC} = 720 \text{ V}$	$p_0 = V_{DC}/2$
	$R_j = R = 1 \text{ m}\Omega$	$p_1 = R$
	$L_j = L = 2.4 \text{ mH}$	$p_2 = 1/L$
	$C_j = C = 0.1 \mu\text{F}$	$p_3 = 1/C$

With the applied mapping, (2) can be expressed as

$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ -p_1 p_2 x_4 - p_2 p_3 x_1 \\ -p_1 p_2 x_5 - p_2 p_3 x_2 \\ -p_1 p_2 x_6 - p_2 p_3 x_3 \end{bmatrix}}_{Ax} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ -p_0 p_2 p_3 u_1 \\ -p_0 p_2 p_3 u_2 \\ -p_0 p_2 p_3 u_3 \end{bmatrix}}_{Bu} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ -p_1 p_2 p_3 z_1 - p_3 z_4 \\ -p_1 p_2 p_3 z_2 - p_3 z_5 \\ -p_1 p_2 p_3 z_3 - p_3 z_6 \end{bmatrix}}_{Ez}. \quad (3)$$

Note that the model is linear instead of bilinear, due to the neglect of the DC side dynamics based on the assumptions made in Sec. 1.

2.2. Reference Model and Tracking Error. The reference trajectory of the output y is given by a sinusoidal, symmetrical reference signal y_{ref}

$$y_{\text{ref}}(t) = \begin{bmatrix} A \sin(\omega t) \\ A \sin(\omega t + \frac{2\pi}{3}) \\ A \sin(\omega t - \frac{2\pi}{3}) \end{bmatrix}, \quad (4)$$

and the output tracking error e_y is defined as $e_y = y - y_{\text{ref}}$.

3. Design of a Super-Twisting Sliding Mode Controller for Reference Tracking. For brevity this section omits a detailed introduction to sliding mode control and focuses on the details important to the implementation.

In order to provide a continuous control signal u while retaining the robustness and finite-time convergence properties of sliding mode control, the STSMC is a well suited method. The design consists of two tasks: defining a suitable sliding surface and finding a control law that drives the sliding variable to zero.

3.1. Sliding Surface Design. The sliding surface must be of relative degree one with respect to the control input for the use of the STSMC. We choose the sliding surface independent for each phase as

$$\sigma_j = \dot{e}_{y,j} + a e_{y,j} = (x_{j+3} - \dot{y}_{\text{ref},j}) + a(x_j - y_{\text{ref},j}) \quad (5)$$

with $a \geq 0$ [3, 4]. Since x_{j+3} is not measurable, it is replaced by its estimate obtained via the Levant differentiator \hat{x}_{j+3} in the implementation. Next, a control law u_j is designed which drives the trajectory on the sliding surface in finite-time, which can be expressed in the reaching condition $\sigma_j \dot{\sigma}_j \leq -\kappa_j |\sigma_j|$ with $\kappa_j > 0$ [4, 5].

3.2. Super-Twisting Algorithm. The super-twisting controller [4] which satisfies the reaching condition is given as

$$\dot{w}_j = \begin{cases} -w_j - c_1 \sqrt{|\sigma_j|} \text{sign } \sigma_j, & |w_j + c_1 \sqrt{|\sigma_j|} \text{sign } \sigma_j| > u_{\max}, \\ c_2 \text{sign } \sigma_j, & |w_j + c_1 \sqrt{|\sigma_j|} \text{sign } \sigma_j| \leq u_{\max}, \end{cases} \quad (6a)$$

$$u_j = -w_j - c_1 \sqrt{|\sigma_j|} \text{sign } \sigma_j, \quad (6b)$$

where the parameters $c_1, c_2 > 0$ are designed to guarantee finite-time convergence [5, 6]. Considering

$$\dot{\sigma}_j = \underbrace{-\ddot{y}_{\text{ref},j} - a \dot{y}_{\text{ref},j} + (a x_{j+3} + (Ax)_{j+3} + (Ez)_{j+3})}_{\phi_j(x,z)} - p_0 p_2 p_3 u_j \quad (7)$$

and assuming $\sup_t |\dot{\phi}_j(x(t), z(t))| \leq h$ the constants can be chosen as $c_1 = 1.5 \sqrt{h}$ and $c_2 = 1.1 h$ [4]. Since $\dot{\phi}_j(x, z)$ contains the unknown disturbance z , the boundedness depends on the application. In order to further study the existence domain of the sliding mode, the concept of equivalent control [7] is used, which is defined as the necessary control signal to keep the trajectory on the sliding surface. During ideal sliding

$\dot{\sigma}_j = 0$ is fulfilled, such that inserting $u_j = u_{eq,j}$ in (7) and solving for $u_{eq,j}$ yields

$$u_{eq,j} = \frac{\phi_j(x, z)}{p_0 p_2 p_3}. \quad (8)$$

Since the control u_j cannot realize arbitrarily large values, the existence domain is limited by $|u_j| \leq u_{\max}$. Considering (8) with respect to the experiments, p_0 must be chosen to comply with the largest expected perturbation [3].

3.3. Levant Differentiator for the Output Voltage. The voltages measurements x_j required in (6) can be measured directly, but not their derivatives x_{j+3} . To avoid the additional measurement of the capacitor currents as in [3], we obtain the estimates \hat{x}_{j+3} via the robust differentiator proposed by Levant [8]. It offers several advantages over a linear differentiator, since the small sampling times of the implementation are uncritical and the sensitivity to high frequency disturbances is improved which is crucial due to the switching nature of the inverter. The derivation is analogous to the STSMC as described in [4]. The differentiator system can be expressed as:

$$\dot{\xi}_{1,j} = -\lambda_1 \sqrt{|\xi_{1,j} - x_j|} \operatorname{sign}(\xi_{1,j} - x_j) + \xi_{2,j}, \quad (9a)$$

$$\dot{\xi}_{2,j} = -\lambda_2 \operatorname{sign}(\xi_{1,j} - x_j), \quad (9b)$$

$$\hat{x}_{j+3} = \xi_{2,j}, \quad (9c)$$

where $\xi_{1,j}$, $\xi_{2,j}$ are the differentiator states. Further $\lambda_1 = 1.5 \sqrt{\gamma}$, $\lambda_2 = 1.1 \gamma$, with $\sup_t |\ddot{x}_j(t)| \leq \gamma$ [4, 8]. Since \ddot{x}_j is unknown, $\ddot{y}_{\text{ref},j}$ is used as a rough estimate, assuming the control error remains small.

4. Experimental Setup and Results. This section briefly presents the experimental setup and the results of the conducted experiment. Thereby we establish the suitability of the testbench for experiments with advanced control methods for inverter systems such as the implemented STSMC.

4.1. Experimental Setup. The experimental setup is shown in Figure 3. The control sampling frequency of the programmable controller is $f_{cc} = 100$ kHz and the frequency of the PWM carrier is $f_{\text{PWM}} = 40$ kHz. The passive load is purely resistive with 40Ω per phase. The electric load draws a sinusoidal and symmetrical current, which is set to exhibit a 8 A amplitude change from the base amplitude of 11 A. The parameters of

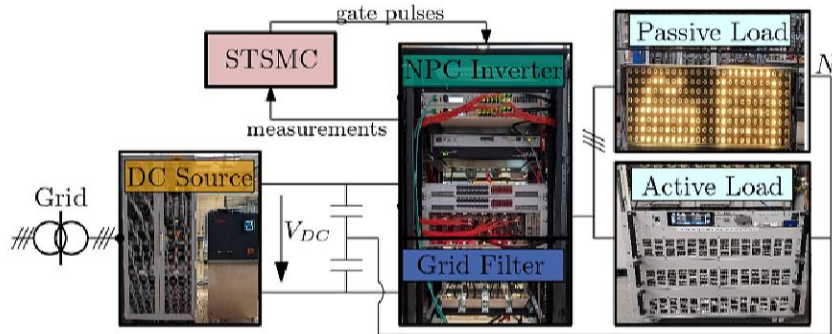


Figure 3. Schematic overview of the experimental setup

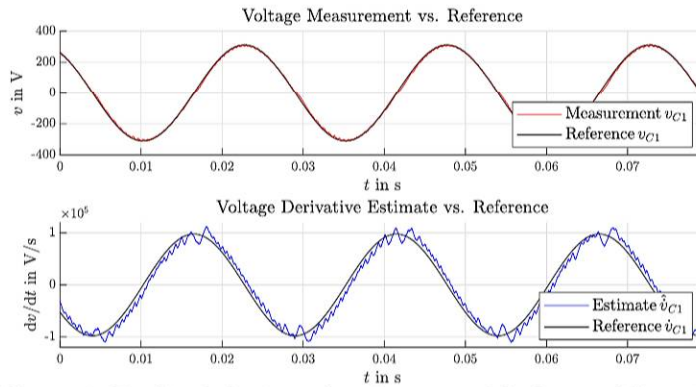


Figure 4. Tracking behavior and performance of the Levant differentiator

the control and differentiator system are chosen as $a = 3.1 \cdot 10^{-3}$, $c_1 = 0.03$, $c_2 = 320$, $\gamma = 5.5 \cdot 10^7$.

4.2. Experimental Results. The upper plot of Figure 4 shows the tracking of the reference voltage with a root mean square error in percent of the peak reference voltage $\text{RMSE}_{\%PV} = 2.4$. The lower plot shows the performance of the described Levant differentiator. Note that a phase lag is inherent to all causal differentiators, here the observed phase lag is around 5.6° . The visible peaks of \dot{v}_{C1} can also be observed in the voltage measurement v_{C1} , but since they are attenuated by an amplitude normalized factor of 5 they are not visible due to the scaling. Large derivatives of the load current are critical, as shown in Sec. 2.2. Figure 5 presents the results during such a rapid change of the load current. The upper plot shows that the voltage deviation caused by the load step, seen in the lower plot, is recovered within 1 ms.

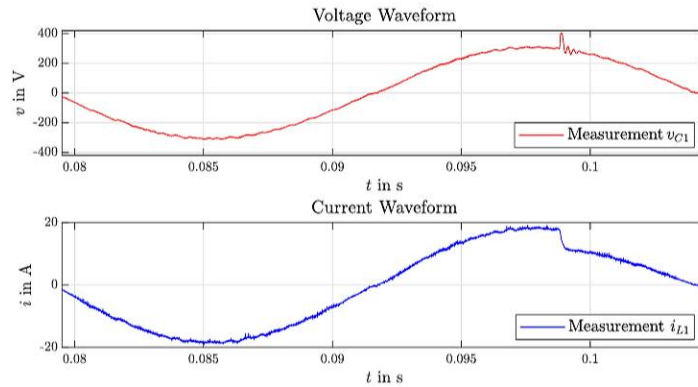
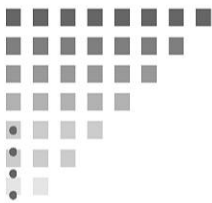


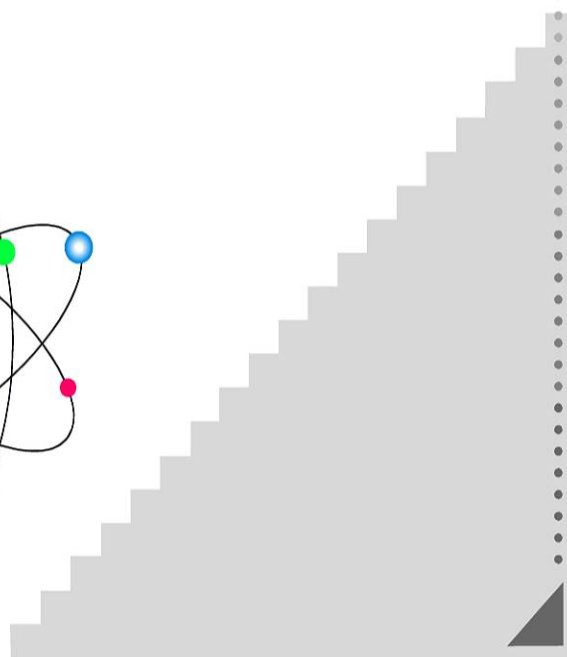
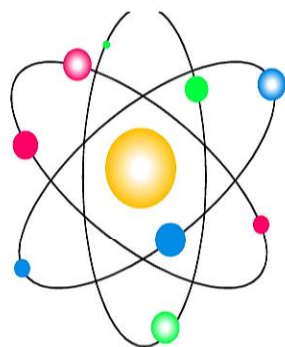
Figure 5. Tracking behavior during a sudden load change.

References

1. Rodriguez J. et al. A survey on neutral-point-clamped inverters // *IEEE Trans. on Ind. Electronics*. 2010. Vol. 57. No 7. P. 2219–2230.
2. Yazdani A., Iravani R. A generalized state-space averaged model of the three-level NPC converter for systematic DC-voltage-balancer and current-controller design // *IEEE Trans. on Power Delivery*. 2005. Vol. 20. No 2. P. 1105–1114.
3. Carpita M., Marchesoni M. Experimental study of a power conditioning system using sliding mode control // *IEEE Trans. on Power Electronics*. 1996. Vol. 11. No 5. P. 731–742.
4. Shtessel Y. et al. *Sliding mode control and observation*. New York: Springer, 2014. 373 p.
5. Brandstädter H. *Sliding mode control of electromechanical systems*. Dissertation. Technische Universität München. 2008.
6. Boiko I. et al. Analysis of steady state behavior of second order sliding mode algorithms // *Proceedings of the American Control Conference*. 2004. Vol. 1. P. 632–637.
7. Sira-Ramirez H. *Sliding mode control*. Springer International Publishing Switzerland, 2015. 272 p.
8. Levant A. Robust exact differentiation via sliding mode technique // *Automatica*. 1998. Vol. 34. No 3. P. 379–384.



*2. Математические
методы в механике и
физике*



УДК 519.67

Алцыбеев Г. О., Бурова И. Г.

Газотурбинный двигатель и сплайновые приближения

1. Введение. Интегральные уравнения Вольтерра второго рода применяются при решении многих задач, в том числе для решения задач динамической геодезии, описания вязкого трения и теплопроводности.

В работах [1–3] рассматривается интегральная математическая модель, описывающая динамику газотурбинного двигателя. Данная математическая модель сводится к интегральному уравнению Вольтерра второго рода. В представленной работе рассматривается нахождение приближенного решения интегрального уравнения Вольтерра различными методами. Традиционно применяются методы, основанные на методах квадратур и методе Галеркина. В работе сравниваются решения интегрального уравнения Вольтерра, полученные при использовании квадратурных интерполяционных формул правых прямоугольников и трапеций, с решениями, полученными при использовании полиномиальных и неполиномиальных сплайнов второго порядка аппроксимации.

2. Постановка задачи. Математическая модель, описывающая динамику газотурбинного двигателя сводится к интегральному уравнению Вольтерра второго рода вида

$$u(t) = f(t) + \int_a^t K(t, s) u(s) ds, \quad t \in [a, b],$$

где $u(t)$ – неизвестная функция, $K(t, s)$ – заданная непрерывная по совокупности аргументов функция, называемая ядром интегрального уравнения, $f(t)$ – заданная функция.

Задача заключается в нахождении значений функции $u(t)$.

Алцыбеев Глеб Олегович – студент, Санкт-Петербургский государственный университет; e-mail: gleb.alcybееv@gmail.com, тел.: +7(931)531-43-75

Бурова Ирина Герасимовна – профессор, Санкт-Петербургский государственный университет; e-mail: burovaig@mail.ru, тел.: +7(921)986-44-15

3. Методы решения задачи. Рассмотрим пример уравнения, в котором ядро $K(t, s) = s + ts^2$, а $u(t)$ – альтернативно заданная функция, график которой представлен на рис. 1–5 (слева). Функция построена на основе графика точного решения уравнения из примера в работе [2].

3.1. Метод квадратур и метод простой итерации. Методы, основанные на квадратурных интерполяционных формулах, хорошо известны [4, 5]. Приближенные значения $\tilde{u}(x)$ функции $u(x)$ в равноотстоящих узлах сетки $a \leq x_1 < x_2 < \dots < x_n \leq b$ находятся по рекуррентной формуле

$$\tilde{u}(x_i) = \sum_{j=1}^i A_j K(x_i, x_j) \tilde{u}(x_j) + f(x_i), \quad i = 1, \dots, n,$$

где A_j – коэффициенты квадратурной интерполяционной формулы.

На рис. 1 представлены результаты вычислений методом правых прямоугольников с количеством узлов сетки $n = 32$, точное и приближенное решение представлены слева, а погрешность – справа. На рис. 2 представлены результаты вычислений методом трапеций с количеством узлов сетки $n = 32$, точное и приближенное решение представлены слева, а погрешность – справа.

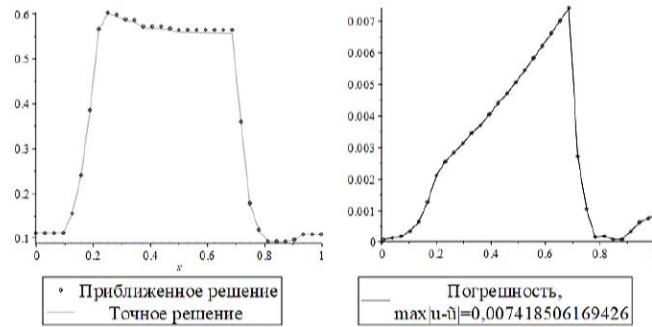


Рис. 1. Точное и приближенное решение, полученное методом правых прямоугольников (слева) и его погрешность (справа)

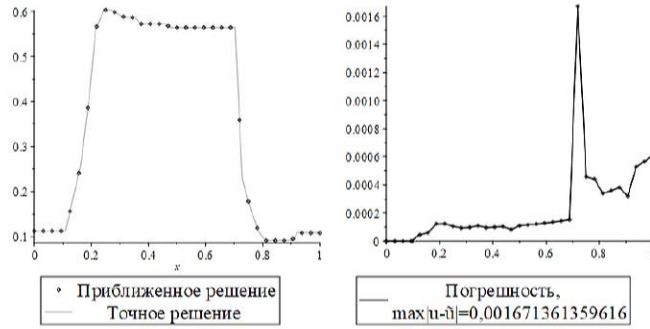


Рис. 2. Точное и приближенное решение, полученное методом трапеций (слева) и его погрешность (справа)

В программной реализации метода простой итерации интеграл вычислялся с помощью квадратурной формулы трапеций. Приближенные значения функции $u(x)$ находятся по рекуррентной формуле

$$\tilde{u}_{k+1}(x_i) = f(x_i) + \int_{x_0}^{x_i} K(x_i, s) \tilde{u}_k(s) ds.$$

Полученные результаты представлены на рис. 3.

3.2. Применение сплайнов. В работах [6,7] описывается применение полиномиальных и неполиномиальных сплайнов для решения уравнения Вольтерра второго рода. Приближенные значения $\tilde{u}(x)$ функции $u(x)$ в узлах сетки $a \leq x_1 < x_2 < \dots < x_n \leq b$ вычисляем последовательно, решая уравнения

$$\tilde{u}(x_k) + \sum_{j=1}^{k-1} W_j(x_k) = f(x_k), \quad k = 1, \dots, n,$$

где

$$W_j(x_k) = \tilde{u}(x_j) \int_{x_j}^{x_{j+1}} K(x_k, s) \omega_j(s) ds + \\ + \tilde{u}(x_{j+1}) \int_{x_j}^{x_{j+1}} K(x_k, s) \omega_{j+1}(s) ds,$$

где $\omega_{j-1}(s)$ и $\omega_j(s)$ – базисные полиномиальные сплайны

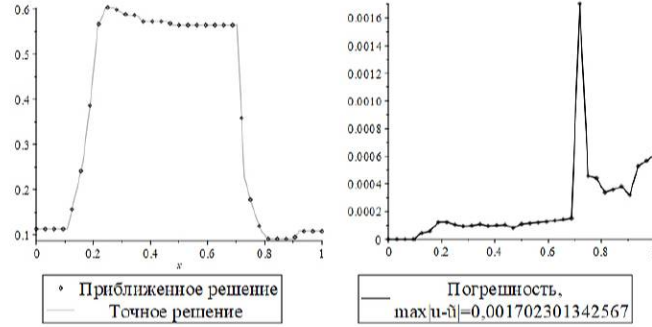


Рис. 3. Точное и приближенное решение, полученное методом простой итерации (слева) и его погрешность (справа)

$$\omega_j(s) = \frac{s - x_{j+1}}{x_j - x_{j+1}}, \quad \omega_{j+1}(s) = \frac{s - x_j}{x_{j+1} - x_j}, \quad s \in [x_j, x_{j+1}]$$

или неполиномиальные. Например, как показано в работе [6], можно взять функцию $\varphi(s) = \exp(s)$ или $\varphi(s) = \exp(-s)$. В этом случае базисные сплайны имеют вид

$$\omega_j(s) = \frac{\varphi(s) - \varphi(x_{j+1})}{\varphi(x_j) - \varphi(x_{j+1})}, \quad \omega_{j+1}(s) = \frac{\varphi(s) - \varphi(x_j)}{\varphi(x_{j+1}) - \varphi(x_j)},$$

$$s \in [x_j, x_{j+1}].$$

Пусть $h = x_{j+1} - x_j$. Можно показать, что в случае полиномиальных сплайнов справедливо неравенство

$$|\tilde{u}(x) - u(x)| \leq 0,25h^2 \|u''\|_{[x_j, x_{j+1}]}, \quad x \in [x_j, x_{j+1}],$$

где $u(x)$ – точное решение, $\tilde{u}(x)$ – приближенное решение,

$$\tilde{u}(x) = u(x_j)\omega_j(x) + u(x_{j+1})\omega_{j+1}(x), \quad x \in [x_j, x_{j+1}].$$

Отметим, что аппроксимация, построенная на равномерной сетке с помощью неполиномиальных сплайнов, имеет погрешность аппроксимации порядка $O(h^2)$.

Результаты вычислений приближенного значения функции $u(x)$ с помощью полиномиальных сплайнов представлены на рис. 4, а неполиномиальных – на рис. 5. Количество узлов сетки $n = 32$.

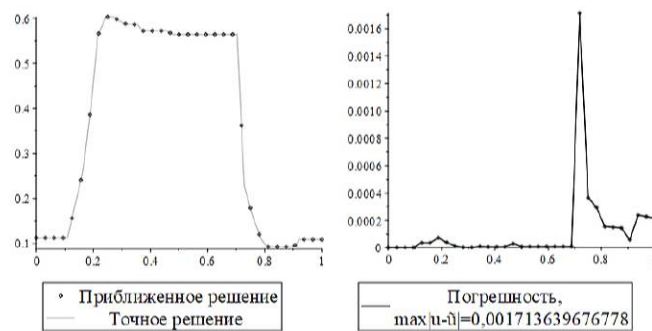


Рис. 4. Точное и приближенное решение, полученное с использованием полиномиальных сплайнов (слева) и его погрешность (справа)

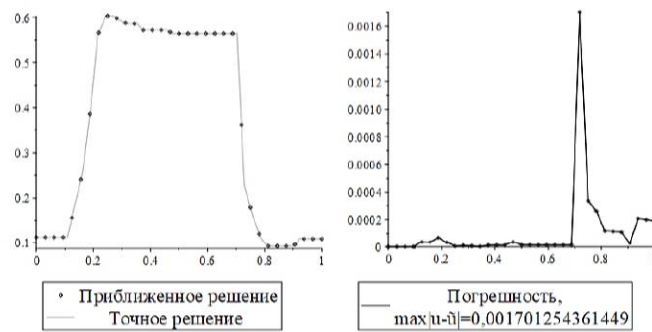


Рис. 5. Точное и приближенное решение, полученное с использованием неполиномиальных сплайнов (слева) и его погрешность (справа)

Максимальное по абсолютной величине значение погрешности при приближении сплайнами оказалось больше, чем при решении

методом трапеций, но в остальных узлах сетки значения получились точнее. Также при использовании сплайнов можно использовать неравномерную сетку. При этом можно построить адаптивную сетку так, что помимо уменьшения погрешности, количество используемых узлов будет также меньше. Погрешности вычислений на неравномерной сетке (20 узлов) с помощью полиномиальных и неполономиальных сплайнов представлены на рис. 6. В качестве примера взяты следующие узлы сетки: $x_1 = 0$, $x_2 = 0,10894$, $x_3 = 0,16341$, $x_4 = 0,19322$, $x_5 = 0,21518$, $x_6 = 0,24390$, $x_7 = 0,26965$, $x_8 = 0,29457$, $x_9 = 0,31653$, $x_{10} = 0,35285$, $x_{11} = 0,37100$, $x_{12} = 0,45880$, $x_{13} = 0,47995$, $x_{14} = 0,70352$, $x_{15} = 0,72846$, $x_{16} = 0,76098$, $x_{17} = 0,78482$, $x_{18} = 0,80678$, $x_{19} = 0,90054$, $x_{20} = 0,91978$, $x_{21} = 1$.

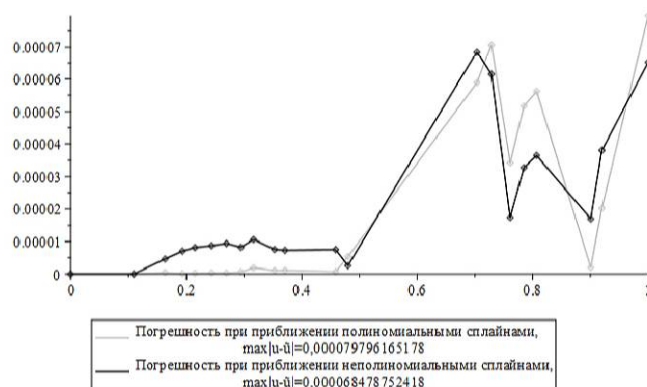


Рис. 6. Погрешность при приближении $u(x)$ полиномиальными и неполономиальными сплайнами на неравномерной сетке

4. Вывод. Рассмотрены методы для определения процесса изменения состояния газотурбинного двигателя. Для нахождения приближенных значений функции $u(x)$ применены различные методы. Для рассмотренных методов написаны программы в среде Maple. Численные эксперименты показывают, что сплайновые аппроксимации дают значительно меньшую погрешность, если применить неравномерную сетку.

Литература

1. Епифанов С. В., Кузнецов В. И., Богаенко И. И. Синтез систем управления и диагностирования газотурбинных двигателей. Киев: Техника, 1998. 312 с.
2. Myrhorod V., Hvozdeva I. On one solution of Volterra integral equation of second kind // AIP Conference Proceedings. 2016. Vol. 1773. Art. No 040006.
3. Миргород В. Ф., Грудинкин В. М. Модальная и интегральная форма математических моделей газотурбинных двигателей // Вісник двигунобудування. 2008. № 3. С. 185–189.
4. Мысовских И. П. Лекции по методам вычислений. М.: Физматлит, 1962. 344 с.
5. Карчевский Е. М. Численные методы решения интегральных уравнений и комплекс программ на языке Matlab: Учебное пособие. Казань: Казанский университет, 2019. 60 с.
6. Burova I. G. On left integro-differential splines and Cauchy problem // International Journal of Mathematical Models and Methods in Applied Sciences. 2015. Vol. 9. P. 683–690.
7. Burova I. G., Alcybeev G. O. Application of splines of the second order approximation to Volterra integral equations of the second kind. Applications in systems theory and dynamical systems // International Journal of Circuits, Systems and Signal Processing. 2021. Vol. 15. P. 63–71.

УДК 517.958, 539.3

Вакаева А. Б., Шувалов Г. М., Костырко С. А.

Морфологическая неустойчивость нанопоры в твердом теле при поверхностной диффузии

1. Введение. За последние годы было выявлено, что упругость может влиять на морфологическую устойчивость пор, отверстий и каналов. При этом дефекты изменяют свою форму под действием внутренних или внешних напряжений [1]. Морфологические изменения могут также влиять на поведение сыпучих материалов, инициирующих зарождение трещин на поверхностях пустот [2]. Развитие процессов пластической деформации и разрушения во многом определяет механические свойства материала в целом и поэтому представляет большой интерес.

В работах [3, 4] методом возмущений границы решена задача об упругом теле с почти круговым нанодефектом. Полученные результаты позволили оценить влияние поверхностных напряжений на напряженно-деформированное состояние вблизи дефекта и, в частности, на концентрацию напряжений. Особенности поведения поверхностного слоя в напряженном материале заключаются в том, что изначально гладкая поверхность становится шероховатой под действием ряда природных явлений: тепла, света, химических веществ, механических напряжений и др. [5].

В рамках настоящего исследования планируется разработать новую континуальную модель зарождения и роста дефектов на поверхности наноматериала, учитывающую термодинамические процессы, локализованные в поверхностных слоях. В предложенной модели зарождение дефекта на свободной поверхности связано, прежде всего, с ее морфологическими изменениями на атомарном уровне в результате механических воздействий, например, при изготовлении, обработке и эксплуатации материала. Доминирующей движущей си-

Вакаева Александра Борисовна – старший преподаватель, Санкт-Петербургский государственный университет; e-mail: a.vakaeva@spbu.ru, тел.: +7(921)796-50-60

Шувалов Глеб Михайлович – аспирант, Санкт-Петербургский государственный университет; e-mail: g.shuvalov@spbu.ru, тел.: +7(921)387-58-63

Костырко Сергей Алексеевич – доцент, Санкт-Петербургский государственный университет; e-mail: s.kostyrko@spbu.ru, тел.: +7(921)406-47-62

Работа выполнена при финансовой поддержке РФФ, грант № 19-71-00062

лой таких превращений является изменение химического потенциала, которое вследствие снижения устойчивости поверхностных атомных слоев приводит к диффузии атомов вдоль поверхности.

Данное исследование является продолжением работы [6], в которой методом возмущения границы было найдено решение задачи для упругого тела с почти круговым nanoотверстием. Цель работы – исследовать влияние упругих свойств поверхности на устойчивость формы отверстия.

2. Постановка задачи. Рассмотрим бесконечное упругое тело с цилиндрической порой. Считаем, что выполнены условия плоской деформации, и в плоскости, перпендикулярной оси цилиндра, граница отверстия Γ мало отличается от окружности радиуса a и определяется равенством

$$z \equiv \zeta = \rho e^{i\theta} = [a + A(\tau) \cos(2\theta)] e^{i\theta}, \quad (1)$$

$$A(0) = A_0, \quad 0 < A(\tau) \ll a \quad \forall \tau,$$

здесь ρ – полярный радиус, θ – полярный угол и A – амплитуда отклонения границы поры от окружности в момент времени τ .

Согласно обобщенному закону Лапласа–Юнга [6], граничное условие имеет вид

$$\sigma_n(\zeta) = \sigma_{nn} + i\sigma_{nt} = \kappa\sigma_{tt}^s - i\frac{1}{h}\frac{d\sigma_{tt}^s}{d\theta} \equiv t^s(\zeta), \quad \zeta \in \Gamma, \quad (2)$$

где σ_{nn}, σ_{nt} – нормальные и касательные напряжения в локальной декартовой прямоугольной системе координат n, t (в уравнении (2) ось n перпендикулярна Γ); σ_{tt}^s – поверхностное напряжение; h – метрический коэффициент; κ – кривизна поверхности.

На бесконечности заданы напряжения $\sigma_{11}^\infty = \sigma_{22}^\infty = \sigma_0$, $\sigma_{12}^\infty = 0$ в системе координат x_1, x_2 и угол поворота $\omega^\infty = 0$. В случае плоской деформации определяющие соотношения поверхностной и объемной теории упругости, согласно [6, 7], можно записать в виде

$$\sigma_{tt}^s = \gamma_0 + M_s \varepsilon_{tt}^s, \quad M_s = \lambda_s + 2\mu_s, \quad (3)$$

$$\sigma_{nn} = (\lambda + 2\mu)\varepsilon_{nn} + \lambda\varepsilon_{tt}, \quad \sigma_{tt} = (\lambda + 2\mu)\varepsilon_{tt} + \lambda\varepsilon_{nn}. \quad (4)$$

В равенствах (3), (4) величина γ_0 – остаточное поверхностное напряжение, которое действует при отсутствии деформаций; ε_{ij} – компоненты объемной деформации в классической теории упругости; ε_{tt}^s – окружная поверхностная деформация; λ_s, μ_s – модули поверхностной упругости, аналогичные постоянным Ламе λ и μ .

Предполагается, что выполнено условие идеального сцепления поверхности с основным материалом

$$\varepsilon_{tt}^s(\zeta) = \varepsilon_{tt}(\zeta), \quad \zeta \in \Gamma. \quad (5)$$

В силу отличия границы поры от круговой формы, при всестороннем растяжении будет наблюдаться концентрация напряжений. Для релаксации напряжений форма границы может меняться под действием поверхностной диффузии, при этом поток атомов вдоль поверхности Γ пропорционален градиенту химического потенциала

$$J(\zeta, \tau) = -\frac{D_s C_s}{k_b T} \frac{\partial \chi(\zeta, \tau)}{\partial s}, \quad (6)$$

где D_s – коэффициент поверхностной самодиффузии, C_s – концентрация поверхностных дефектов, Ω – атомный объем, T – абсолютная температура и k_b – постоянная Больцмана.

Локальный химический потенциал χ определяется как свободная энергия на единицу добавляемого объема

$$\chi(\zeta, \tau) = \chi_0 - \kappa(\zeta, \tau) U_s \Omega + U(\zeta, \tau) \Omega, \quad (7)$$

где χ_0 – химический потенциал в случае кругового отверстия, U и U_s – удельные объемная энергия деформации и поверхностная энергия соответственно.

Рассматривая поток атомов через элементарный объем вблизи границы поры, из закона сохранения масс с учетом (6), (7) получим выражение для скорости изменения формы поры

$$\frac{\partial g(\theta, \tau)}{\partial \tau} = K_s h(x_1, \tau) \frac{\partial^2}{\partial s^2} [U(\zeta, \tau) - \kappa(x_1, \tau) U_s(\zeta, \tau)], \quad (8)$$

$$g(\theta, \tau) = A(\tau) \cos(b\theta), \quad K_s = \frac{D_s C_s \Omega}{k_b T}.$$

Как и в работе [8], морфологическая устойчивость поверхности поры в напряженном твердом теле рассматривается в квазистатической постановке. Следовательно, чтобы определить упругую энергию деформации U и поверхностную энергии U_s , необходимо решить статическую задачу о деформации плоскости с отверстием (1)–(5). Решение этой задачи с использованием метода возмущений, формул Мусхелишвили и соотношений Гертвина–Мердока получено в

[6]. Подставляя выражения для упругой U и межфазной U_s энергии в уравнение (8) и интегрируя его, получим зависимость изменения формы поры со временем как функцию физических и геометрических параметров задачи.

3. Численные результаты. На рис. 1 представлен график зависимости относительной амплитуды возмущения границы кругового отверстия от радиуса a при различных параметрах $M_s = \{6; 60\}$ Н/м (пунктирные и сплошные линии соответственно), напряжениях на бесконечности $\sigma_0 = \{1; 1,5\}$ ГПа (кривые 1, 2 соответственно) и остаточном поверхностном напряжении $\gamma_0 = \{0,5; 1\}$ Н/м. Объемные параметры Ламе соответствуют алюминию и приняты $\lambda = 58,17$ ГПа и $\mu = 26,13$ ГПа. Критический радиус отверстия a_{cr} определяется из пересечения графиков с осью абсцисс. Если радиус отверстия в начальный момент времени меньше критического, форма отверстия с течением времени стремится к окружности.

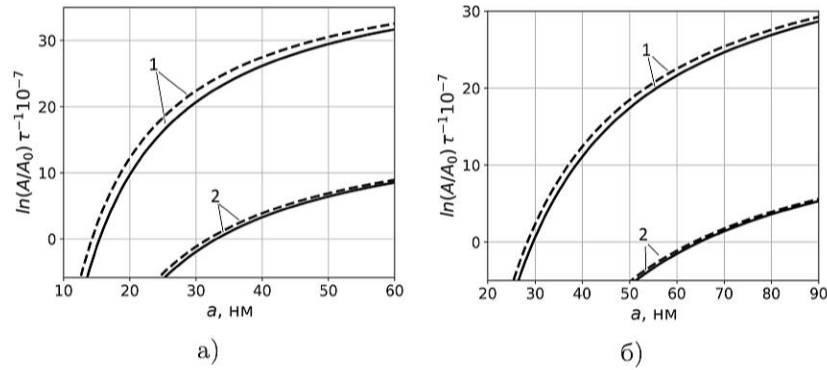


Рис. 1. Зависимость амплитуды отклонения A от радиуса отверстия a
а) при $\gamma_0 = 0,5$ Н/м и б) при $\gamma_0 = 1,0$ Н/м

Из рис. 1 видно, что критическое значение увеличивается с ростом поверхностной жесткости M_s и остаточного поверхностного напряжения γ_0 и уменьшается с увеличением напряжений на бесконечности σ_0 .

Таким образом, решена задача о морфологической устойчивости нанотверстия в твердом теле при поверхностной диффузии для случая, когда граница отверстия задана по косинусоидальному закону.

Проанализирована зависимость амплитуды отклонения границы от радиуса базового отверстия для различных значений остаточного поверхностного напряжения.

Литература

1. Suo Z., Wang W. Diffusive void bifurcation in stressed solid // Journal of Applied Physics. 1994. Vol. 76. No 6. P. 3410–3421.
2. Newcomb S. A., Tressler R. E. Slow crack growth in sapphire fibers at 800 to 1500 C // Journal of the American Ceramic Society. 1993. Vol. 76. No 10. P. 2505–2512.
3. Vakaeva A. B., Grekov M. A. Effect of interfacial stresses in an elastic body with a nanoinclusion // AIP Conference Proceedings. 2018. Vol. 1959. Art. No 070036.
4. Vakaeva A. B., Shuvalov G. M., Kostyrko S. A., Grekov M. A. The finite element modeling of an elastic body with a nanometer surface relief // AIP Conference Proceedings. 2019. Vol. 2116. Art. No 290008.
5. Gharahi A., Schiavone P. Effective elastic properties of plane micro polar nanocomposites with flexural effects // International Journal of Mechanical Sciences. 2018. Vol. 149. P. 84–92.
6. Вакаева А. Б. Эффект поверхностных напряжений и формы нанометрового рельефа поверхности отверстия в упругом теле // Процессы управления и устойчивость. 2016. Т. 3. № 1. С. 154–158.
7. Gurtin M. E., Murdoch A. I. A continuum theory of elastic material surfaces // Arch. Ration. Mech. and Anal. 1975. Vol. 57. No 4. P. 291–323.
8. Костырко С. А., Шувалов Г. М. Влияние упругих свойств поверхности твердого тела на процесс ее реорганизации под действием напряжений // Процессы управления и устойчивость. 2018. Т. 5. № 1. С. 224–228.

Расчет распределения потенциала в двумерной диодной системе с лезвийным полевым катодом

1. Постановка задачи. Полевые катоды, обладающие неоспоримыми преимуществами по сравнению с термокатадами (высокое быстродействие, безынерционность и т.п.), находят широкое применение в разнообразных современных вакуумных электронных устройствах [1].

В работе представлен расчет распределения потенциала в диодной эмиссионной системе с лезвийным полевым катодом на плоской подложке. Анод – плоскость, параллельная подложке. Для вычисления электростатического потенциала во всей области системы используется метод разделения переменных в декартовых координатах (x, y, z) . В силу того, что форма лезвийного катода при $z = const$ не меняется, распределение потенциала $U((x, y)$ зависит от двух координат (x, y) . Влияние полевого катода на распределение поля заменим влиянием N заряженных нитей с плотностями зарядов $\tau_i(y)$, $i = 1, \dots, N$, расположенных в плоскости xOy с координатами $x = x_0$, $y = y_i$, $i = 1, \dots, N$. Положим, что поверхность реального катода совпадает с нулевой эквипотенциалью [2].

2. Распределение электростатического потенциала. Параметры системы: L – высота катода; $y = 0$, $0 \leq x \leq x_1$ – поверхность подложки катода; $y = y_{N+1}$, $0 \leq x \leq x_1$ – поверхность анода; $x = 0$, $x = x_1$, $0 \leq y \leq y_{N+1}$ – границы области по переменной x ; $x_0 = x_1/2$, $y = y_i$, $i = 1, \dots, N$ – координаты заряженных нитей с плотностью зарядов $\tau_i(y)$.

Виноградова Екатерина Михайловна – профессор, Санкт-Петербургский государственный университет; e-mail: vincat2008@yandex.ru, тел.: +7(812)428-42-35

Доронин Григорий Геннадьевич – аспирант, Санкт-Петербургский государственный университет; e-mail: fluctum@gmail.com, тел.: +7(812)428-42-35

Работа выполнена при финансовой поддержке РФФИ, грант № 20-07-01086

Граничные условия: $U(x, 0) = 0$, $0 \leq x \leq x_1$, - на подложке эмиттера; $U(x, y_{N+1}) = U_0$, $0 \leq x \leq x_1$, - на аноде; $U(0, y) = U_0 \frac{y}{y_{N+1}}$,

$U(x_1, y) = U_0 \frac{y}{y_{N+1}}$, $0 \leq y \leq y_{N+1}$ - на границе системы по переменной y .

Пользуясь методикой, представленной в работах [3, 4], распределение потенциала $U(x, y)$ можно записать в виде:

при $0 \leq x < x_0$

$$U(x, y) = U_0 \frac{y}{y_{N+1}} + \frac{2}{\pi \varepsilon_0} \sum_{i=1}^N \tau_i \sum_{m=1}^{\infty} \frac{\sin[\alpha_m y_i] \sinh[\alpha_m (x_1 - x_0)]}{m \sinh[\alpha_m x_1]} \times \quad (1)$$

$$\times \sinh[\alpha_m x] \sin[\alpha_m y],$$

при $x_0 < x \leq x_1$

$$U(x, y) = U_0 \frac{y}{y_{N+1}} + \frac{2}{\pi \varepsilon_0} \sum_{i=1}^N \tau_i \sum_{m=1}^{\infty} \frac{\sin[\alpha_m y_i] \sinh[\alpha_m x_0]}{m \sinh[\alpha_m x_1]} \times \quad (2)$$

$$\times \sinh[\alpha_m (x_1 - x)] \sin[\alpha_m y],$$

$$\alpha_m = \pi m / y_{N+1}. \quad (3)$$

3. Результаты численных расчетов. В соответствии с формулами (1)–(3), для того чтобы нулевая эквипотенциаль проходила через вершину катода (x_0, L) , где L - высота катода, требуется решить уравнение

$$U(x_0, L) = 0. \quad (4)$$

Для вычисления распределения потенциала рассмотрены следующие значения параметров в безразмерных величинах: $y_{N+1} = 10$, $y_i = y_{N+1} \frac{9i}{10N}$, $\tau_i = \tau = -1,45$, $i = 1, \dots, N$, $N = 50$, $L = y_N + \delta$, $U_0 = 100$, $\varepsilon_0 = 1$ при варьировании значений x_1 и x_0 . Значение величины τ вычислялось по формуле (4).

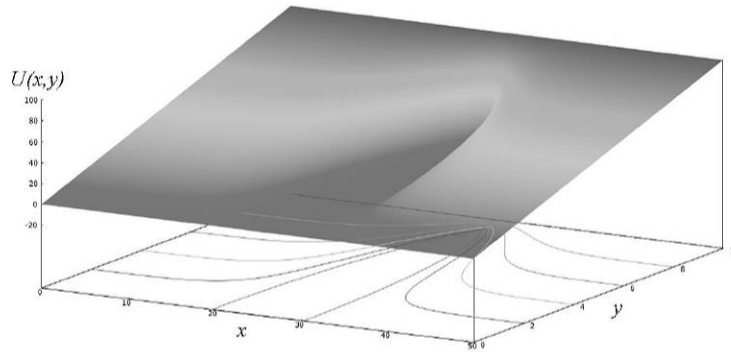


Рис. 1. Распределение потенциала при $x_1 = 20, \delta = 10^{-3}$

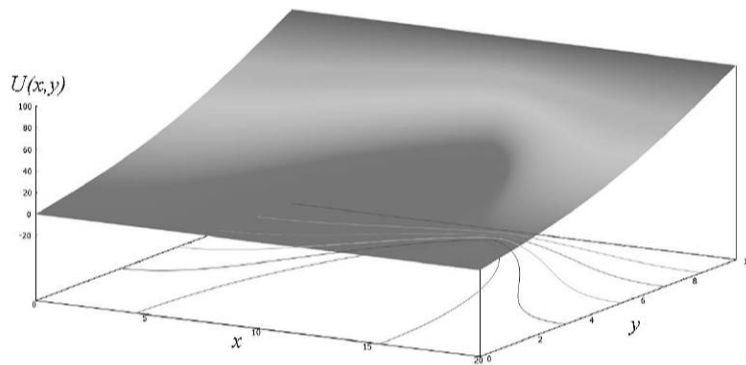


Рис. 2. Распределение потенциала при $x_1 = 50, \delta = 10^{-3}$

На рис. 1, 2 представлено распределение электростатического потенциала и картины эквипотенциальных линий во всей области исследуемой диодной системы.

4. Заключение. В данной работе представлены результаты моделирования двумерной диодной эмиссионной системы с полевым катодом лезвийной формы. Влияние полевого катода на распределение электростатического потенциала заменено влиянием произвольного числа заряженных нитей. Распределение потенциала во всей области

исследуемой системы получено в аналитическом виде (1)–(3). Данный способ моделирования систем на основе полевых катодов является более гибким по сравнению с методом, предложенным в работах [2, 3], где форма катода задается с помощью заряженной плоскости. Изменение границ системы, варьирование количества заряженных нитей, их координат, а также изменение значений плотностей зарядов нитей позволяют влиять на распределение потенциала в системе и с большой точностью получать требуемую форму катода.

Литература

1. Forbes R. G. Physical electrostatics of small field emitter arrays/clusters // Journal of Applied Physics. 2016. Vol. 120. No 5. Art. No 054302.
2. Виноградова Е. М., Доронин Г. Г. Математическое моделирование двумерной периодической системы полевых эмиттеров // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2020. Т. 16. № 2. С. 121–128.
3. Доронин Г. Г., Виноградова Е. М. Моделирование полевого эмиттера в двумерной диодной системе с диэлектрическими слоями // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 90–93.
4. Миролобов Н. Н. Методы расчета электростатических полей. М.: Высшая школа, 1963. 209 с.

УДК 519

Демьянович Ю. К., Булова И. Г., Доронина А. Г.,
Мирошниченко И. Д.

Об аппроксимации методом конечных элементов в плоской области

1. Введение. В данной работе дается обобщение аппроксимационных соотношений, в котором исключено требование их однозначной разрешимости. В результате модифицируется аппарат аппроксимации локальными функциями. Используется траектория функционалов в сопряженном пространстве, вводится понятие ленты для данной траектории, а также понятие ширины ленты в данной точке. Разработанный аппарат аппроксимации применяется к получению оценок аппроксимации одномерных и многомерных сплайнов, в частности, в случае аппроксимаций Р. Куранта и М. Зламала.

Вопросы аппроксимации тесно связаны с отысканием приближенных решений большинства тех задач, которые не удастся решить точно [1–5]. Для построения аппарата аппроксимации, необходимого при постановке задач моделирования различных явлений, широко применяются аппроксимационные соотношения. Упомянутые соотношения часто используются для определения координатных функций на некотором подразделении рассматриваемой области определения. Первыми в этом направлении были работы С. Г. Михлина [2] и Дж. Стрэнга. В них рассматривалась кубическая сетка. В дальнейшем аппроксимационные соотношения были рассмотрены для подразделений различных областей вплоть до произвольных дифференцируемых многообразий [3–5].

Демьянович Юрий Казимирович – профессор, Санкт-Петербургский государственный университет; e-mail: y.demjanovich@spbu.ru, тел.: +7(921)438-73-41

Булова Ирина Герасимовна – профессор, Санкт-Петербургский государственный университет; e-mail: i.g.burova@spbu.ru, burovaig@mail.ru тел.: +7(921)986-44-15

Доронина Александра Геннадьевна – ассистент, Санкт-Петербургский государственный университет; e-mail: a.doronina@spbu.ru, тел.: +7(963)246-00-59

Мирошниченко Ирина Дмитриевна – старший преподаватель, Санкт-Петербургский государственный университет; e-mail: irina_mir_@mail.ru, тел.: +7(921)380-62-65

Работа выполнена при финансовой поддержке СПбГУ, Pure ID 75207094

Цель данной работы – дать обобщение аппроксимационных соотношений с тем, чтобы расширить область их применения.

В качестве примера здесь получена равномерная оценка для аппроксимации Р. Куранта с константой, равной единице.

2. Аппроксимационные соотношения. Пусть M – замкнутая область в пространстве \mathbf{R}^n . Рассмотрим пространство $C^1(M)$ непрерывно дифференцируемых в M функций w с нормой $\|w\|_{C^1(M)} = \max_{\xi \in M} |w(\xi)|$. Эту норму будем обозначать иногда символом $\|w(\xi)\|_{C^1(\xi \in M)}$. Через $C^s(M)$ обозначим линейное нормированное пространство s раз непрерывно дифференцируемых функций $w(\xi)$, $\xi \in M$, с нормой $\|w\|_{C^s(M)} = \sum_{|\beta| \leq s} \|w^\beta(\xi)\|_{C(M)}$.

Рассмотрим выпуклую открытую область Ω евклидова пространства \mathbf{R}^n , предполагая, что $0 \in \Omega$. Символом $C^*(\bar{\Omega})$ обозначим линейное пространство линейных функционалов g над пространством $C(\bar{\Omega})$. Как правило, действие линейного функционала $g \in C^*(\bar{\Omega})$ на функцию $w(\xi)$, $w \in C(\bar{\Omega})$, записывают в виде $\langle g, w(\bullet) \rangle$ или в виде $\langle g, w \rangle$, а в некоторых случаях – в виде $\langle g, w(\xi) \rangle$.

Носителем ненулевого функционала g называется пересечение всех открытых множеств $\omega \in \mathbf{R}^n$, $\omega \cap \Omega \neq \emptyset$, обладающих следующим свойством: для любой функции $u \in C(\bar{\Omega})$ с носителем в множестве $\Omega \setminus \omega$ выполнено свойство $\langle g, u \rangle = 0$. Таким образом,

$$\text{supp } g = \bigcap_{\substack{\forall \omega, \omega \cap \Omega \neq \emptyset, \\ \text{supp } u \in \Omega \setminus \omega, \langle g, u \rangle = 0}} \omega.$$

Сужение функции $w(\xi)$ на множество M , $M \subset \bar{\Omega}$, обозначим $w|_M$ или $w(\xi)|_{\xi \in M}$. Условимся считать, что

$$\langle g, w|_{\text{supp } g} \rangle = \langle g, w \rangle.$$

Пусть для каждого $t \in \bar{\Omega}$ в пространстве $C(\bar{\Omega})$ определен линейный функционал $f(t)$, $\text{supp } f \subset \bar{\Omega}$. Множество $f = \{f(t) | \forall t \in \bar{\Omega}\}$ называется траекторией в пространстве $C^*(\bar{\Omega})$. Каждой точке $t \in \bar{\Omega}$ поставим в соответствие множество $S_t = \{[s, t] | \forall \xi \in \text{supp } f(t)\}$, где $[s, t]$ – прямолинейный отрезок, соединяющий точки s и t . Совокупность $B(f) = \{S_t | \forall t \in \bar{\Omega}\}$ называется *лентой траектории* f , а число

$$h_t(f) = \max_{\xi \in \text{supp } f(t)} \|\xi - t\|_{\mathbf{R}^n} \quad (1)$$

называется *шириной ленты* $B(f)$ в точке t .

Если $w \in C(\overline{\Omega})$, то имеем

$$|\langle f(t), w(\bullet) \rangle| = |\langle f(t), w(\xi) |_{\xi \in \text{supp}f(t)} \rangle| \quad \forall t \in \overline{\Omega},$$

так что

$$|\langle f(t), w(\bullet) \rangle| \leq \|f(t)\|_{C^*(\overline{\Omega})} \|w\|_{C(\text{supp}f(t))}.$$

Иначе говоря,

$$|\langle f(t), w(\bullet) \rangle| \leq \|f(t)\|_{C^*(\overline{\Omega})} \|w(\xi)\|_{C(\xi \in \text{supp}f(t))}. \quad (2)$$

Предположим, что выполнено свойство

$$\langle f(t), (\bullet - t)^\alpha \rangle = \delta_{0,\alpha} \quad \forall t \in \mathbf{R}^n, \quad |\alpha| \leq m. \quad (3)$$

Соотношения (3) будем называть *аппроксимационными соотношениями*.

Аппроксимационные соотношения (3) можно переписать в эквивалентном виде

$$\langle f(t), \varphi_\alpha(\bullet) \rangle = \varphi_\alpha(t) \quad \forall t \in \mathbf{R}^n, \quad |\alpha| \leq m,$$

где $\varphi_\alpha(t) = t^\alpha$.

По формуле Тейлора для функции $u \in C^{m+1}$ имеем

$$u(\xi) = \sum_{|\alpha| \leq m} \frac{1}{\alpha!} u^{(\alpha)}(x) (\xi - t)^\alpha + R_m(\xi, t), \quad (4)$$

где

$$R_m(\xi, t) = (m+1) \int_0^1 (1-\tau)^m \sum_{|\alpha|=m+1} \frac{(\xi-t)^\alpha}{\alpha!} u^{(\alpha)}(t+\tau(\xi-t)) d\tau. \quad (5)$$

Из соотношения (5) вытекает оценка

$$|R_m(\xi, t)| \leq C_m(u) \|\xi - t\|_{\mathbf{R}^n}^{m+1}, \quad (6)$$

где

$$C_m(u) = (m+1) \sum_{|\alpha|=m+1} \frac{\|u^{(\alpha)}\|_{C(\bar{\Omega})}}{\alpha!}. \quad (7)$$

Применяя $f(t)$ к соотношению (4), получаем

$$\langle f(t), u(\bullet) \rangle = \sum_{|\alpha| \leq m} \frac{1}{\alpha!} u^{(\alpha)}(t) \langle f(t), (\bullet - t)^\alpha \rangle + \langle f(t), R_m(\bullet, t) \rangle. \quad (8)$$

Используя предположение (3), из соотношения (8) находим

$$\langle f(t), u(\bullet) \rangle = u(t) + \langle f(t), R_m(\bullet, t) \rangle. \quad (9)$$

Полагая

$$\tilde{u} = \langle f(t), u(\bullet) \rangle, \quad (10)$$

формулу (9) перепишем в виде

$$\tilde{u}(t) - u(t) = \langle f(t), R_m(\bullet, t) \rangle. \quad (11)$$

Согласно (2) имеем

$$|\langle f(t), R_m(\bullet, t) \rangle| \leq \|f(t)\|_{C^*(\bar{\Omega})} \|R_m(\bullet, t)\|_{C(\text{supp} f(t))}, \quad (12)$$

где

$$\begin{aligned} \|R_m(\bullet, t)\|_{C(\text{supp} f(t))} &= \|R_m(\xi, t)\|_{C(\xi \in \text{supp} f(t))} = \\ &= \max_{\xi \in \text{supp} f(t)} |R_m(\xi, t)|. \end{aligned} \quad (13)$$

Используя неравенство (6), найдем

$$\max_{\xi \in \text{supp} f(t)} |R_m(\xi, t)| \leq C_m(u) \max_{\xi \in \text{supp} f(t)} \|\xi - t\|_{\mathbf{R}^n}^{m+1}. \quad (14)$$

Теорема 1. Пусть $f = \{f(t) | \forall x \in \overline{\Omega}\}$ – траектория в пространстве $C^*(\overline{\Omega})$, и пусть справедливы аппроксимационные соотношения (3). Для функции u из класса $C^{m+1}(\overline{\Omega})$ построим аппроксимацию \tilde{u} по формуле (10). Тогда справедливо следующее неравенство

$$|\tilde{u}(t) - u(t)| \leq C_m(u) \|f(t)\|_{C^*(\Omega)} [h_t(f)]^{m+1}, \quad (15)$$

где $C_m(u)$ определяется формулой (7), а величина $h_t(f)$ указана в соотношении (1).

Доказательство. Из формул (11) и (12), принимая во внимание (13), находим

$$\begin{aligned} |\tilde{u}(t) - u(t)| &\leq \|f(t)\|_{C^*(\Omega)} \|R_m(\xi, t)\|_{C(\xi \in \text{supp} f(t))} = \\ &= \|f(t)\|_{C^*(\Omega)} \max_{\xi \in \text{supp} f(t)} |R_m(\xi, t)|. \end{aligned}$$

Используя соотношение (14), имеем

$$|\tilde{u}(t) - u(t)| \leq \|f(t)\|_{C^*(\Omega)} C_m(u) \max_{\xi \in \text{supp} f(t)} \|\xi - t\|_{\mathbf{R}^n}^{m+1}. \quad (16)$$

Учитывая в формуле (16) обозначение (1) для ширины $h_t(f)$ ленты $B(f)$ у траектории f , получим соотношение (15).

Теорема 1 доказана.

3. Кусочно-линейная аппроксимация. Рассмотрим построение координатных функций Р. Куранта.

Кроме декартовых координат точки t в евклидовом пространстве \mathbf{R}^n в дальнейшем понадобятся барицентрические координаты. Пусть точки x_1, \dots, x_{n+1} пространства \mathbf{R}^n служат вершинами невырожденного симплекса T , являющегося открытым множеством (таким образом, симплекс T гомеоморфен открытому шару). Рассмотрим $(n+1)$ -компонентную вектор-функцию (столбец) $\varphi(t) = (1, t^{(1)}, \dots, t^{(n)})$. В дальнейшем все рассматриваемые векторы считаются вектор-столбцами (заметим, что мы их часто записываем в виде строки). Барицентрическими координатами точки $t = (t^{(1)}, \dots, t^{(n)})$ пространства \mathbf{R}^n называется упорядоченная совокупность чисел

$$\lambda(t) = (\lambda_1(t), \dots, \lambda_{n+1}(t)),$$

определяемых соотношениями

$$\lambda_j(t) = \frac{\det\left(\varphi(x_1), \dots, {}^{tj}\varphi(t), \dots, \varphi(x_{n+1})\right)}{\det\left(\varphi(x_1), \dots, \varphi(x_{n+1})\right)}, \quad (17)$$

где символ ${}^{tj}\varphi(t)$ означает, что j -й столбец определителя заменен столбцом $\varphi(t)$, $j = 1, \dots, n+1$. Как известно [1], барицентрические координаты обладают свойствами:

- 1) $\lambda_i(x_j) = \delta_{i,j}$,
- 2) $\sum_{i=1}^{n+1} \lambda_i(t) \equiv 1 \quad \forall t \in \mathbf{R}^n$,
- 3) $\lambda_i(t) > 0 \quad \forall i \in \{1, \dots, n+1\} \iff t \in T$.

Пусть область Ω имеет кусочно-линейную границу. Рассмотрим ее симплицальное подразделение \mathcal{T} . Множество $\mathcal{T}^0 = \{x_j \mid j \in J\}$ вершин x_j этого подразделения называется его нульмерным остовом; здесь J – конечное множество индексов. Пусть $(x_j^{(1)}, \dots, x_j^{(n)})$ – вектор, составленный из декартовых координат вершины x_j . Здесь и дальше все векторы считаем вектор-столбцами. Вершины называем также узлами рассматриваемого подразделения. Символами T обозначаем n -мерные открытые симплексы подразделения \mathcal{T} .

Выделим вершину x_i этого подразделения и рассмотрим замыкание объединения симплексов, инцидентных данной вершине. Это замыкание называется телом барицентрической звезды для вершины x_i . Обозначим его \mathcal{Z}_i . Координатная функция $\omega_i(t)$ аппроксимации Р. Куранта, соответствующая вершине x_i , имеет носителем упомянутую барицентрическую звезду \mathcal{Z}_i и может быть определена аппроксимационными соотношениями

$$\sum_{x_j \in T} \varphi(x_j) \omega_j(t) \equiv \varphi(t), \quad t \in T, \quad \text{supp } \omega_i = \mathcal{Z}_i, \quad i \in J. \quad (18)$$

Благодаря невырожденности симплекса T , определитель этой системы отличен от нуля (он представляет собой «ориентированный n -мерный объем» этого симплекса). Не нарушая общности, будем считать, что вершины x_j симплекса T занумерованы числами $1, \dots, n+1$.

Тогда решение системы (18) может быть представлено по формулам Крамера в виде

$$\omega_j(t) = \frac{\det(\varphi(x_1), \dots, {}^j\varphi(t), \dots, \varphi(x_{n+1}))}{\det(\varphi(x_1), \dots, \varphi(x_{n+1}))}. \quad (19)$$

Формулы (17)–(19) показывают, что $\omega_j(t)$ являются барицентрическими координатами точки t относительно вершин x_j симплекса T . Таким образом,

$$\omega_j(t) \equiv \lambda_j(t) \quad \forall j \in \{1, \dots, n+1\}.$$

В качестве траектории f в пространстве $C^*(\bar{\Omega})$ рассмотрим семейство функционалов $f(t)$, $t \in \bar{\Omega}$, определяемое формулой

$$\langle f(t), u \rangle = \sum_{x_j \in \bar{T}} u(x_j) \omega_j(t), \quad t \in \bar{T}, \quad T \in \mathcal{T}. \quad (20)$$

Из формул (1) и (20) следует, что ширина ленты рассматриваемой траектории f в точке $t \in T$ удовлетворяет неравенству

$$h_t(f) \leq \text{diam } T, \quad (21)$$

где $\text{diam } T = \sup_{\xi, \eta \in T} \|\xi - \eta\|_{\mathbf{R}^n}$. Замечая, что аппроксимационные соотношения (17) можно переписать в форме

$$\sum_{x_j \in T} x_j^\alpha \omega_j(t) \equiv t^\alpha, \quad t \in T, \quad |\alpha| \leq 1, \quad \text{supp } \omega_i = Z_i, \quad i \in J, \quad (22)$$

воспользуемся теоремой 1 при $m = 1$.

Теорема 2. *Рассмотрим траекторию f , определенную формулой (20). Для функции u из класса $C^2(\bar{\Omega})$ построим аппроксимацию \tilde{u} по формуле*

$$\tilde{u}(t) = \sum_{x_j \in \bar{T}} u(x_j) \omega_j(t), \quad t \in \bar{T}, \quad T \in \mathcal{T},$$

где функции ω_j получены из аппроксимационных соотношений (17). Тогда справедливо неравенство

$$|\tilde{u}(t) - u(t)| \leq \sum_{|\alpha|=2} \|u^{(\alpha)}\|_{C(\bar{\Omega})} (\text{diam } T)^2. \quad (23)$$

Доказательство. Воспользуемся формулой (15), полагая $m = 1$.
Имеем

$$|\tilde{u}(t) - u(t)| \leq C_1(u) \|f(t)\|_{C^*(\Omega)} [h_t(f)]^2, \quad (24)$$

где согласно (7)

$$C_1(u) = \sum_{|\alpha|=2} \|u^{(\alpha)}\|_{C(\bar{\Omega})}. \quad (25)$$

Кроме того, из соотношения (20),

$$|\langle f(t), u \rangle| \leq \max_{x_j \in \bar{T}} |u(x_j)| \sum_{x_j \in \bar{T}} |\omega_j(t)|, \quad t \in \bar{T}, \quad T \in \mathcal{T}. \quad (26)$$

Учитывая неотрицательность функций $\omega_j(t)$ и принимая во внимание соотношение (22) при $\alpha = (0, \dots, 0)$, имеем

$$\sum_{x_j \in \bar{T}} |\omega_j(t)| = 1. \quad (27)$$

Из (26) и (27) получаем

$$\|f(t)\|_{C^*(\Omega)} \leq 1. \quad (28)$$

Используя формулы (21), (25) и (28) в соотношении (24), получаем неравенство (23).

Теорема 2 доказана.

4. Заключение. Дано обобщение аппроксимационных соотношений с использованием траектории функционалов в сопряженном пространстве и понятия ширины ленты траектории в точке. Приведена оценка для аппроксимации Р. Куранта с константой, равной единице.

Литература

1. Сьярле Ф. Метод конечных элементов для эллиптических задач. М.: Мир, 1980. 512 с.
2. Михлин С. Г. Вариационно-сеточная аппроксимация // Записки научных семинаров ЛОМИ. 1974. Т. 48. С. 32–188.
3. Dem'yanovich Y. K., Gerasimov I. V. Local coarsening of simplicial subdivisions // Journal of Mathematical Sciences. 2016. Vol. 216. No 2. P. 219–236.
4. Diop E. H. S., Boudraa A.-O., Prasath V. B. S. Optimal nonlinear signal approximations based on piecewise constant functions // Circuits, Systems, and Signal Processing. Vol. 39. No 5. 2020. P. 2673–2694.
5. Демьянович Ю. К. Всплески в обобщенных пространствах Хаара // Проблемы математического анализа. Межвузовский сборник. 2020. Вып. 106. С. 615–634.

УДК 519.62

Зубахина Т. С.

**Анализ устойчивости методов
Рунге – Кутты – Чебышёва второго порядка
для уравнений с запаздыванием**

Рекомендовано к публикации доцентом Ереминым А. С.

1. Введение. Для систем обыкновенных дифференциальных уравнений (ОДУ), собственные числа которых расположены в длинной узкой полосе вдоль действительной отрицательной полуоси (например, получающихся при дискретизации по пространству некоторых уравнений в частных производных параболического типа), существуют явные стабилизированные методы Рунге – Кутты, которые по аналитической форме их функции устойчивости называются методами Рунге – Кутты – Чебышёва.

К сожалению, в литературе не удалось найти анализ применения таких методов к дифференциальным уравнениям с запаздывающим аргументом (ДУЗА).

Анализ устойчивости для чебышёвских методов первого порядка применительно к ДУЗА был проведён в [1]. В настоящей работе рассматривается расширение методов Рунге – Кутты – Чебышёва второго порядка [2] для решения ДУЗА и проводится анализ их P -устойчивости [3].

2. Формулировка чебышёвских методов второго порядка для ДУЗА. Для описания метода достаточно рассмотреть автономное уравнение с одним запаздыванием

$$y'(t) = f(y(t), y(t - \tau))$$

с заданной предысторией $y(t) = \varphi(t)$ при $t \leq t_0$. Пусть совершено n шагов и найдены приближения y_i к значениям $y(t_i)$, $i = 1, \dots, n$. Решение в точке $t_{n+1} = t_n + h$ s -этапным методом, расширяющим

Зубахина Татьяна Сергеевна – магистрант, Санкт-Петербургский государственный университет; e-mail: tanya_postcrossing@mail.ru, тел.: +7(904)330-41-91

метод решения ОДУ, предложенный ван дер Хауэном и Соммейером [2], находится по формулам

$$\begin{aligned}
Y_0 &= y_n, & F_0 &= f(y_n, \tilde{y}(t_n - \tau)), \\
Y_1 &= y_n + b_1 w_1 h F_0, \\
Y_i &= y_n + \mu_i h \left(f(Y_{i-1}, \tilde{y}(X_{i-1} - \tau)) - a_{i-1} F_0 \right) + \\
&\quad + \nu_i (Y_{i-1} - y_n) + \kappa_i (Y_{i-2} - y_n), \quad i = 2, \dots, s, \\
y_{n+1} &= Y_s,
\end{aligned} \tag{1}$$

где

$$\mu_i = \frac{2b_i w_1}{b_{i-1}}, \quad \nu_i = \frac{2b_i w_0}{b_{i-1}}, \quad \kappa_i = -\frac{b_i}{b_{i-2}}, \quad i = 2, \dots, s.$$

Параметры b_0 и b_1 являются свободными (мы рассматриваем случай $b_0 = b_1 = b_2$), а остальные параметры a_i , b_i , w_1 и w_0 вычисляются как $a_i = 1 - b_i T_i(w_0)$, $i = 0, \dots, s$,

$$b_s = \frac{T_s''(w_0)}{(T_s'(w_0))^2}, \quad w_1 = \frac{T_s'(w_0)}{T_s''(w_0)}, \quad w_0 = 1 + \frac{\eta}{s^2},$$

где $T_k(x)$ – полином Чебышёва первого рода степени k . Абсциссы этапов X_i , $i = 0, \dots, s-1$ находятся по тем же формулам (1), что и Y_i , если положить $f \equiv 1$ и y_n заменить на t_n . Неотрицательный параметр η называется коэффициентом демпфирования. При его росте область комплексной плоскости λh , в которой метод Рунге–Кутты–Чебышёва устойчив для ОДУ $y' = \lambda y$, $\lambda \in \mathbb{C}$ несколько укорачивается вдоль действительной оси, но утолщается вдоль мнимой, что позволяет избежать её сужения до нуля в некоторых точках.

Значения запаздывающего решения находятся через *непрерывное расширение* метода $\tilde{y}(t)$, которое полагается равным $\varphi(t)$ при $t \leq t_0$, а на $(k+1)$ -м шаге при $t \in [t_k, t_{k+1}]$ строится интерполяционный полином, который в каждой точке $t_k + \theta h$, $\theta \in [0, 1]$ обеспечивает некоторый требуемый порядок точности. Для сохранения второго порядка сходимости метода (1) при решении ДУЗА достаточно использовать линейную интерполяцию между точками сетки:

$$\tilde{y}(t_k + \theta h) = y_k(1 - \theta) + y_{k+1}\theta, \quad \theta \in [0, 1]. \tag{2}$$

Более точным будет использование квадратичного интерполянта,

который легко построить, если в записи (1) выразить y_{n+1} через $f(Y_i, \tilde{y}(X_i - \tau))$, $i = 0, \dots, s - 1$ (как это делается в традиционной записи методов Рунге–Кутты [4]), и положить коэффициенты квадратичными полиномами от θ :

$$\tilde{y}(t_k + \theta h) = y_k + h \sum_{i=0}^{s-1} p_i(\theta) f(Y_i, \tilde{y}(X_i - \tau)), \quad p_i(\theta) = p_{i1}\theta + p_{i2}\theta^2. \quad (3)$$

Детали построения *непрерывных расширений* см., например, в [3, 4]. Отметим, что квадратичный интерполянт (3) определяется неоднозначно при $s \geq 3$.

3. Линейный анализ устойчивости. Основой для анализа устойчивости численных методов решения ДУЗА является обобщение тестового уравнения Далквиста [3]:

$$y'(t) = \lambda y(t) + \mu y(t - \tau). \quad (4)$$

Запаздывание τ считается постоянным.

Один из возможных вариантов обобщения A -устойчивости на ДУЗА (4) приводит к понятию P -устойчивости.

Определение 1. Область P -устойчивости численного метода для решения ДУЗА (4) – это набор S_P таких пар комплексных чисел (α, β) , где $\alpha = h\lambda$, $\beta = h\mu$, что численное решение $\{y_n\}_{n \geq 0}$ системы (4), полученное на постоянном шаге $h = \tau/m$, $m \in \mathbb{N}$, по абсолютному значению стремится к нулю при $n \rightarrow \infty$ для всех функций предыстории $\varphi(t)$ и всех постоянных τ . В настоящей работе рассматриваются только действительные значения λ и μ (и, соответственно, α и β).

При применении численного метода к (4) получается разностное уравнение, устойчивость которого определяется корнями характеристического полинома. Их можно выразить в форме

$$\xi = R^* \left(\alpha, \frac{\beta}{\xi^m} \right).$$

Здесь рациональная функция P -устойчивости имеет вид

$$R^*(\alpha, z) = 1 + (\alpha + z)p^T(I - \alpha A - zP)^{-1}e, \quad (5)$$

где вектор p и матрица A составлены из коэффициентов метода

Рунге–Кутты в его традиционной записи, а P – матрица, соответствующая интерполяционному полиному в этапных абсциссах.

С использованием функции (5) область P -устойчивости можно записать в виде:

$$P = \{(\alpha, \beta) \in \mathbb{C}^2 : \alpha \in S_A \text{ и } |\beta| < \sigma_\alpha\}, \quad (6)$$

где

$$S_A = \{\alpha \in \mathbb{C} : |R^*(\alpha, 0)| < 1\},$$

$$\sigma_\alpha = \inf_{z \in \Gamma_\alpha} |z|, \quad \Gamma_\alpha = \{z \in \mathbb{C} : |R^*(\alpha, z)| = 1\}.$$

4. Трёхэтапный метод. В силу того, что при $s = 2$ с точки зрения устойчивости явные методы Рунге–Кутты второго порядка не имеют никакой свободы, рассмотрение чебышёвских методов второго порядка имеет смысл проводить при $s \geq 3$. Рассмотрим недемпфированный метод (1) с тремя этапами ($\eta = 0, s = 3$):

$$A = \begin{bmatrix} 0 & 0 & 0 \\ \frac{3b_1}{8} & 0 & 0 \\ \frac{6b_1-3}{16b_1} & \frac{3}{16b_1} & 0 \end{bmatrix}, \quad b = \begin{bmatrix} \frac{b_1-4}{9b_1} & \frac{4}{9b_1} & \frac{8}{9} \end{bmatrix},$$

где b_1 – свободный параметр.

При линейной интерполяции функция устойчивости вне зависимости от b_1 имеет вид

$$R_1^*(\alpha, z) = \frac{-\alpha^3 - (z+8)\alpha^2 - (7z+16)\alpha - 8z - 16}{\alpha z + 8z - 16}.$$

На рис. 1 (слева) представлен график области, где $|R_1^*(\alpha, z)| \leq 1$.

При квадратичной интерполяции есть два свободных параметра. Одна из возможных функций устойчивости имеет вид

$$R_4^*(\alpha, z) = \frac{(31\alpha + 168)z^2 + (63\alpha^2 + 423\alpha + 416)z\alpha + 512}{512 + 8z^2 + (7\alpha - 96)z} + \frac{32\alpha^3 + 256\alpha^2 + 512\alpha + 512}{512 + 8z^2 + (7\alpha - 96)z}. \quad (7)$$

График области устойчивости приведён на рис. 1 (справа). Отметим, что мы провели анализ для многих вариантов выбора свободных

параметров при квадратичной интерполяции, однако для всех них получились «провалы» к нулю в точке $\alpha = -4$. При линейной интерполяции этот «провал» становится вертикальной прямой, то есть, метод куда более устойчив, чем при интерполяции второго порядка.

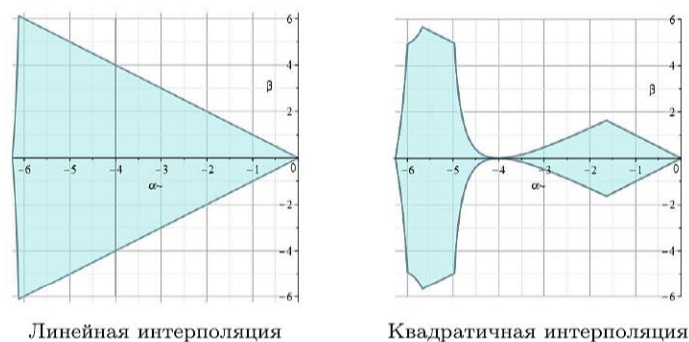


Рис. 1. Области устойчивости трёхэтапного метода с использованием интерполяции разных порядков

5. Заключение. Вероятно, при применении методов РКС второго порядка для ДУЗА, аналогичное преимущество линейной интерполяции сохранится и для большего числа этапов. Наблюдаемые «провалы» исчезают при повышении коэффициента демпфирования, что приводит к сильному сокращению области вдоль оси α .

Литература

1. Зубахина Т. С. ВКР бакалавра: Анализ устойчивости методов Рунге–Кутты–Чебышёва для уравнений с запаздыванием. СПбГУ, 2020.
2. Van der Houwen P., Sommeijer B. On the internal stage Runge-Kutta methods for large m -values // Z. Angew. Math. Mech. 1980. Vol. 60. P. 479–485.
3. Bellen A., Zennaro M. Numerical Methods for Delay Differential Equations. Oxford University Press, 2013. 413 p.
4. Хайрер Э., Нёрсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежёсткие задачи. М.: Мир, 1990. 512 с.

УДК 519.688

Казанков В. К., Холодова С. Е.

Исследование и анализ уравнения теории идеальной несжимаемой электропроводящей жидкости

1. Введение. Рассматривается краевая задача магнитной гидродинамики, которая в случае длинных волн малой амплитуды редуцируется к одному уравнению. Решение данного дисперсионного уравнения сводится к поиску корней полинома восьмой степени с комплексными коэффициентами. В статье представлен численный метод и алгоритм, созданный на основе машинного обучения и итерационного численного метода, который позволяет отыскивать различные корни данного полинома. Проведен анализ зависимости решения от магнитного числа Рейнольдса, который позволяет оценить качественно динамику процесса.

2. Постановка задачи. Динамика несжимаемой электропроводящей жидкости описывается системой дифференциальных уравнений в частных производных [1]:

$$\frac{\partial \mathbf{v}}{\partial t} + \langle \mathbf{v} \cdot \nabla \rangle \mathbf{v} = -\frac{\nabla p}{\rho} - 2\boldsymbol{\omega} \times \mathbf{v} - g\mathbf{z} + \frac{1}{\mu\rho} [\text{rot } \mathbf{b} \times \mathbf{b}],$$

$$\frac{\partial \mathbf{b}}{\partial t} = \text{rot} [\mathbf{b} \times \mathbf{v}] + \frac{1}{\mu\sigma} \Delta \mathbf{b},$$

$$\text{div } \mathbf{b} = 0,$$

Казанков Владислав Константинович – студент, Университет ИТМО;
e-mail: v.kazankov98@gmail.com, тел.: +7(977)300-20-48

Холодова Светлана Евгеньевна – доцент, Университет ИТМО; e-mail:
kholodovase@itmo.ru, тел.: +7(965)021-00-41

с граничными условиями

$$\begin{aligned}
p(x, y, -h_B) &= p_0(x, y, z), \quad \mathbf{b}(x, y, -h_B) = \mathbf{b}_0(x, y, z), \\
v_z(x, y, -Z, t) &= -v_x \frac{\partial Z}{\partial x} - v_y \frac{\partial Z}{\partial y}, \\
v_z(x, y, -h_B, t) &= -\frac{\partial h_B}{\partial t} - v_x \frac{\partial h_B}{\partial x} - v_y \frac{\partial h_B}{\partial y}, \\
b_n(x, y, -h_B, t) &= B_{n0}(x, y, t), \\
b_x \frac{\partial Z}{\partial x} + b_y \frac{\partial Z}{\partial y} + b_z &= b_{n0}^{(e)}(x, y, t),
\end{aligned}$$

где \mathbf{v} – скорость жидкости в системе, вращающейся со скоростью ω , p – давление, ρ – плотность, \mathbf{b} – вектор магнитной индукции поля, μ – магнитная проницаемость, σ – электрическая проводимость, $p_0, \mathbf{b}_0, B_{n0}, b_{n0}^{(e)}, Z$ – заданные функции, h_B – неизвестная свободная поверхность жидкого слоя. В случае длинных волн малой амплитуды решение исходной краевой задачи сводится к решению одного скалярного уравнения

$$\begin{aligned}
D (D_t^2 + \alpha^2)^2 \left\{ \left[\left(D_t - \frac{\Delta}{Rm} \right) D_t - \frac{D^2}{\mu\rho} \right] [\Delta_2 \xi + \langle \nabla \ln H_0, \nabla \xi \rangle] - \right. \\
\left. - \frac{\mathfrak{D} \left(\ln H_0, \alpha \left(D_t - \frac{\Delta}{Rm} \right) \xi \right)}{\mathfrak{D}(x, y)} \right\} + \frac{1}{g(\mu\rho)^2 H_0} (D_t^2 + \alpha^2) [F^2 + \\
+ (\alpha D^2)^2] \left[D\xi + \frac{B_{n0}}{2H_0\sqrt{f}} \langle \nabla \xi, \nabla (H_0 + Z) \rangle \right] = \frac{DH_0 + b_{n0}^{(e)}}{(\mu\rho)^2 H_0}
\end{aligned} \tag{1}$$

для некоторой модифицированной функции возмущения глубины слоя ξ [1].

В работах [2, 3] показано существование решения уравнения (1) в виде гармоника

$$\xi = \text{Im}(A \exp(i(kx + ly - \sigma t))), \tag{2}$$

причем частота σ является решением дисперсионного уравнения

$$P_n(\sigma) = \sum_{k=0}^n c_k \sigma^k,$$

где $c_k \in \mathbb{C}$ определяются из соотношений физических величин, которые в дальнейшем будем называть физическими параметрами. Корни полинома $P_n(\sigma)$ обозначим за $\tilde{\sigma}$. Согласно основной теореме алгебры, $\tilde{\sigma}$ ровно n штук с учетом их кратностей. Каждый характеризует динамику исследуемого процесса.

Для анализа уравнения (1) следует найти корни $\tilde{\sigma}$ полинома $P_n(\sigma)$. В рассматриваемом случае $n = 8$, и уравнение

$$P_n(\sigma) = 0$$

не позволяет применить аналитические подходы.

Итерационные численные методы, основанные на построении касательных полиномов $P_q(\sigma)$, где $q < n$, обладают высокой скоростью сходимости к корню $\tilde{\sigma}$ из начального приближения σ^0 , иначе говоря, $\sigma^0 \rightarrow \tilde{\sigma}$. При этом $\tilde{\sigma} \in Q$, где Q принадлежит внутренности пересечения поверхностей, образованных полиномами $P_q(\sigma)$ и $P_n(\sigma)$.

Возникает необходимость в выборе набора начальных приближений $\{\sigma_k^0\}$ таких, чтобы $\sigma_k^0 \rightarrow \tilde{\sigma}_j$, где $\{\tilde{\sigma}_j\}_{j \leq n}$ – множество уникальных корней $\tilde{\sigma}_j$. Обычно для генерации $\{\sigma_k^0\}$ используется метод Монте-Карло, который не имеет под собой достаточного аналитического обоснования. Для решения поставленной задачи рассмотрим универсальный метод поиска $\tilde{\sigma}$ на основе машинного обучения.

3. Алгоритм. Архитектура алгоритма состоит из трех последовательных блоков:

1. Роение частиц.
2. Итерационный метод.
3. Пост-обработка.

3.1. Роение частиц. Под частицей будем подразумевать метод имитации отжига, который минимизирует функционал вида

$$L(\sigma) = |P_n(\sigma)|.$$

Так как минимизация проводится не в одномерном пространстве, а на комплексной плоскости, то возникает необходимость в модификации классического метода [4]. В случае оптимизации в одномерном пространстве происходит генерация одного возможного перехода σ^1

из σ^0 . Модификация состоит в рассмотрении множества $\{\sigma_j^1\}$, которое вложено в одномерную сферу радиуса ε с центром в точке σ^0 , где $\varepsilon = \varepsilon_1 + \varepsilon_2$ состоит из случайного числа ε_1 , равномерно распределенного на отрезке $[0, 1]$, и величины гарантированного шага $\varepsilon_2 \sim 1 \cdot 10^{-5}$.

Тогда

$$\sigma^1 = \min_{\sigma \in \{\sigma_j^1\}} L(\sigma).$$

Затем последовательность действий в итерации остается неизменной. Пусть N – количество частиц в рое, т. е. требуется ровно N раз применить метод имитации отжига для функционала $L(\sigma)$. Начальным приближением назовем $\{\sigma_k^0 : L(\sigma_k^0) \leq u\}$, где u – пороговое значение такое, что $u \sim 10$.

3.2. Итерационный метод. В статье [5] представлен алгоритм, обобщающий итерационный метод Ньютона. Его реализация представляется вторым блоком. Для $\{\sigma_k^0\}$ достаточно 50 итераций, чтобы гарантировать точность порядка $1 \cdot 10^{-15}$. Обозначим за M множество кандидатов в корни полинома $P_n(\sigma)$, полученное из множества $\{\sigma_k^0\}$ после второго блока.

3.3. Пост-обработка. Завершающий блок состоит из добавления в множества M комплексно-сопряженных чисел $\bar{\sigma}$ и проверки результирующего множества Q на условия:

- $L(\sigma) \leq 1 \cdot 10^{-15}$;
- если $L(\sigma) \geq L(\text{Re}(\sigma))$, то $\sigma \in Q, \forall \sigma \in M$;
- $\sigma_i \neq \sigma_j$, при $i \neq j, \forall \sigma_i, \sigma_j \in Q$.

4. Реализация модели и результаты. Итоговая модель имеет физические входные параметры: A – амплитуда гармоника, H – значение невозмущенной глубины слоя, g – величина ускорения свободного падения, α – параметр Кориолиса, μ – магнитная проницаемость, ρ – плотность, Rm – магнитное число Рейнольдса, k и l – волновые числа, b_{0x} и b_{0y} – проекции вектора магнитной индукции внешнего поля на оси x и y , $b_{n0}^{(e)}$ – нормальная компонента вектора магнитной индукции на ограничивающей жидкий слой поверхности, ΔZ – оператор Лапласа, примененный к граничной поверхности Z слоя; и программные параметры: N – количество частиц в рое, r –

радиус шара, из которого генерируются случайные числа и подаются в первый блок, h – шаг дискретизации зависимости физического параметра от $\tilde{\sigma} \in Q$.

Представленную модель удалось реализовать на языке программирования Python с использованием библиотек NumPy, Pandas, Tqdm, Matplotlib.

На рис. 1 представлены зависимости $\tilde{\sigma}$ от магнитного числа Рейнольдса $Rm \in [0,01; 20]$. Правый график показывает изменение $Re(\tilde{\sigma})$, а левый – $Im(\tilde{\sigma})$, остальные физические параметры фиксировались равными 1, кроме $g = 9,8$, с шагом $h = 0,2$ и $r = 100$.

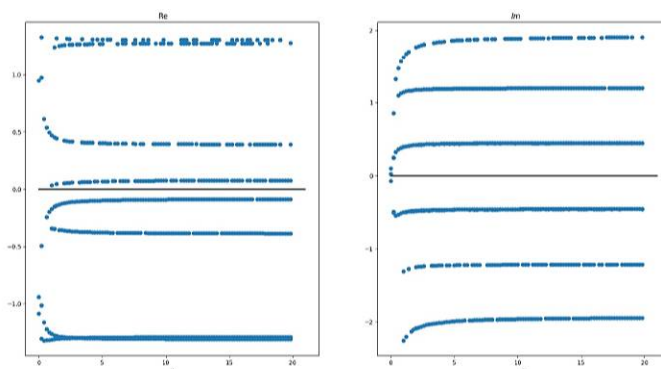


Рис. 1. Рой состоит из пяти частиц

Представленная графическая реализация позволяет произвести качественный анализ существующих магнитогидродинамических режимов жидкой среды. Пусть $\sigma = \sigma_1 + i\sigma_2$, тогда (2) представимо в виде

$$\xi = -A \sin(kx + ly) \sin(\sigma_1 t) e^{\sigma_2 t}.$$

Качественное исследование проведено для $Rm = 0,01$, результат представлен на рис. 2. Из представленной визуализации динамики процесса можно сделать вывод о существовании вполне периодического режима на некотором промежутке времени, при $t \rightarrow \infty$ процесс приобретает неустойчивый характер. Кроме того, из рис. 1 видно,

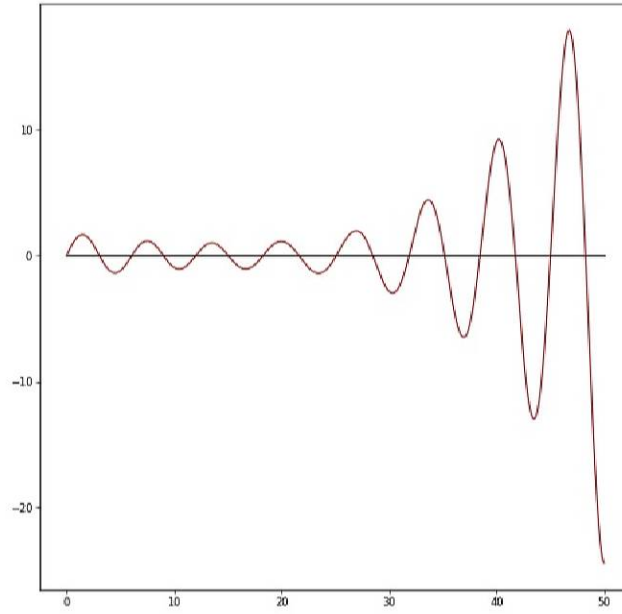


Рис. 2. Визуализация динамики процесса

что существуют $\sigma_2 > 0$, поэтому на большом временном промежутке не возникает устойчивых режимов. Согласно основной теореме алгебры, у $P_n(\sigma)$ существует хотя бы один комплексный корень, следовательно, у процесса не может быть периодического режима, так как для его существования требуется, чтобы $\forall \tilde{\sigma} \in \mathbb{R}$.

5. Заключение. Разработан метод поиска корней полинома произвольной степени в комплексной области с комплексными коэффициентами, реализован алгоритм на языке программирования Python. Представлен качественный и численный анализ динамики идеальной вращающейся электропроводящей жидкости переменной глубины.

Литература

1. Peregudin S., Peregudina E., Kholodova S. On integration of the system of MHD equations modeling wave processes in a rotating liquid with arbitrary magnetic Reynolds number // Journal of Physics: Conference Series. 2019. Vol. 1268. No 1. Art. No 012055.
2. Peregudin S., Peregudina E., Kholodova S. The influence of dissipative effects on dynamic processes in a rotating electrically conductive liquid medium // Journal of Physics: Conference Series. 2019. Vol. 1359. No 1. Art. No 012118.
3. Холодова С. Е. Динамика вращающегося слоя идеальной электропроводной несжимаемой жидкости // Вычислительная математика и математическая физика. 2008. Т. 48. № 5. 882–898.
4. Шамин Р. В. Машинное обучение в задачах экономики. М: Грин Принт, 2019. 140 с.
5. Иванисов А. В., Полищук В. К. Метод нахождения корней полиномов, сходящийся при любом начальном приближении // Вычислительная математика и математическая физика. 1985. Т. 25. № 5. 643–653.

УДК 532.591

Онегов Н. А.

**Анализ влияния внутренних волн
на процесс разлива нефти
и на основные характеристики флюида
в нефтеносном пласте**

Рекомендовано к публикации профессором Перегудиным С. И.

Рекомендовано к публикации доцентом Шагиахметовым А. М.

1. Введение. Волны на поверхности жидкости – характерное физическое явление для нашего мира. Большой вклад в развитие теории движения жидкости внесли Леонард Эйлер и Жозеф Луи Лагранж. В работе [1] подробно рассматривается линейная волновая теория, в монографии [2] детально изучается вопрос внутренних возмущений в жидкости. В современном мире, когда все приборы настроены на максимальную точность, очень остро стоит вопрос о том, следует ли учитывать все показатели или некоторыми характеристиками можно пренебречь. В представленной работе рассматривается влияние волн малой амплитуды в нефтеносном пласте на основные характеристики флюида, изучается роль возмущенного воздействия в процессе разлива, решается задача значимости рассматриваемых показателей в реальных условиях.

2. Расчет параметров волны. Рассмотрим задачу о движении двух слоев идеальной неоднородной жидкости над горизонтальным твердым недеформируемым дном. Так в первом приближении можно описать пласт, в котором нефть согласно закону Архимеда, находится на водном зеркале. Все величины, характеризующие нижний слой, обозначаются индексом 1, а верхний – индексом 2. Введем декартову систему координат таким образом, чтобы невозмущенная поверхность слоев жидкости имела нулевую аппликату. Исходя из результатов исследований [2, 3], получим краевую задачу для системы уравнений с частными производными, решения которой – волновые функции имеют следующий вид:

Онегов Никита Андреевич – студент, Санкт-Петербургский горный университет; e-mail: feniks8000@gmail.com, тел.: +7(953)153-37-99

$$\begin{aligned}
\dot{v}_{j_z}(z, x, y, t) &= V_j(z)M_j(z, x, y, t), \\
\dot{v}_{j_x}(z, x, y, t) &= (R_j(z)V_j(z) - k_1V_j'(z)) \frac{M_j(z, x, y, t)}{i|\bar{k}|^2}, \\
\dot{v}_{j_y}(z, x, y, t) &= (W_j(z)V_j(z) - k_2V_j'(z)) \frac{M_j(z, x, y, t)}{i|\bar{k}|^2}, \\
\dot{p}_j(z, x, y, t) &= \frac{\tilde{p}_j}{ir_j(z)} V_j(z)M_j(z, x, y, t), \\
\dot{\rho}_j(z, x, y, t) &= \frac{\tilde{\rho}_j}{ir_j(z)} V_j(z)M_j(z, x, y, t), \\
\dot{\eta}_1(z, x, y, t) &= \frac{A_1^1 \chi}{ir_1(0)} M_1(z, x, y, t), \\
\dot{\eta}_2(z, x, y, t) &= \frac{\chi}{ir_2(H_2)} V_2(H_2)M_2(z, x, y, t), \\
M_j(z, x, y, t) &= \exp \left(i(k_1x + k_2y - \omega t) + \int_0^z \alpha_j(\epsilon) d\epsilon \right), \quad \alpha_j(z) = \frac{N_j^2(z)}{2g}, \\
R_j(z) &= \frac{k_1k_2v_j' - k_2^2u_j'}{r_j(z)} - k_1\alpha_j(z), \\
W_j(z) &= \frac{k_1k_2u_j' - k_2^2v_j'}{r_j(z)} - k_2\alpha_j(z), \\
V_1(z) &= \chi(A_1^1 \cos(\sqrt{q_1}z) + A_2^1 \sin(\sqrt{q_1}z)), \\
V_2(z) &= \chi(A_1^2 \cos(\sqrt{q_2}z) + A_2^2 \sin(\sqrt{q_2}z)), \\
A_1^1 &= \frac{C_1C_4\sqrt{q_2} \cos(\sqrt{q_1}H_1)}{\sin(\sqrt{q_1}H_1)C_2\sqrt{q_1}C_1 + (C_1C_3 - C_5) \cos(\sqrt{q_1}H_1)}, \\
A_1^2 &= \frac{A_1^1}{tg(\sqrt{q_1}H_1)}, \quad A_2^1 = \frac{A_1^1}{C_1}, \quad A_2^2 = \bar{\xi},
\end{aligned}$$

где $\bar{\xi}$ – произвольная действительная величина, N_j – частота плавучести j -й жидкости, а χ – случайная действительная постоянная.

3. Исследование и оценка влияния возмущений на параметры флюида в пласте. Предполагается, что плотность жидкости не зависит от глубины напрямую в связи с недостаточной эффективной мощностью пласта для значимой стратификации флюида, а изменяется только от компоненты, создаваемой возмущениями. Поверхности представляются функциями от времени и глубины. Координаты x и y используются при расчетах только как начальная фаза периодической функции, для вычислений принимаются за единицу. Рассмотрим нефтяной пласт со следующими параметрами, представленными в СИ:

$$\tilde{p}_j = 18 \cdot 10^6, \quad H_1 = 20, \quad H_2 = 15, \quad \tilde{\rho}_2 = 910, \quad \tilde{\rho}_1 = 1020.$$

Принимается, что $u_j = v_j = v_j$, истинная мощность пласта $K = 300$, характеристика $\chi = 1$, а модуль амплитуды $|A_2^2| = 0,13$. Согласно информации, представленной в источниках [1, 4], примем значения параметров, которые описывают рассматриваемое движение:

$$v_1 \approx v_2 = 0,176, \quad \lambda_1 \approx \lambda_2 = 0,0044.$$

В модели экспоненциальной стратификации частота внутренних колебаний $N_j(z)$ постоянна, и закон дисперсии, связывающий частоту волны ω и волновой вектор k , для коротких волн можно вычислить при помощи приближения Буссинеска, для рассматриваемого движения $\omega_j = N_j$.

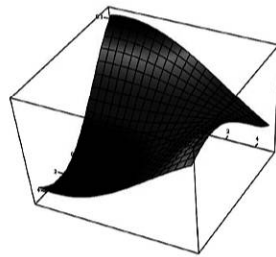


Рис. 1. Зависимость изменения плотности от времени и глубины

Выбранные характеристики являются грубыми, в частности, в силу того, что период убывает с увеличением высоты, а длина волны обратно пропорциональна частоте. Поэтому представленные расчеты позволяют дать только лишь достаточно точную оценку реального влияния. Изменение плотности нефти задается поверхностью (см. рис. 1). Как наглядно показывает график, построенный на основе описанных

выше расчетов, амплитуда колебания плотности не превышает значения 0,15. Проведем проверку рассматриваемых расчетов в условный момент времени $t = 13$. Амплитуда возмущений в данный момент времени равна 0,068, усредненное по синусу значение колебаний амплитуды возмущенной поверхности $\rho'_2 = 0,034$. Значения амплитуд возмущенного движения поверхности нефти и раздела флюидов соответственно равны $2,294 \cdot 10^{-4}$ и $-1,931 \cdot 10^{-4}$, плотность по определению есть $\rho = \frac{m}{V}$, $m = \rho_2 V$, $V = 900000$, изменение объема нефти $\Delta V = V \frac{\eta_1(z, x, y, t) - \eta_2(z, x, y, t)}{H_2}$. Тогда, с учетом полученных значений, $\Delta\rho_2 = 0,026$, а погрешность методов оценки

$$\nabla = \frac{\Delta\rho_2 - \rho'_2}{\rho'_2} \cdot 100\% = 23,53\%,$$

что говорит о большой разности, причиной накопления которой являются допущения, описанные выше, а также тот факт, что расчет плотности по определению не учитывает возмущенное движение. Однако для прикладных задач нефтегазовой отрасли эти показатели ничтожно малы, чтобы в действительности повлиять на процесс эксплуатации и разработки:

$$\nabla_1 = \frac{\Delta\rho_2}{\rho_2} \cdot 100\% = 2,857 \cdot 10^{-3}\%,$$

$$\nabla_2 = \frac{\rho'_2}{\rho_2} \cdot 100\% = 3,736 \cdot 10^{-3}\%.$$

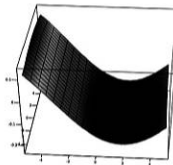


Рис. 2. Зависимость изменения давления от времени и глубины но, что оказываемое волнами малой амплитуды дополнительное влияние на давление не окажет никако-

Если учесть градиент плотности, силы, действующие в порах, вязкость, межмолекулярные и капиллярные взаимодействия, которые постоянно присутствуют в пласте, то влияние данного волнового возмущения становится ничтожно мало. Зависимость давления представляется поверхностью, изобра-

го влияния на добычу флюида. Для успешной и более эффективной эксплуатации месторождения знание характера этой зависимости не является необходимым, также эти колебания не смогут повлиять и на классифицирование нефти по плотностному параметру.

4. Влияние волн малой амплитуды на характер разлива нефти. Компании заранее должны оценивать потенциальные риски и ущерб, который может произойти в случае разливов. Часто разлив происходит в водной среде или рядом с ней, поэтому нефть так или иначе попадает на территорию океанов, морей и рек. Для того чтобы оценить характер влияния данного волнового возмущения, следует вычислить проекции скорости на оси. Глубина водного слоя принимается равной 60 м. По данным исследований [5, 6] толщина нефтяной пленки принимается 13 мм.

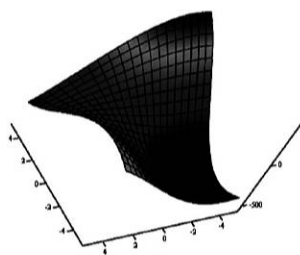


Рис. 3. Зависимость изменения компоненты скорости волны по осям абсцисс и ординат от времени и глубины, масштаб 1000000:1

Зависимости характеризуются графиком, изображенным на рис. 3. Без учета сил трения, вязкости, сопротивления воздуха, солености воды и влияния ветра, которые имеют место на водоемах, модуль скорости на своем пике достигает примерно $500 \cdot 10^{-6}$. Следовательно, волны малой амплитуды не могут значительно повлиять на разлив при его своевременной ликвидации.

5. Вывод. В данной работе рассмотрена модель нефтеносного пласта, построены поверхности колебательного движения и изучено влияние волн малой амплитуды на основные параметры флюида. Доказано, что волновое возмущение оказывает малое влияние на характеристики жидкостей в пласте и на их разлив даже без учета вязкости, градиента плотности и т. д., поэтому в реальных условиях воздействием гидродинамических параметров можно пренебречь.

Литература

1. Harald E. Krogstad, Oilvind A. Arntsen Linear wave theory Part A-Regular waves. Trondheim: Norwegian university of science and technology trondheim, 2000. 42 с.
2. Перегудин С. И. Волновые движения в жидких и сыпучих средах. СПб.: Изд-во Санкт-Петерб. ун-та, 2004. 288 с.
3. Алешков Ю. З. Волны на поверхности сыпучих сред, вызванные потоком жидкости // Вестник Санкт-Петербургского университета. Сер. 1. 2002. Вып. 4. № 25. С. 35–43.
4. Цапаев С. Гравитационно-капиллярные волны на поверхности жидкости. Долгопрудный: Изд-во Московского физико-технического университета, 2019. 13 с.
5. Куликова И. С., Мерициди И. А. Особенности локализации разливов нефти из подводных переходов трубопроводов в условиях открытой воды // Территория Нефтегаз. 2012. № 5. С. 48–53.
6. Муратов И. Ф. Способ оценки толщины слоя нефти над водой и устройство для его реализации // Молодой ученый. 2018. № 51. С. 47–49.

УДК 539.3

Пегливанова М. М.

Коррозия пластины из упругопластического материала с SD-эффектом

Рекомендовано к публикации профессором Прониной Ю. Г.

1. Введение. Коррозия наносит большой урон экономике всех стран. Потери от коррозии в среднем составляют около десяти процентов ежегодно производимой массы металлов. При этом выход из строя отдельных деталей может привести к серьезным авариям, поэтому при оценке прочности конструкций очень важно учитывать наличие коррозионных процессов [1]. Данная работа посвящена определению срока службы пластины из идеального упругопластического материала, δ подверженной коррозии с постоянной скоростью под действием изгибающего момента. Коррозия оболочек из упругопластического материала изучалась в работах [2–4]. В отличие от указанных работ здесь рассматривается пластина из материала с SD эффектом, т. е. с различными пределами текучести на растяжение и сжатие. Задачи об изгибе балок с SD-эффектом исследованы в [5]. Полученные в данной работе результаты при нулевой скорости коррозии совпадают с решением, найденным в [5].

2. Постановка задачи. Рассматривается задача о чистом изгибе прямоугольной пластины постоянным изгибающим моментом M . Пусть под действием внешней среды пластина равномерно корродирует (растворяется) по лицевым поверхностям с постоянной суммарной (по обеим поверхностям) скоростью a . Считаем, что в начальный момент времени $t = 0$ толщина пластины равна h_0 . Материал пластины является идеальным упругопластическим и обладает SD-эффектом с пределом текучести на растяжение σ_y^+ и пределом текучести на сжатие $\sigma_y^- = d\sigma_y^+$, где $d \geq 1$.

Требуется определить время t^* , за которое пластина полностью перейдет в пластическое состояние.

3. Решение. Совместим нейтральную плоскость пластины с плоскостью Oxz . Ось Oy направим вниз. Для решения будем ис-

Пегливанова Мария Михайловна – студент, Санкт-Петербургский государственный университет; e-mail: st067846@student.spbu.ru, тел.: +7(928)196-38-81

пользовать техническую теорию изгиба. Будем считать, что величина изгибающего момента такая, что в начальный момент времени в пластине возникают только упругие напряжения $\sigma_{xx} = yM/I$. При уменьшении толщины пластины напряжения в пластине начинают возрастать и в какой-то момент времени достигнут предела текучести σ_y^+ на растянутой стороне, чуть позднее будет достигнут предел текучести σ_y^- . Далее образовавшиеся зоны пластичности будут расширяться до тех пор, пока вся пластина не перейдет в пластическое состояние.

Введем величины h^+ и h^- , которые характеризуют расстояние от срединной плоскости до границы упругой зоны в растянутой и сжатой областях соответственно (рис. 1). На рис. 1 области пластины, которые находятся в пластическом состоянии выделены темным цветом.

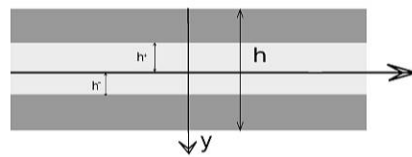


Рис. 1. Сечение пластины

Так как пределы текучести при растяжении и сжатии у пластины различны, то при развитии пластических зон имеет место смещение нейтральной плоскости (с расположенной в ней плоскостью Oxz) относительно срединной плоскости на некоторую величину c . На рис. 2 также представлены эпюры напряжений для четырех описанных выше стадий. Величину c находим из условия равновесия всех усилий, направленных вдоль оси абсцисс.

костью Oxz) относительно срединной плоскости на некоторую величину c . На рис. 2 также представлены эпюры напряжений для четырех описанных выше стадий. Величину c находим из условия равновесия всех усилий, направленных вдоль оси абсцисс.

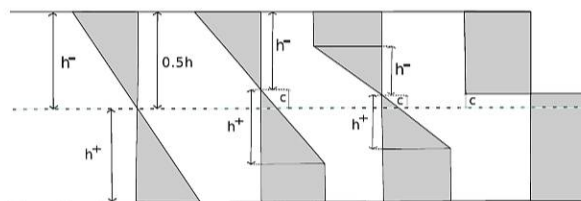


Рис. 2. Эпюры напряжений для четырех стадий деформирования

Сначала выведем уравнение, связывающее величину изгибающего момента с пределами текучести при наличии двух зон пластичности:

$$M = \int_{c-\frac{h}{2}}^{-h^-} -\sigma_y^- y dy + \int_{-h^-}^{h^+} \sigma_y^+ y^2/h^+ dy + \int_{h^+}^{\frac{h}{2}+c} \sigma_y^+ y dy. \quad (1)$$

Подставив $h^+ = h^- = 0$ в формулу (1), определим соотношение, связывающее величину момента с толщиной пластины, в тот момент времени, когда она полностью перешла в пластическое состояние

$$M = \frac{\sigma_y^+ h^2 d}{2(d+1)}. \quad (2)$$

Теперь рассмотрим влияние коррозии на пластину. Под действием коррозии толщина пластины за время t уменьшается с h_0 до $h = h_0 - at$. Используя формулу (2), выразим время достижения предельного состояния

$$t^* = \frac{h_0 - \sqrt{\frac{2M(d+1)}{\sigma_y^+ d}}}{a}. \quad (3)$$

Это и будет искомая долговечность пластины.

Следует отметить, что наличие поверхностных дефектов может привести к локализации коррозии [6–8]. В таком случае представленное решение перестает быть справедливым.

4. Пример расчетов. В качестве примера приведем расчет в пакете MATLAB [9] долговечности пластины, которая корродирует по лицевым поверхностям с постоянной скоростью $a = 1$ мм/год. Величину d будем рассматривать на промежутке от 1 до 1,5. Изгибающим моментам придадим различные значения: $M_1 = 6 \times 10^5$, $M_2 = 4,5 \times 10^5$, $M_3 = 3 \times 10^5$ Па·м³. На рис. 3 изображены графики зависимости долговечности t^* от величины d для разных моментов: M_1 (сплошная линия), M_2 (пунктирная линия), M_3 (штрихпунктирная линия). Очевидно, что с ростом величины момента долговечность сокращается. Анализ графика показывает, что при увеличении величины d увеличивается и долговечность. Следует отметить, что это наиболее заметно при малых моментах. Так, с ростом d время перехода в пластическое состояние при M_1 увеличивается на 6%, при M_2 – на 8%, а при M_3 – на 11%.

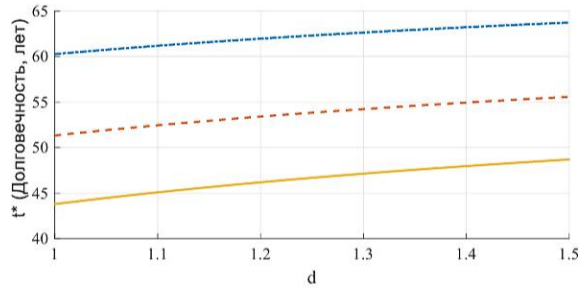


Рис. 3. Зависимости долговечности пластины от величины $d = \sigma_y^- / \sigma_y^+$ при различных изгибающих моментах

5. Заключение. В статье рассмотрена задача определения срока службы пластины из идеального упругопластического материала с SD-эффектом, которая корродирует с постоянной скоростью под действием внешней среды. В результате получена формула для вычисления времени перехода пластины в пластическое состояние.

Кроме того, было исследовано влияние SD-эффекта на долговечность. Также сделаны выводы о том, что с увеличением величины d увеличивается и долговечность пластины, что наиболее сильно заметно при малых моментах.

Литература

1. Пронина Ю. Г. Оценка устойчивости упругой трубы под давлением коррозионных сред // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2006. № 3. С. 55–63.
2. Пронина Ю. Г. Механохимическая коррозия полого цилиндра из идеального упруго-пластического материала под действием постоянного давления полого цилиндра из идеального упруго-пластического материала под действием постоянного давления // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2006. № 3. С. 121–130.

3. Пронина Ю. Г. Равномерная механохимическая коррозия полой сферы из идеального упругопластического материала под действием постоянного давления // Вестник Санкт-Петербургского университета. Математика. Механика. Астрономия. 2009. № 1. С. 113–122.
4. Седова О. С., Пронина Ю. Г. О выборе эквивалентного напряжения в задачах о механохимической коррозии сферических элементов // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2016. Т. 12. № 2. С. 33–44.
5. Павилайнен Г. В., Кропачева Н. Ю. Влияние пластической анизотропии на напряженно-деформированное состояние балок при изгибе // Перспективные материалы и технологии: монография: в 2 томах / под. ред. чл.-корр. Рубаника В.В. Витебск: УО «ВГТУ», 2019. Т. 2. С. 52–66.
6. Shuvalov G. M., Vakaeva A. B., Shamsutdinov D. A., Kostyrko S. A. The effect of nonlinear terms in boundary perturbation method on stress concentration near the nanopatterned bimaterial interface // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2020. Т. 16. № 2. С. 165–176.
7. Zhao S., Pronina Y. On the stress state of a pressurised pipe with an initial thickness variation, subjected to non-homogeneous internal corrosion // E3S Web of Conferences. 2019. Art. No 01013.
8. Stareva I. A., Pronina Y. G. The influence of the initial thickness deviation of a steel tube subjected to general corrosion under its own weight on its durability // AIP Conference Proceedings. 8th Polyakhov's Reading: Proceedings of the International Scientific Conference on Mechanics. 2018. Art. No 070034.
9. Колпак Е. П. Вычисления в Matlab. Учебное пособие. Казань: ООО «Бук», 2016. 173 с.

УДК 519.711.2

Рыбалкин А. А.

Математическое моделирование балансирующего робота на сферическом колесе

Рекомендовано к публикации доцентом Коровкиным М. В.

1. Введение. Балансирующий робот на сферическом колесе представляет большой интерес в робототехнике. За счет того, что он имеет единственную точку контакта с поверхностью, он способен легко перемещаться во всех направлениях. Поэтому он, по сравнению с обычным колесным роботом, имеет высокую подвижность и маневренность. Следующее преимущество – робота можно сделать высоким, при этом, чем он выше, тем более устойчив. И еще одно преимущество данной версии колесного робота – он может двигаться по наклонным и подвижным поверхностям. Заметим, в литературе нет описания математических моделей данной версии робота, а в открытых источниках можно найти краткую информацию лишь о трехколесной версии [1]. В данной работе рассматривается случай балансирующего робота с четырьмя электроприводами на омниколесах и приводится пример его математической модели.

2. Постановка задачи. Рассмотрим балансирующий робот, тело которого представляет собой сплошной цилиндр. К нему под углом 90° в плоскости основания подсоединены четыре электропривода, которые опущены на некоторый фиксированный угол. На оси каждого из электроприводов подсоединены омниколеса. Балансирующий робот в свою очередь находится на поверхности полого шара. Предполагается, что шар свободно перемещается на гладкой, ровной поверхности. Схематичный рисунок представлен на рис. 1. Требуется построить математическую модель данного робота.

3. Кинематика. Для расчета кинематики системы были введены системы координат, показанные на рис. 1. Инерциальная система координат обозначена $Oxyz$, система координат центра шара – $Sx_1y_1z_1$. Переход от системы координат $Oxyz$ к системе координат $Sx_1y_1z_1$ представляет собой параллельный перенос – матрица T_0 [2].

Рыбалкин Алексей Александрович – студент, Санкт-Петербургский государственный университет; e-mail:aleksej.rynalkin@gmail.com, тел.: +7(950)664-93-34

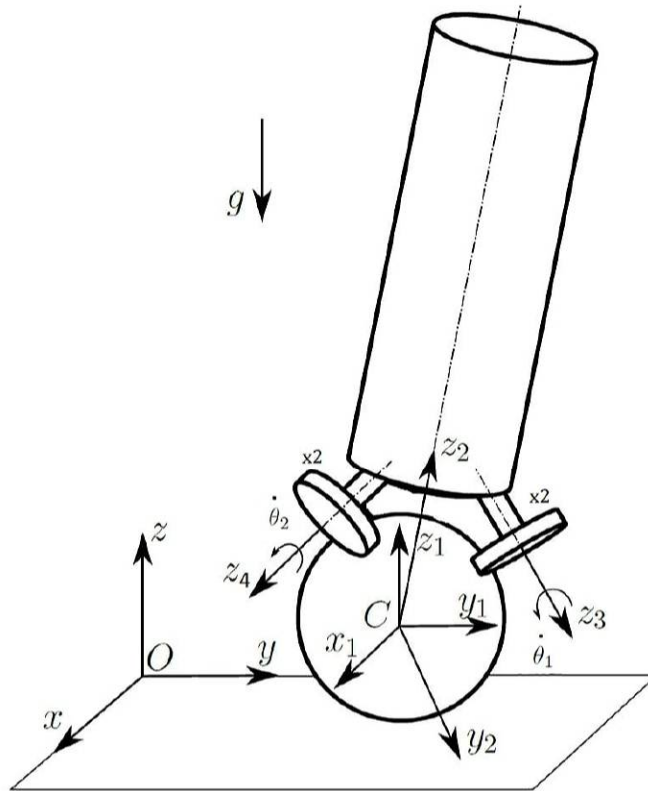


Рис. 1. Объект управления

Для описания вращения шара были использованы углы поворота вокруг осей. Таким образом, система координат $O_1x'_1y'_1z'_1$ (точки C и O_1 совпадают), связанная с шаром, получается из системы координат $Cx_1y_1z_1$ путем поворота осей Cx_1, Cy_1, Cz_1 на углы $\varphi_x, \varphi_y, \varphi_z$. Отсюда матрица перехода от системы координат $Cx_1y_1z_1$ к системе координат $O_1x'_1y'_1z'_1$:

$$T_1^t = R_x(\varphi_x)R_y(\varphi_y)R_z(\varphi_z),$$

где $R_x(\varphi_x)$ – матрица поворота вокруг оси Cx_1 на угол φ_x , $R_y(\varphi_y)$ – матрица поворота вокруг оси Cy_1 на угол φ_y , $R_z(\varphi_z)$ – матрица поворота вокруг оси Cz_1 на угол φ_z .

Аналогично можно описать вращение тела балансирующего робота в пространстве. Система координат $O_2x_2y_2z_2$ (точки C и O_2 совпадают), связанная с телом балансирующего робота, получается из $Cx_1y_1z_1$ путем поворота осей Cx_1, Cy_1, Cz_1 на углы ψ_x, ψ_y, ψ_z . Отсюда матрица перехода от системы координат $Cx_1y_1z_1$ к системе координат $O_2x_2y_2z_2$:

$$T'_2 = R_x(\psi_x)R_y(\psi_y)R_z(\psi_z),$$

где $R_x(\psi_x)$ – матрица поворота вокруг оси Cx_1 на угол ψ_x , $R_y(\psi_y)$ – матрица поворота вокруг оси Cy_1 на угол ψ_y , $R_z(\psi_z)$ – матрица поворота вокруг оси Cz_1 на угол ψ_z .

Системы координат $O_3x_3y_3z_3, O_4x_4y_4z_4, O_5x_5y_5z_5, O_6x_6y_6z_6$ связаны с омниколесами. Для перехода от системы координат $O_2x_2y_2z_2$, связанной с телом балансирующего робота, к системе координат, связанной с омниколесом, требуется выполнить несколько элементарных переходов. Это параллельный перенос до задней части электропривода, поворот вокруг оси O_2x_2 или O_2y_2 , затем снова параллельный перенос до середины омниколеса и поворот на угол ротора электропривода.

Рассмотрим вычисление матрицы перехода для первого омниколеса:

$$T'_3 = T_{red}R_x(-\alpha_1)T_{edcw}R_z(\theta_1),$$

где T_{red} – параллельный перенос до задней части электропривода, $R_x(-\alpha_1)$ – поворот вокруг оси O_2x_2 на фиксированный угол $-\alpha_1$ (знак «минус» указывает на поворот по часовой стрелке), T_{edcw} – параллельный перенос до центра омниколеса, $R_z(\theta_1)$ – поворот вокруг оси ротора на его угол отклонения θ_1 . Вычисление матриц перехода для остальных омниколес проводится аналогично. Для вычисления матриц перехода от инерциальной системы координат к остальным подвижным системам координат необходимо перемножить полученные матрицы перехода:

$$T_1 = T_0T'_1, \quad T_2 = T_0T'_2, \quad T_3 = T_2T'_3,$$

$$T_4 = T_2T'_4, \quad T_5 = T_2T'_5, \quad T_6 = T_2T'_6.$$

Таким образом, для описания положения системы имеем следующий вектор обобщенных координат:

$$\mathbf{q} = (x, y, \varphi_x, \varphi_y, \varphi_z, \psi_x, \psi_y, \psi_z, \theta_1, \theta_2, \theta_3, \theta_4)^T,$$

где x, y – координаты центра шара; $\varphi_x, \varphi_y, \varphi_z$ – углы, описывающие вращение шара; ψ_x, ψ_y, ψ_z – углы, описывающие вращение тела; $\theta_1, \theta_2, \theta_3, \theta_4$ – углы поворота омниколеса вокруг оси двигателя.

4. Динамика. На систему наложено семь не голономных связей: четыре связи качения омниколес по шару, две связи для скорости центра шара, одна связь отсутствия верчения шара. Все уравнения можно получить из уравнений Эйлера и геометрических параметров балансирующего робота.

Таким образом, имеем:

- число обобщенных координат $m = 12$;
- число связей $s = 7$;
- число степеней свободы $n = 5$.

Для построения уравнений движения будет использоваться вектор псевдоскоростей

$$\dot{\pi} = (\dot{x}, \dot{y}, \dot{\psi}_x, \dot{\psi}_y, \dot{\psi}_z).$$

Для построения дифференциальных уравнений движения необходимо вычислить энергию ускорений для каждого звена [3]:

$$U = \sum_{i=1}^N \frac{1}{2} \text{tr} \left(\ddot{T}_i H_i \ddot{T}_i^T \right),$$

где $N = 6$ – количество звеньев, T_i – матрица перехода от инерциальной системы координат до системы координат i -го звена, H_i – матрица инерции i -го звена tr – след матрицы.

Так как все звенья системы имеют простейшую форму (полый шар, сплошной цилиндр), поэтому матрицы инерции имеют простой вид и здесь не рассматриваются.

В общем случае

$$U = U(q_1, \dots, q_m, \dot{q}_1, \dots, \dot{q}_m, \ddot{q}_1, \dots, \ddot{q}_m, t),$$

но с помощью уравнений связи ее можно свести к функции

$$U = U(q_1, \dots, q_m, \dot{\pi}_1, \dots, \dot{\pi}_n, \ddot{\pi}_1, \dots, \ddot{\pi}_n, t).$$

Теперь рассмотрим дифференциальные уравнения движения Апеля в псевдокоординатах:

$$\frac{\partial U}{\partial \ddot{\pi}} = \mathbf{Q}, \quad (1)$$

где \mathbf{Q} – обобщенные силы, соответствующие псевдокоординатам, которые можно разделить на две составляющие: силы от моментов приводов \mathbf{Q}_1 и от сил тяжести \mathbf{Q}_2 .

Преобразуем уравнение (1). Для этого введем радиус-вектор

$$\mathbf{f} = \mathbf{f}(\mathbf{q}, \dot{\pi}, \ddot{\pi}, t) = \frac{\partial U}{\partial \ddot{\pi}} - \mathbf{Q}_1 - \mathbf{Q}_2 = 0.$$

Тогда

$$\mathbf{f}(\mathbf{q}, \dot{\pi}, \ddot{\pi}, t) = A(\mathbf{q})\ddot{\pi} + \mathbf{b}(\mathbf{q}, \dot{\pi}) + \mathbf{c}(\mathbf{q}) - \mathbf{Q}_1 = 0,$$

где

$$\begin{aligned} A(\mathbf{q}) &= \frac{\partial \mathbf{f}}{\partial \ddot{\pi}}, \\ \mathbf{b}(\mathbf{q}, \dot{\pi}) &= \mathbf{f} - A(\mathbf{q})\ddot{\pi} - \mathbf{c}(\mathbf{q}) + \mathbf{Q}_1, \\ \mathbf{c}(\mathbf{q}) &= -\mathbf{Q}_2. \end{aligned}$$

Отсюда

$$\ddot{\pi} = A^{-1}(\mathbf{q})(\mathbf{Q}_1 - \mathbf{b}(\mathbf{q}, \dot{\pi}) - \mathbf{c}(\mathbf{q})). \quad (2)$$

Сделав замену в уравнении (2)

$$\omega = (\pi, \dot{\pi}) = (x, y, \psi_x, \psi_y, \psi_z, \dot{x}, \dot{y}, \dot{\psi}_x, \dot{\psi}_y, \dot{\psi}_z),$$

получим 10 нелинейных уравнений первого порядка, которые описывают поведение системы.

5. Заключение. В работе представлена кинематика и динамика балансирующего робота на сферическом колесе, выведены дифференциальные уравнения движения системы. В дальнейшем планируется провести линеаризацию данной системы в виде ЛТИ-объекта, а также построить управление для стабилизации данной системы в положении равновесия, и применить различные регуляторы для изучения управляемости данного робота.

Литература

1. История одного диплома или как сделать шаробота [Электронный ресурс]: <https://habr.com/ru/post/306456/> (дата обращения: 15.03.2021).
2. Бабаджанянц Л. К., Пупышев Ю. А., Пупышева Ю. Ю. Классическая механика. Учебное пособие. СПб.: Изд-во СПбГУ, 2013. 259 с.
3. Гантмахер Ф. Р. Лекции по аналитической механике. М.: Наука, 1966. 300 с.

УДК 517.929.2

Хоанг В. Н.

Дифференциально-разностное уравнение с распределенными параметрами на графе

Рекомендовано к публикации профессором Провоторовым В. В.

1. Введение. В работе рассматривается вопрос построения приближенного решения дифференциально-разностного уравнения с распределенными параметрами на графе.

2. Основные обозначения. Используются обозначения, принятые в работах [1, 2]. Ребра γ графа Γ имеют одинаковую длину и параметризованы отрезком $[0, 1]$; $\partial\Gamma$ – множества граничных узлов графа; Γ_0 – объединение всех ребер, не содержащих концевых точек; $L_2(\Gamma)$ – банахово пространство измеримых на Γ_0 функций, суммируемых с p -й степенью; $W_{\frac{1}{2}}(\Gamma)$ – пространство функций из $L_2(\Gamma)$, имеющих обобщенную производную первого порядка также из $L_2(\Gamma)$. Обозначим через $\Omega_a(\Gamma)$ множество непрерывных во всех внутренних узлах функций $u(x)$ из класса $W_{\frac{1}{2}}(\Gamma)$, удовлетворяющих краевому условию $u(x)|_{\partial\Gamma} = 0$ и соотношениям: $\sum_{\gamma \in R(\xi)} a(1)_\gamma \frac{du(1)_\gamma}{dx} = \sum_{\gamma \in r(\xi)} a(0)_\gamma \frac{du(0)_\gamma}{dx}$. Здесь $R(\xi)$ и $r(\xi)$ – множества ребер γ , соответственно ориентированных к узлу ξ и от узла ξ . Замыкание в норме $W_{\frac{1}{2}}(\Gamma)$ множества $\Omega_a(\Gamma)$ обозначим через $W_0^1(a, \Gamma)$.

3. Дифференциально-разностное уравнение. Рассмотрим дифференциально-разностное уравнение

$$\frac{1}{\tau}(u(k) - u(k-1)) + \Lambda u(k) = f(k), \quad (1)$$

с начальным и краевым условиями

$$u(0) = \varphi(x), \quad u(k)|_{x \in \partial\Gamma} = 0, \quad k = 1, \dots, M, \quad (2)$$

Хоанг Ван Нгуен – аспирант, Воронежский государственный университет;
e-mail: fadded9x@gmail.com, тел.: +7(952)544-27-74

где $\tau > 0$ – действительное число, $f(k) \in L_2(\Gamma)$, $\varphi(x) \in L_2(\Gamma)$, $\Lambda u(k) = -\frac{d}{dx} \left(a(x) \frac{du(k)}{dx} \right) + b(x)u(k)$. Коэффициенты $a(x), b(x)$ – фиксированные, измеримые и ограниченные на Γ_0 функции, суммируемые с квадратом: $0 < a_* \leq a(x) \leq a^*$, $|b(x)| \leq \beta$, $x \in \Gamma_0$.

Определение. Слабым решением уравнения (1) с условиями (2) называются функции $u(k) \in W_0^1(a, \Gamma)$, $k = 1, \dots, M$, удовлетворяющие интегральному тождеству

$$\begin{aligned} \frac{1}{\tau} \int_{\Gamma} [u(k) - u(k-1)] \eta(x) dx + \int_{\Gamma} \left(a(x) \frac{du(k)}{dx} \frac{d\eta(x)}{dx} + b(x)u(k)\eta(x) \right) dx = \\ = \int_{\Gamma} f(k)\eta(x) dx, \quad u(0) = \varphi(x) \end{aligned}$$

для любой функции $\eta(x) \in W_0^1(a, \Gamma)$.

Теорема [3]. Для любых $k_0 \geq 0$ и любых $\varphi(x) \in L_2(\Gamma)$ слабое решение $u(k)$ однозначным образом определено при $k_0 \leq k \leq M$, $k_0 < M < \infty$.

Множество собственных функций $\{\phi_i(x)\}$ краевой задачи в пространстве $W_0^1(a, \Gamma) : \Lambda \phi = \lambda \phi$, образует базис в $W_0^1(a, \Gamma)$ и $L_2(\Gamma)$ [4]. Обозначим через $\{\lambda_i\}$, $i = 1, 2, \dots$, – множество собственных значений. Решение краевой задачи (1), (2) представляется в виде

$$u(k) = \sum_{i=1}^{\infty} C_i^k \phi_i(x).$$

Отсюда $\Lambda u(k) = \sum_{i=1}^{\infty} C_i^k \Lambda \phi_i(x) = \sum_{i=1}^{\infty} C_i^k \lambda_i \phi_i(x)$. Для функций $f(x)$, $\varphi(x)$ имеют место разложения по базису в $L_2(\Gamma)$:

$$f(k) = \sum_{i=1}^{\infty} f_i^k \phi_i(x), \quad f_i^k = (f(k), \phi_i(x)), \quad k = 1, \dots, M,$$

$$\varphi(x) = \sum_{i=1}^{\infty} \varphi_i \phi_i(x), \quad \varphi_i = (\varphi(x), \phi_i(x)).$$

Из (1) при $k = 1$ следует $u(1) + \tau \Lambda u(1) = u(0) + \tau f(1)$. Откуда имеем $C_i^1 = \frac{\varphi_i + \tau f_i^1}{1 + \tau \lambda_i}$, а значит

$$u(1) = \sum_{i=1}^{\infty} \frac{\varphi_i + \tau f_i^1}{1 + \tau \lambda_i} \phi_i(x). \quad (3)$$

Из (1) при $k = 2$ следует $u(2) + \tau \Lambda u(2) = u(0) + \tau f(2)$,

$$C_i^2 = \frac{\varphi_i + \tau f_i^1 + \tau (1 + \tau \lambda_i) f_i^2}{(1 + \tau \lambda_i)^2},$$

$$u(2) = \sum_{i=1}^{\infty} \frac{\varphi_i + \tau f_i^1 + \tau (1 + \tau \lambda_i) f_i^2}{(1 + \tau \lambda_i)^2} \phi_i(x) \quad (4)$$

и т. д.

При $k = M$ аналогично получим $C_i^M = \frac{\varphi_i + \tau \sum_{n=0}^{M-1} (1 + \tau \lambda_i)^n f_i^{n+1}}{(1 + \tau \lambda_i)^M}$,

$$u(M) = \sum_{i=1}^{\infty} \frac{\varphi_i + \tau \sum_{n=0}^{M-1} (1 + \tau \lambda_i)^n f_i^{n+1}}{(1 + \tau \lambda_i)^M} \phi_i(x). \quad (5)$$

4. Отыскание приближенного решения. Пусть заданы функции $\varphi(x) \in L_2(\Gamma)$, $f(k) \in L_2(\Gamma)$, действительные числа $\tau > 0, \varepsilon > 0$, натуральные числа $N > 0, M > 0$.

Укажем алгоритм отыскания приближенного решения:

1. Для $k = 1$ согласно (3) определяется приближенное решение

$$u^N(1) = \sum_{i=1}^N \frac{\varphi_i + \tau f_i^1}{1 + \tau \lambda_i} \phi_i(x).$$

2. Для $k = 2$ согласно (4) определяется приближенное решение

$$u^N(2) = \sum_{i=1}^N \frac{\varphi_i + \tau f_i^1 + \tau (1 + \tau \lambda_i) f_i^2}{(1 + \tau \lambda_i)^2} \phi_i(x)$$

и т. д.

3. Для $k = M$ согласно (5) определяется приближенное решение

$$u^N(M) = \sum_{i=1}^N \frac{\varphi_i + \tau \sum_{n=0}^{M-1} (1 + \tau \lambda_i)^n f_i^{n+1}}{(1 + \tau \lambda_i)^M} \phi_i(x).$$

4. Останов процесса построения приближенного решения $u^N(k)$ задачи (1), (2) происходит при выполнении неравенства

$$\sum_{k=1}^M \|u^N(k) - u^{N-1}(k)\|_{W_0^1(a, \Gamma)} < \varepsilon$$

при некотором заданном $\varepsilon > 0$. В противном случае число N увеличивается на единицу.

5. Пример. Пусть Γ – граф с тремя ребрами $\gamma_1, \gamma_2, \gamma_3$ и внутренним узлом ξ . Ребро γ_1 ориентировано к узлу ξ , ребра γ_2, γ_3 ориентированы от узла ξ . Все ребра параметризованы отрезком $[0, 1]$. Рассмотрим краевую задачу (1), (2) при $a(x) = 1, b(x) = f(k) = 0$,

$$\frac{1}{\tau} (u(k) - u(k-1)) + \Lambda u(k) = 0, \quad (6)$$

$$u(0) = \varphi(x), \quad u(k)|_{x \in \partial \Gamma} = 0, \quad k = 1, \dots, M. \quad (7)$$

Здесь $\Lambda u(k) = -\frac{d^2 u(k)}{dx^2}$. Начальная функция

$$\varphi(x) = \begin{cases} \sin \pi x + \pi x, & x \in \gamma_1, \\ \sin \pi x + \pi(1-x), & x \in \gamma_2, \\ \sin \pi x + \pi(1-x), & x \in \gamma_3. \end{cases}$$

Краевая задача

$$\Lambda \phi = \lambda \phi, \quad \phi|_{\partial \Gamma} = 0$$

имеет множество собственных значений $\{\lambda_i\}$ и соответствующие собственные функции $\{\phi_i(x)\}$, $i = 1, 2, \dots$. При $i = 2j$, $\{\lambda_i\}$ имеют кратность 2, $\lambda_{2j} = \pi^2 j^2$, $j = 1, 2, \dots$,

$$\phi_{2j,1}(x) = \begin{cases} \frac{1}{\pi j} \sin \pi j(x-1), & x \in \gamma_1, \\ \frac{1}{\pi j} \sin \pi jx, & x \in \gamma_2, \\ 0, & x \in \gamma_3, \end{cases}$$

$$\phi_{2j,2}(x) = \begin{cases} \frac{1}{\pi j} \sin \pi j(x-1), & x \in \gamma_1, \\ 0, & x \in \gamma_2, \\ \frac{1}{\pi j} \sin \pi jx, & x \in \gamma_3. \end{cases}$$

Разложим $\varphi(x)$ по базису $\phi_{2j,1}(x), \phi_{2j,2}(x), j = 1, 2, \dots$. Коэффициенты Фурье функции $\varphi(x)$

$$\varphi_{2j,1} = (\varphi(x), \phi_{2j,1}(x)) = \int_{\Gamma} \varphi(x) \phi_{2j,1}(x) dx = 0,$$

$$\varphi_{2j,2} = (\varphi(x), \phi_{2j,2}(x)) = \int_{\Gamma} \varphi(x) \phi_{2j,2}(x) dx = 0.$$

При $i = 2j - 1, \lambda_{2j-1} = \frac{\pi^2(2j-1)^2}{4}, j = 1, 2, \dots,$

$$\phi_{2j-1}(x) = \begin{cases} \cos \frac{\pi}{2} (2j-1)(x-1), & x \in \gamma_1, \\ \cos \frac{\pi}{2} (2j-1)x, & x \in \gamma_2, \\ \cos \frac{\pi}{2} (2j-1)x, & x \in \gamma_3. \end{cases}$$

Разложим $\varphi(x)$ по базису $\phi_{2j-1}(x), j = 1, 2, \dots$. Коэффициенты Фурье функции $\varphi(x)$

$$\varphi_{2j-1} = \int_{\Gamma} \varphi(x) \phi_{2j-1}(x) dx = \frac{8(2j-1)^2 + 16}{4\pi(2j-1)^2 - \pi(2j-1)^4}.$$

Расположим собственные значения в порядке возрастания, причем с учетом кратности и перенумеруем их. Получим множество

собственных значений $\{\lambda_j\}, j = 1, 2, \dots, \lambda_1 \leq \lambda_2 \leq \lambda_3 \leq \dots$. Соответственно перенумеруем собственные функции, получим множество $\{\phi_j(x)\}, j = 1, 2, \dots$. Приближенное решение краевой задачи (6), (7) представляется в виде

$$u^N(k) = \sum_{j=1}^N \frac{\varphi_j}{(1+\tau\lambda_j)^k} \phi_j(x).$$

6. Заключение. В работе представлен метод аппроксимации дифференциальной системы параболического типа с распределенными параметрами на графе, использующий подход Ротэ (метод полудискретизации по t). Указан алгоритм определения слабого решения дифференциально-разностной системы и приведен пример. Разработанный алгоритм является основой для формирования программы для ЭВМ.

Литература

1. Хоанг В. Н. Дифференциально-разностная краевая задача для параболической системы с распределенными параметрами на графе // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 127–132.
2. Провоторов В. В. Синтез оптимального граничного управления параболической системы с запаздыванием и распределенными параметрами на графе // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2017. Т. 13. № 2. С. 209–224.
3. Provotorov V. V., Sergeev S. M., Hoang V. N. Countable stability of a weak solution of a parabolic differential-difference system with distributed parameters on the graph // Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes. 2020. Vol. 16. No 4. P. 402–414.
4. Volkova A. S., Provotorov V. V. Generalized solutions and generalized eigenfunctions of boundary-value problems on a geometric graph // Russian Mathematics. 2014. Vol. 58. No 3. P. 1–13.

УДК 517.958, 539.3

Шувалов Г. М., Костырко С. А.

Влияние поверхностной упругости на морфологическую устойчивость поверхности ультратонкого плёночного покрытия

1. Введение. Формирование устойчивого рельефа на поверхности твёрдых тел является важной проблемой в ряде прикладных задач. Например, возможность управлять ростом рельефа поверхности материала позволяет создавать гидрофобные покрытия. Также микрорельеф поверхности играет важную роль в определении белковой адгезии имплантатов, влияя таким образом на их биосовместимость.

Одним из способов формирования необходимого рельефа на поверхности различных изделий является нанесение ультратонких покрытий, которые из-за рассогласования кристаллических решёток материалов находятся в наряженном состоянии. Для релаксации напряжений атомы на поверхности плёнки могут перемещаться, формируя тем самым необходимый рельеф. В связи с этим, вопрос о том, какие факторы влияют на формирование устойчивого рельефа поверхности ультратонкого плёночного покрытия, представляет особый интерес.

Стоит отметить, что в большинстве теоретических работ, направленных на изучение процесса формирования поверхностных дефектов в твёрдых телах, пренебрегали влиянием поверхностной упругости, поскольку оно считалось относительно малым по сравнению с объёмным упругим поведением. Однако в наноразмерных материалах отношение площади поверхности к занимаемому материалом объёму увеличивается, в связи с чем возрастает влияние поверхностных и межфазных напряжений. В данной работе на основе модели Гёртина – Мёрдока будет учтено влияние поверхностных напряжений на морфологическую устойчивость поверхности ультратонкого плёночного покрытия.

Шувалов Глеб Михайлович – аспирант, Санкт-Петербургский государственный университет; e-mail: g.shuvalov@spbu.ru, тел.: +7(921)387-58-63

Костырко Сергей Алексеевич – доцент, Санкт-Петербургский государственный университет; e-mail: s.kostyrko@spbu.ru, тел.: +7(921)406-47-62

Работа выполнена при финансовой поддержке РФФИ, проект № 19-31-90024

2. Постановка задачи. В качестве модели ультратонкого плёночного покрытия толщины h_f , осаждённого на подложку толщины $h_s \gg h_f$ и находящегося в условиях плоской деформации, рассмотрим неоднородную упругую полуплоскость $B = B_1 \cup B_2$ комплексной переменной $z = x_1 + ix_2$ ($i^2 = -1$) с криволинейной свободной границей S_1 и прямолинейной межфазной границей S_2 (τ – время)

$$B_1 = \{z : 0 < x_2 < h_f + A(\tau) \cos(bx_1)\}, \quad B_2 = \{z : x_2 < 0\},$$

$$S_1 = \{z : z \equiv \zeta_1 = x_1 + i[h_f + A(\tau) \cos(bx_1)]\},$$

$$S_2 = \{z : z \equiv \zeta_2 = x_1\}, \quad b = \frac{2\pi}{a}, \quad A(0) = A_0, \quad A(\tau) \ll a \quad \forall \tau. \quad (1)$$

В соответствии с моделью поверхностной/межфазной упругости Гёртина–Мёрдока [1] свободная и межфазная границы представляются в виде пренебрежимо тонких слоёв, идеально связанных с объёмными фазами. Для описания их упругого поведения будем использовать упрощённые определяющие соотношения поверхностной упругости, предполагая, что поверхностная энергия зависит только от поверхностного напряжения и не зависит от градиента вектора перемещений

$$\sigma_{tt}^s(\zeta_j) = \gamma_j^0 + (\lambda_j^s + 2\mu_j^s)\varepsilon_{tt}^s(\zeta_j), \quad \zeta_j \in S_j, \quad j = \{1, 2\}, \quad (2)$$

где σ_{tt}^s – окружное поверхностное/межфазное напряжение, ε_{tt}^s – окружная поверхностная/межфазная деформация, γ_j^0 – остаточное поверхностное/межфазное напряжение, λ_j^s и μ_j^s – модули поверхностной/межфазной упругости, аналогичные константам Ламе.

Связь напряжений и деформаций в объёмных фазах B_1 и B_2 выражается законом Гука, который в случае плоской деформации имеет вид

$$\sigma_{nn}(z) = (\lambda_j + 2\mu_j)\varepsilon_{nn}(z) + \lambda_j\varepsilon_{tt}(z), \quad \sigma_{nt}(z) = 2\mu_j\varepsilon_{nt}(z), \quad (3)$$

$$\sigma_{tt}(z) = (\lambda_j + 2\mu_j)\varepsilon_{tt}(z) + \lambda_j\varepsilon_{nn}(z), \quad z \in B_j, \quad j = \{1, 2\},$$

где ε_{nn} , ε_{nt} , ε_{tt} и σ_{nn} , σ_{nt} , σ_{tt} – компоненты тензора объёмных деформаций и напряжений в локальной декартовой прямоугольной системе координат n, t (ось n перпендикулярна S_j), λ_j и μ_j – параметры Ламе объёмной фазы B_j .

Поскольку предполагается, что объёмные и поверхностные фазы когерентно связаны, на границах должны выполняться условия

$$\varepsilon_{tt}^s(\zeta_1) = \varepsilon_{tt}(\zeta_1), \quad \zeta_1 \in S_1, \quad (4)$$

$$\Delta u(\zeta_2) = u^+(\zeta_2) - u^-(\zeta_2) = 0, \quad \zeta_2 \in S_2,$$

где $u = u_1 + iu_2$ – вектор перемещений, u_1 и u_2 – перемещения вдоль декартовых осей (x_1, x_2) и $u^\pm = \lim_{z \rightarrow \zeta_2 \pm i0} u(z)$.

Условия механического равновесия на свободной и межфазной границах формулируются в терминах обобщённого закона Юнга – Лапласа [2]

$$\sigma(\zeta_1) = \kappa(x_1)\sigma_s(x_1) - i \frac{1}{h} \frac{d\sigma_s(x_1)}{dx_1}, \quad \zeta_1 \in S_1, \quad (5)$$

$$\Delta\sigma(\zeta_2) = \sigma^+(\zeta_2) - \sigma^-(\zeta_2) = i \frac{d\tau_s(x_1)}{dx_1}, \quad \zeta_2 \in S_2, \quad (6)$$

где $\sigma_s(x_1) = \sigma_{tt}^s(\zeta_1)$ – поверхностное напряжение на криволинейной границе S_1 , $\tau_s(x_1) = \sigma_{tt}^s(\zeta_2)$ – межфазное напряжение на прямолинейной границе S_2 , $\sigma = \sigma_{nn} + i\sigma_{nt}$ – вектор напряжений, $\sigma^\pm = \lim_{z \rightarrow \zeta_2 \pm i0} \sigma(z)$, h и κ – метрический коэффициент и кривизна искривлённой границы S_1 соответственно.

Напряжения $\sigma_{\alpha\beta}$, $\alpha, \beta = \{1, 2\}$, в глобальной декартовой прямоугольной системе координат (x_1, x_2) и угол поворота материальной частицы ω на бесконечности удовлетворяют условиям

$$\lim_{x_2 \rightarrow -\infty} \sigma_{22} = \lim_{x_2 \rightarrow -\infty} \sigma_{12} = \lim_{x_2 \rightarrow -\infty} \omega = 0, \quad \lim_{x_2 \rightarrow -\infty} \sigma_{11} = \sigma_2. \quad (7)$$

Для релаксации напряжений рельеф свободной поверхности плёнки может меняться. Требуется определить зависимость амплитуды рельефа A от времени τ при учёте поверхностной диффузии и упругого деформирования твёрдого тела с ультратонким плёночным покрытием переменной толщины.

3. Эволюционное уравнение. Будем считать, что самоорганизация искривлённой поверхности плёнки происходит под действием поверхностной диффузии, определяемой производной химического потенциала вдоль S_1 . Тогда, аналогично [3, 4], уравнение эволюции свободной поверхности S_1 запишем в виде

$$\frac{\partial g(x_1, \tau)}{\partial \tau} = K_s h(x_1, \tau) \frac{\partial^2}{\partial s^2} [U(\zeta, \tau) - \kappa(x_1, \tau) U_s(\zeta, \tau)], \quad (8)$$

$$g(x_1, \tau) = A(\tau) \cos(bx_1), \quad K_s = \frac{D_s C_s \Omega}{k_b T},$$

где U – плотность объёмной энергии деформации, U_s – удельная поверхностная энергия, Ω – атомный объём, D_s – коэффициент поверхностной самодиффузии, C_s – концентрация поверхностных дефектов, k_b – постоянная Больцмана, s – длина дуги вдоль кривой S_1 и T – абсолютная температура.

В данной работе поверхностная энергия U_s в эволюционном уравнении (8) не является постоянной и определяется аналогично энергии деформации объёмного упругого тела по формуле Клапейрона

$$U = \frac{1}{2} (\varepsilon_{nn} \sigma_{nn} + \varepsilon_{tt} \sigma_{tt} + 2\varepsilon_{nt} \sigma_{nt}), \quad U_s = \frac{1}{2} \varepsilon_{tt}^s \sigma_{tt}^s.$$

Таким образом, чтобы проинтегрировать уравнение (8), необходимо решить краевую задачу (1)–(7). Заметим, что уравнение (8) может быть использовано для описания изменения формы поверхности не только в случае процессов релаксации напряжений, но также при коррозионном растрескивании [5].

4. Численные результаты. Согласно подходу, описанному в [2], решение задачи (1)–(7) о совместной деформации плёнки и подложки строится с использованием принципа суперпозиции, комплексных потенциалов Гурса–Колосова и формул Мусхелишвили. В соответствии с методом возмущений, компоненты тензора напряжений и деформаций представляются в виде ряда по степеням малого параметра, в качестве которого выступает отношение максимальной амплитуды A к длине волны возмущения a . Подставляя полученные соотношения в выражение (8) и отбрасывая нелинейные относительно малого параметра слагаемые, получим обыкновенное дифференциальное уравнение. Решение этого уравнения даёт изменение амплитуды периодического рельефа поверхности ультратонкого плёночного покрытия со временем как функцию, зависящую от длины волны искривления a , продольного напряжения σ_2 , остаточных напряжений γ_j^0 и упругих констант λ_j , μ_j и λ_j^s , μ_j^s объёмных и поверхностных фаз соответственно.

В качестве примера рассмотрим алюминиевую плёнку на металлической подложке. Будем считать, что коэффициенты Пуассона материалов плёнки и подложки совпадают, т. е. $\nu_1 = \nu_2$. Такое предположение позволит проанализировать влияние подложки всего лишь через один параметр – относительную жёсткость $r = \mu_1/\mu_2$. Объёмные параметры Ламе для алюминия приняты равными $\lambda_2 = 58,17$ ГПа и $\mu_2 = 26,13$ ГПа. Для анализа влияния поверхностной упругости рассмотрим поверхностную/межфазную жёсткость $M_j = \lambda_j^s + 2\mu_j^s$. В ряде работ можно найти поверхностные параметры Ламе для алюминия с ориентацией кристаллической решётки [1,1,1], определённые методом молекулярного моделирования: им соответствует продольная поверхностная жёсткость $M_1 = 6$ Н/м. Поскольку поверхностные упругие постоянные зависят от ориентации кристаллической решётки, здесь также исследуется влияние этого параметра.

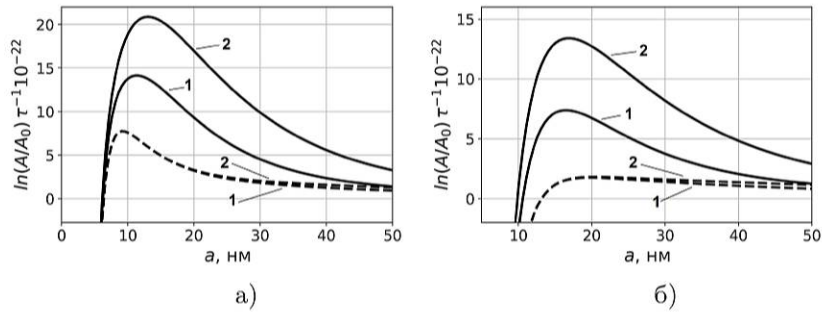


Рис. 1. Зависимость приведённой амплитуды от длины волны возмущения a а) при $M_1 = 6$ Н/м и б) при $M_1 = 30$ Н/м

На рис. 1 представлена зависимость относительной амплитуды от длины волны искривления a при продольных напряжениях в плёнке $\sigma_1 = r\sigma_2 = 50$ МПа, толщине покрытия $h_f = 5$ нм (сплошные линии) и $h_f = 15$ нм (пунктирные линии), относительной жёсткости $r = 3$ (кривые 1) и $r = 10$ (кривые 2), остаточном поверхностном напряжении $\gamma_1^0 = 1$ Н/м и поверхностной жёсткости $M_1 = 6$ Н/м (а) и $M_1 = 30$ Н/м (б). Межфазная жёсткость M_2 и остаточное напряжение γ_2^0 интерфейса приняты равными нулю, так как было обнаружено, что они практически не влияют на морфологическую устойчивость свободной поверхности плёнки. Критические значения a_{cr} длины волны искривления определяются из пересечения кривых с абсциссой. При $a < a_{cr}$ происходит сглаживание рельефа.

5. Заключение. Предложен теоретический подход для анализа морфологической устойчивости рельефа свободной поверхности ультратонкого плёночного покрытия с учётом ее упругих свойств на основе модели Гёртина–Мёрдока. В качестве примера рассмотрена алюминиевая плёнка на металлической подложке. Обнаружено, что критическое значение длины волны a_{cr} возрастает с увеличением поверхностной продольной жёсткости M_1 и относительной жёсткости r . В случае жёсткой относительно подложки плёнки критические значения длины волны возрастают с увеличением толщины плёночного покрытия и стремятся к значениям, полученным для поверхности твёрдого тела без покрытия [3].

Литература

1. Gurtin E., Murdoch A. Surface stress in solids // International Journal of Solids and Structures. 1978. Vol. 14. P. 431–440.
2. Kostyrko S. A., Altenbach H., Grekov M. A. Stress concentration in ultra-thin coating with undulated surface profile // Proceedings of the 7th International Conference on Coupled Problems in Science and Engineering. 2017. P. 1183–1192.
3. Костырко С. А., Шувалов Г. М. Влияние упругих свойств поверхности твердого тела на процесс ее реорганизации под действием напряжений // Процессы управления и устойчивость. 2018. Т. 5. № 1. С. 224–228.
4. Шувалов Г. М. Влияние упругих свойств гетероэпитаксиальной системы на устойчивость границы соединения двух материалов // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 133–137.
5. Седова О. С., Пронина Ю. Г. О выборе эквивалентного напряжения в задачах о механохимической коррозии сферических элементов // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2016. Т. 12. № 2. С. 33–44.

UDC 681.5

Schafberger J., Hagenmeyer V., Gröll L.

Modeling of a distillation column without measurement data

1. Introduction. Distillation columns are frequently used for the separation of liquid mixtures. It is standard to create the models for control design purposes by means of identification based on step experiments instead of physical modeling. However, the measurement data may not be available for various reasons. Firstly, the data cannot be made available for reasons of confidentiality. Secondly, obtaining suitable data may be expensive, e.g., if required sensors are missing. Thirdly, conducting experiments to acquire data is often impossible due to technical and safety reasons.

Our application consists in a distillation column and suffers from the aforementioned problem of lacking data. More precisely, we consider a vacuum distillation column with a structured packing for the separation of a mixture of N-Methyl-2-Pyrrolidone (NMP) and water. The separation process of this substance mixture is of particular importance since it arises during the manufacturing of batteries, which are applied in e.g., electrical vehicles or as energy storage in the electrical grid. However, at the same time, this process is challenging for control design since the separation occurs within a small section of the distillation column and thus causes large temperature gradients. As the currently implemented control concept does not achieve satisfying behavior, we developed a structural model of the plant from which an enhanced concept can be deduced. The advantage of our approach is that only geometry data are required and thus no measurement data are needed.

2. Process Description. In Figure 1 a simplified scheme of the considered distillation column is depicted. The plant includes further

Schafberger Jan – student, Karlsruhe Institute of Technology; e-mail: jan.schabberger@student.kit.edu, phone: +49(721)6082-92-00

Hagenmeyer Veit – professor, Karlsruhe Institute of Technology; e-mail: veit.hagenmeyer@kit.edu, phone: +49(721)6082-92-00

Gröll Lutz – professor, Karlsruhe Institute of Technology; e-mail: lutz.groell@kit.edu, phone: +49(721)6082-57-36

This work was supported by the Helmholtz Association under the Innovation Pool Project Energy Transition and Circular Economy.

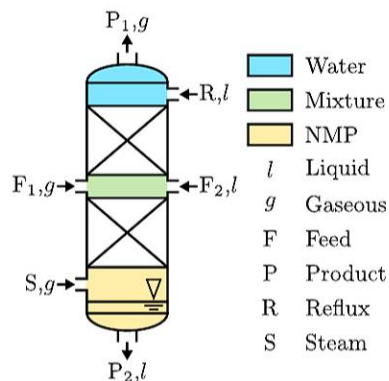


Figure 1. Scheme of the distillation column

subsystems which are not displayed for reasons of clarity. In the plant both mixture components are separated to a high degree of purity for reuse.

In the sump the liquid leaving the lower packing is collected. The liquid is conveyed continuously from the sump via a pump, whereby the flow P_2 is divided into two partial flows. The first flow leaves the plant as product flow while the second one is transported to an evaporator. The evaporated liquid S reenters the distillation column and streams into the lower packing. Between the packings, the NMP-water-mixture enters the distillation column in both gaseous F_1 and liquid state F_2 . Together with the liquid from the upper packing, the liquid feed flows into the lower packing, whereas the gaseous feed enters the upper one. The vapor P_1 leaving the upper packing streams towards a condenser. Afterwards, the water is collected in a tank from which the product flow and the reflux R are taken. The latter is fed back into the head. While the amount of vaporized liquid in the sump and the reflux can be controlled, the feed flows can not be manipulated.

3. Structural Model. To model the distillation column, the equilibrium stage approach is used. In doing so, the continuous packing is divided into discrete theoretical stages using the HETP (height equivalent to a theoretical plate) [1] which is a substance and load-dependent property of packings. At each theoretical stage, the liquid and the vapor are assumed to be in equilibrium. This approach has the advantage that only few parameters are needed. Moreover, it comes to use in the static, thermal dimensioning of distillation columns [2], and thus, we are able

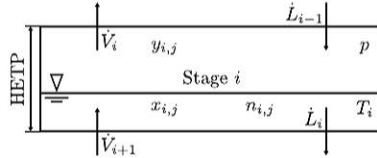


Figure 2. Theoretical stage

to use this data for the validation of the static model behavior.

In Figure 2 a single theoretical stage is depicted. Each stage i is assigned an amount of substance $n_{i,j}$, a molar fraction $x_{i,j}$ in the liquid and $y_{i,j}$ in the gaseous phase of both mixture components j . Other variables are the temperature T_i and the pressure p . The molar liquid \dot{L}_i and vapor flows \dot{V}_i between the stages are determined by an approach given in [3]. In order to include the influence of heat losses, the equation for determining the gas flow is modified. With the heat exchange coefficient α , the outer surface A_S of a stage and the ambient temperature T_0 the equation follows to

$$\dot{V}_i(t) = \frac{\dot{V}_{i+1}(t)h''_{i+1}(t) + \dot{L}_{i-1}(t)h'_{i-1}(t) - \dot{L}_i(t)h'_i(t) - \alpha A_S(T_i(t) - T_0)}{h''_i(t)}, \quad (1)$$

where h'_i is the molar enthalpy of the liquid and h''_i that of the gaseous phase. This extension was already successfully used for the static analysis of packed distillation columns [4]. Since the molar volumes of water and NMP are strongly different, a novel volume-based approach using Bernoulli's principle

$$\dot{L}_i(t) = \frac{k_1 A_D}{v_i(t)} \sqrt{\max \left\{ 0, \frac{2g(n_i(t)v_i(t) - V_0)}{k_2 A_D} \right\}}, \quad (2)$$

for the determination of the liquid flows is used instead of the original substance-based one. Thereby, v_i denotes the molar volume, A_D the column's cross sectional area and g the gravitational acceleration. The parameters k_1 and k_2 are used to adjust the overall holdup and the liquid level. With this approach the liquid holdup at a stage is divided into a static V_0 and a dynamic share which is illustrated in Figure 3. This allows to adjust the flow velocity independently from the overall liquid holdup at each stage. The existence of a static and a dynamic holdup is also supported by CFD simulations and experiments [5]. Furthermore, this approach offers the advantage of being very transparent and is therefore well suited for adjusting the model dynamics.

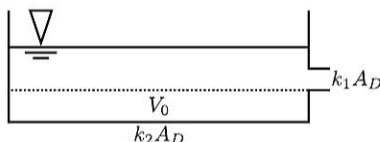


Figure 3. Illustration of the liquid flow model

In the following, further important model equations are presented where the focus is on the distillation column. The composition of the gaseous phase is calculated using Dalton's and Raoult's laws [1]

$$y_{i,j}(t) = \frac{x_{i,j}(t)p_j^*(T_i)}{p}, \quad (3)$$

where p_j^* is the vapor pressure. The temperature is obtained solving

$$p = x_{i,1}(t)p_1^*(T_i(t)) + x_{i,2}(t)p_2^*(T_i(t)) \quad (4)$$

with respect to the temperature. The vapor pressure p_j^* of the substances is calculated from

$$p_j^*(T_i) = \exp\left(A_j + \frac{B_j}{T_i(t)} + C_j \ln(T_i(t)) + D_j T_i(t)^{E_j}\right), \quad (5)$$

where A_j , B_j , C_j , D_j and E_j are substance dependent parameters. The material balance of both components follows from

$$\dot{n}_{i,j}(t) = \dot{V}_{i+1}(t)y_{i+1,j}(t) + \dot{L}_{i-1}(t)x_{i-1,j}(t) - \dot{V}_i(t)y_{i,j}(t) - \dot{L}_i(t)x_{i,j}(t). \quad (6)$$

The distillation column's model consists of several theoretical stages. Thus, a large system of coupled nonlinear equations is obtained for the distillation column alone.

Next, the influence of the parameters on the plant dynamics, starting from the equilibrium, is demonstrated. Let us at first discuss the results for the static holdup with the help of Figure 4. It can be seen that an increase of the static holdup hardly affects the response to an increase of the feed but slows down the rise in temperature for that of the heating steam. When one increases the dynamic holdup, the results shown in Figure 5 are obtained. The time until the change of the liquid feed affects the temperature increases strongly, whereas the response to an increased steam flow is less severe. Finally we considered the heat exchange coefficient. In Figure 6 it can be seen that the bigger the exchange parameter the shorter the time until the system comes to rest.

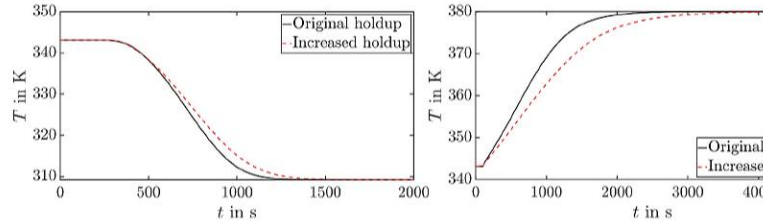


Figure 4. Temperature at a potential measuring point in the lower packing for an increase of the liquid feed's mass (left) and that of the heating steam (right) at $t = 100$ s with different static holdups

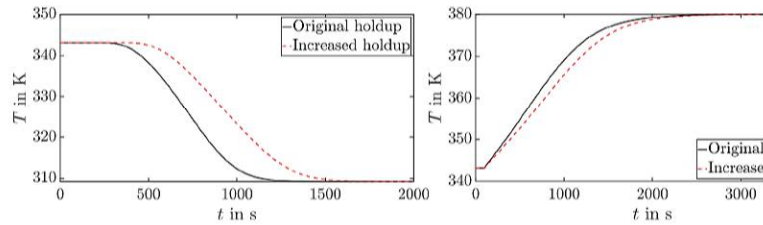


Figure 5. Temperature at a potential measuring point in the lower packing for an increase of the liquid feed's mass (left) and that of the heating steam (right) at $t = 100$ s with different dynamic holdups

Further, the system's qualitative behavior is affected. For small values of α the system shows almost integrator behavior, whereas for larger values PTn behavior is observed. It should be noted that the temperature at a stage is limited by the substance properties and thus the system comes to rest for the cases shown in red and blue, but not for the one in black. In the latter case, the temperature profile in the column moves slowly towards the head which can be deduced from the right graphic of Figure 6.

4. Model Simplifications. The nonlinear model of the distillation column alone is of about 100th order. In addition, there are further equations from the remaining system components. Thus, to simplify analysis and controller design, the system's complexity has to be reduced. As a first step, structural simplifications are made, neglecting the parts of the system that showed constant behavior in the operating range during simulation. Next, system components with significantly faster

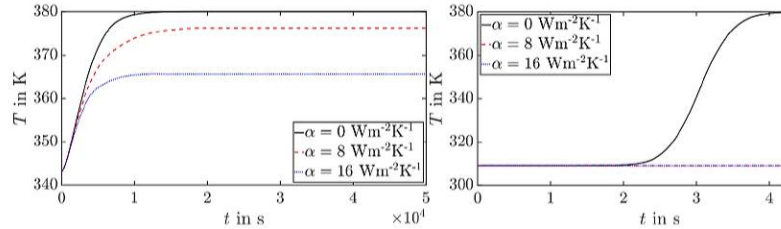


Figure 6. Temperature at a potential measuring point (left) and that of some stage above (right) in the lower packing for an increase of the heating steam’s mass for different heat exchange parameters

dynamics than the overall system’s behavior are neglected. After this step, the system consists of about eighty state variables. In the next step, a Taylor linearization is applied. Finally, an order reduction is performed using the method of Balanced Truncation. As a result, the system order is further reduced to twelve states.

In the following, the results of the simplification are presented in the closed control loop using the same controller parameters for all plants. Figure 7 depicts the system response to an increase of the liquid feed. The control error and the manipulated variable of the linearization as well as the reduced order approximation show good agreement with the nonlinear system’s behavior. In particular, the steady state values are effectively equivalent. Also the system behavior for an increase of the reflux’s mass is investigated. The results are depicted in Figure 8. A good match of the curves is observed even better than in the case of the liquid feed since the absolute change is smaller.

5. Outlook. We successfully designed a structural model which matches the known static behavior very well and which allows to adjust the step behavior with a small number of physical parameters. Thus, we see potential to adapt the model without measurements, using only practical experience from operators and qualitative observations. Thereafter, the system will be analyzed in detail. On this basis an improved control concept for the plant will be developed. The analysis can further be used to make suggestions for structural changes of the plant to improve the performance.

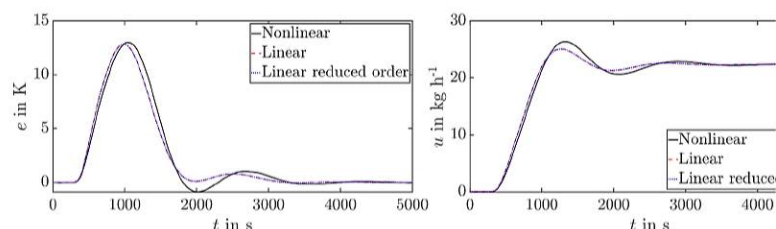


Figure 7. Control error and manipulated variable for an increase of the reflux's mass by five percent at $t = 100$ s

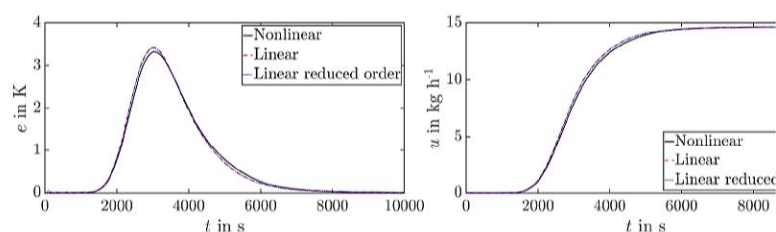
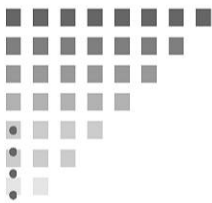


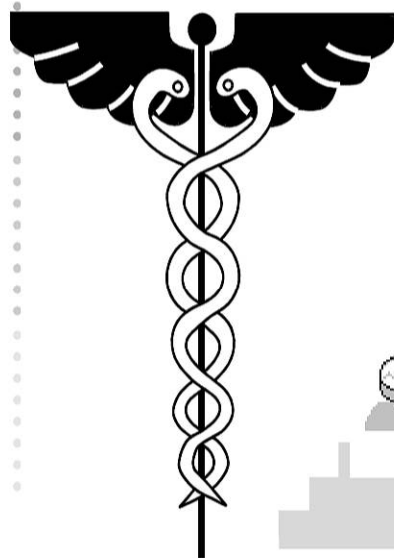
Figure 8. Control error and manipulated variable for an increase of the reflux mass by five percent at $t = 100$ s

References

1. Grassmann P., Widmer F., Sinn H. Introduction to thermal process engineering. Walter de Gruyter & Co, 1996 (in German).
2. Kister H. Z. Distillation design. McGraw-Hill, Inc., 1992. 710 p.
3. Albaz M., Karacan S., Cabbar Y., Hapoğlu H. Application of model predictive control and dynamic analysis to a pilot distillation column and experimental verification // Chemical Engineering Journal. 2002. Vol. 88. P. 163–174.
4. Peng J., Lextrait S., Edgar T. F., Eldridge R. B. A comparison of steady-state equilibrium and rate-based models for packed reactive distillation columns // Industrial & Engineering Chemistry. 2002. Vol. 41. P. 2735–2744.
5. Raynal L., Royon-Lebeaud A. A multi-scale approach for CFD calculations of gasliquid flow within large size column equipped with structured packing // Chemical Engineering Science. 2007. Vol. 62. P. 7196–7204.



*3. Математические
модели медико-
биологических систем*



УДК 004.8

Барташук А. Е.

Прогнозирование показателя Глисона рака предстательной железы с помощью искусственной нейронной сети

Рекомендовано к публикации доцентом Корниковым В. В.

1. Введение. На сегодняшний день рак предстательной железы (РПЖ) считается одной из самых серьезных медицинских проблем среди мужского населения. В России по заболеваемости он опережает рак кожи и рак желудка [1], а также занимает третье место среди онкологических заболеваний по смертности у мужчин [2].

Обычно на разных стадиях диагностики РПЖ используются анализ на простатический специфический антиген (PSA) и трансректальная биопсия простаты. Однако роль первого показателя довольно часто подвергается сомнению, что связано с высокой частотой ложноположительных результатов, а роль второго также ставится под вопрос из-за ряда возможных осложнений. Также существуют проблемы обработки и анализа больших массивов данных и нахождения закономерностей среди этих данных. Искусственная нейронная сеть, которая может быть использована для решения данных проблем, обучается таким образом, что дает некий необходимый прогноз на основе входных переменных.

В данной работе описан процесс получения и предобработки данных, их исследование, а также анализ применения генетического алгоритма для решения задачи прогнозирования РПЖ на основе показателя Глисона. Для исследования была выбрана база данных пациентов с диагностированным раком предстательной железы за период с 1990 г. по 2012 г. в городе Санкт-Петербурге. Поскольку данный ресурс не располагает достаточно полными данными и подробность информации разнится от пациента к пациенту, для решения задачи в первую очередь необходимо было произвести предобработку данных, отсеивая неподходящие случаи, а также отказаться от более подробного углубления в исследование (например, отказ от исполь-

Барташук Анастасия Евгеньевна – студент, Санкт-Петербургский государственный университет; e-mail: st064149@student.spbu.ru, тел.: +7(913)476-99-70

зования дополнительного входного параметра, такого как свободный PSA). Для избежания проблем с конфиденциальностью данных пациентов все имена и прочие данные, которые могли бы привести к идентификации личности, были опущены.

2. Подготовка данных. Для получения необходимого набора информации из упомянутой выше базы данных была осуществлена выборка по наличию у пациентов данных о росте и весе для дальнейшей возможности вычисления индекса массы тела (ИМТ). Также в качестве релевантной информации для исследования были выделены следующие признаки, которые получены из обработки исходной базы данных: индекс массы тела, возраст, объем простаты, PSA. На основе этих признаков была осуществлена дальнейшая выборка. Также для анализа работы нейронной сети был выделен показатель Глисона как следующий необходимый признак, и далее выборка осуществлялась уже по нему. Полученные в итоге данные были лишены идентификационной и лишней, не относящейся к исследованию, информации и собраны в новую простейшую базу данных.

Однако стоит заметить, что для более точных результатов, конечно, понадобится более подробная база данных.

3. Анализ входных данных. Прежде чем перейти к созданию нейронной сети, было проведено исследование признаков. Построив гистограммы распределения каждого из них, мы увидели, что нормально распределен только необходимый нам показатель Глисона, остальные же признаки распределены мультимодально (рис. 1).

Проведя несложные исследования, мы получили математические ожидания и значения дисперсий для каждого из признаков, исходя из которых можно сформулировать несколько локальных выводов:

1. Большая часть больных страдает ожирением (ИМТ от 30) или избыточным весом (ИМТ от 25).
2. Самый рискованный возраст составляет 64–65 лет, так как насчитывает больше всего случаев диагностики рака предстательной железы. Также можно заметить, что в зоне риска находятся мужчины после 50, следовательно, начиная с этого возраста, им рекомендовано периодическое посещение специалиста с целью раннего выявления болезни.
3. Чаще всего у людей встречается показатель Глисона, соответ-

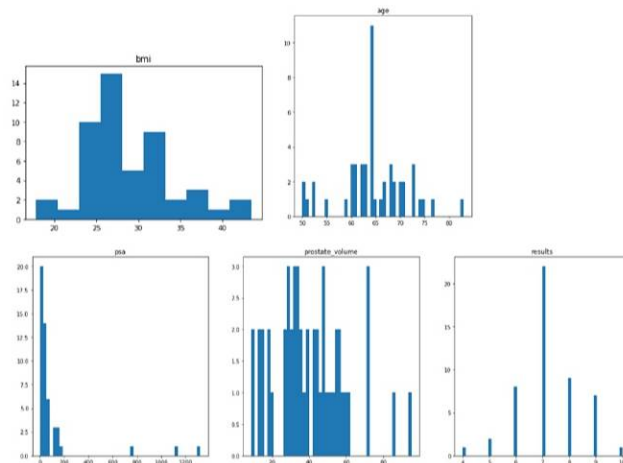


Рис. 1. Гистограммы признаков

ствующий 7 баллам. Опухоли простаты с таким значением считаются агрессивными опухолями средней степени. В подобных случаях рекомендуется незамедлительно начать лечение. Если опухоль локально ограничена, то лучшее решение – проведение операции полного (радикального) удаления простаты [4].

4. Дисперсии PSA и объема простаты велики, а следовательно, данные признаки имеют больший разброс.

После этого с помощью корреляционного анализа была определена теснота связи между исследуемыми признаками, в особенности, зависимость показателя Глисона от остальных признаков. Результаты представлены на рис. 2 в графическом виде.

Как можно наблюдать, признаки не скоррелированы, линейной зависимости нет.

Убедимся в этом, представив результаты корреляционного анализа в текстовом виде, т. е. коэффициентами корреляции. В нашем случае сильной связи между какими-либо признаками нет (рис. 3). Есть средняя между PSA и объемом простаты.

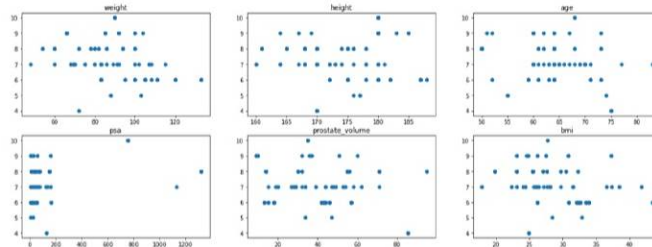


Рис. 2. Диаграммы разброса

	bmi	age	psa	prostate_volume	res
bmi	1.000000	-0.102308	-0.092395	-0.075100	-0.219818
age	-0.102308	1.000000	0.055071	0.127148	-0.206229
psa	-0.092395	0.055071	1.000000	0.395396	0.177251
prostate_volume	-0.075100	0.127148	0.395396	1.000000	-0.102375
res	-0.219818	-0.206229	0.177251	-0.102375	1.000000

Рис. 3. Таблица коэффициентов корреляции

Таким образом, мы удостоверились, что связь между исследуемыми параметрами в основном слабая, следовательно, прямой связи между признаками нет, поэтому теперь можно приступить к поиску нелинейных зависимостей.

4. Обзор генетического алгоритма. Обучение нейросети методом генетического алгоритма основано на идее естественного отбора в живой природе. В данной модели роль генов играют наборы чисел, весовые коэффициенты нейронной сети. Мутация происходит путем случайного изменения весового коэффициента на случайную величину. Диапазон и вероятность мутации заданы начальными значениями, но в ходе эволюции также могут мутировать. Скрещивание происходит путем случайного смешивания нейронов двух нейросетей. Полученная таким образом новая нейросеть занимает место в популяции. Задачей генетического алгоритма является минимизация целевой функции. В качестве целевой функции используется средняя ошибка нейронной сети [3].

Для анализа работы алгоритма и выявления лучшей структуры

были выбраны модели, представленные на рис. 4. Количество эпох: 500.

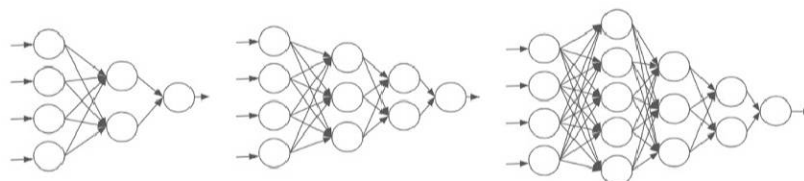


Рис. 4. Схемы выбранных моделей нейронных сетей

Удостоверимся в качестве работы алгоритма на примере с набором точных данных, так как медицинские данные, с которыми мы работаем, относятся к неточным. Проверку будем проводить рассмотрением так называемой проблемы XOR (исключающего ИЛИ).

Проведя необходимые вычисления, мы получили, что лучше всего себя показывает модель с двумя скрытыми слоями: сходимость алгоритма к локальному минимуму наступила за около 150 эпох и ошибка составила менее чем 10^{-12} . Алгоритм ведет себя достаточно адекватно, поэтому применим его для наших неточных данных. Для тестировочной выборки средние ошибки составили: 11,52%; 9,17% и 11,8% соответственно. По данным результатам можно заметить, что лучшего всего работает модель с четырьмя слоями, которая и будет использоваться в дальнейшем.

5. Решение. Для более точной интерпретации дальнейших результатов и построения более качественных моделей был использован генетический алгоритм с четырьмя слоями, описанный выше. Реализация алгоритма осуществлена на языке Python 3 с применением библиотек `numpy`, `matplotlib.pyplot`, `random`, `math` и `abc`. Для отображения результатов использован бесплатный облачный сервис Google Colaboratory, в котором и происходила разработка.

В качестве обучающей выборки были взяты 150 пациентов с подходящим набором данных. Входные данные были нормализованы на промежутке $[0,1]$ с использованием минимаксной функции.

Результаты для обучающей выборки до и после обучения представлены на рис. 5, первичные веса выбраны с использованием функции `rnd.random()`.

ПМТ	Возраст	PSA	Объем простаты	Выход желаемый	Выход реальный до обучения	Выход реальный после обучения
27.14	60	124	71	7	9.9999927	7.0003050
24.42	64	1129	53	7	9.9999927	7.0000741
31.56	63	39.1	34	7	9.9999927	7.0004258
26.2	63	8.2	13	6	9.9999927	6.8012323

Рис. 5. Результаты для обучающей выборки до и после обучения

Результаты для тестировочной выборки представлены на рис. 6.

ПМТ	Возраст	PSA	Объем простаты	Выход желаемый	Выход реальный
27.79	54	7.5	46	7	7.135210461
33.56	60	159	54	6	6.997292955
28.37	60	331.7	65	8	7.460873878
24.5	66	178.4	35	7	7.013821505

Рис. 6. Результаты для тестировочной выборки

Значения итоговых весов для слоев составляют $[0,2599687; 0,4848484; 0,1319265; 0,3023255]$, $[0,7689616; 0,4587972; 0,5936147]$ и $[0,6529135; 0,4905528]$ соответственно.

6. Заключение. В данной работе описаны подход к подготовке и исследованию данных медицинского датасета, содержащего информацию о пациентах с подтвержденным раком предстательной железы в Санкт-Петербурге в период с 1990 г. по 2012 г., а также результаты обучения и применения генетического алгоритма ИНС к нашему датасету. По результатам работы нейронной сети можно сказать, что она отлично справляется с точными данными (ошибка составляет менее 10^{-12}). Однако при работе с медицинскими данными ошибка возрастает до 9,2%, что, на самом деле, не самый лучший результат, который можно было ожидать. Несмотря на это, позволительно говорить о достаточно успешном исследовании: мы выяснили, что

факторы влияющие на появление и развитие рака предстательной железы слабосвязны и нелинейно зависимы. Также были получены веса для выбранной модели нейронной сети, а следовательно, можно продолжить исследование, отталкиваясь на их основе и на основе дальнейшего анализа информации, полученной в ходе статистического анализа входных параметров.

Литература

1. Рак у мужчин: наиболее распространенные типы, симптомы // Euroonco.ru URL:<https://yandex.ru/turbo/euroonco.ru/s/oncology/onkologicheskie-zabolevaniya-u-muzhchin> (дата обращения: 15.12.20).
2. Статистика: онкозаболевания и смертность в России и мире. Пункт: у мужчин // OncoVed.ru URL:<https://oncoved.ru/common/statistika-onkozabolevaniya-i-smertnost-v-rossii-i-mire#u-muzhchin> (дата обращения: 15.12.20).
3. Шуравин А. П. Исследование обучаемости нейронной сети на примере предсказания числовых рядов // Научные достижения и открытия 2017: сборник статей III Международного научно-практического конкурса. Пенза: Наука и Просвещение, 2017. С. 31–36.
4. Сумма Глисона 7 (3+4 или 4+3) // Raka-prostati.net URL:<http://www.raka-prostati.net/summa-glisona-7.html> (дата обращения: 21.03.21).

УДК 004.8

Белавин С. А.

Использование методов кластерного анализа для диагностирования заболеваний мозга

Рекомендовано к публикации профессором Дегтяревым А. Б.

1. Введение. Сложность задач, возникающих в области нейрофизиологии, а также объем данных, которые используются для диагностики и лечения пациентов, неуклонно растут. В связи с этим разработка эффективных методов обработки этих данных становится наиболее приоритетной. В настоящее время медицинские исследования и, в частности, нейрофизиология, являются обширной областью для применения методов искусственного интеллекта.

По оценкам, в 2018 году во всем мире насчитывалось около 50 миллионов человек, страдающих деменцией. К 2030 году это число по прогнозам увеличится до 82 миллионов, а уже к 2050 году достигнет отметки 152 миллиона человека, что почти в 3 раза больше, чем на данный момент [1]. Болезнь Альцгеймера – это тип деменции, составляющий около 60–80% всех случаев, который вызывает проблемы с памятью, мышлением и поведением. Повреждение мозга начинается задолго до того, как появляются первые когнитивные проблемы [2]. Для диагностики болезни Альцгеймера требуется комплексное обследование: изучается семейная медицинская история, применяются различные методы нейровизуализации (например, МРТ), проводятся нейрофизиологические исследования (например, ЭЭГ), когнитивные тесты на оценку памяти и мышления, анализы крови [3].

Ранняя диагностика болезни Альцгеймера может позволить назначить своевременное лечение для замедления прогрессирования заболевания. Так как на практике приходится вручную анализировать полученные результаты обследований, возможны ошибки, связанные с человеческим фактором. Кроме того, не всегда удастся поставить точный диагноз. Учитывая большое количество источников данных во время обследования, необходимо знать, на какие показате-

Белавин Сергей Андреевич – студент, Санкт-Петербургский государственный университет; e-mail: sa.belavin15@gmail.com, тел.: +7(911)207-10-09

тели стоит обращать внимание и как их лучше совмещать.

В данной работе рассматривается применение методов кластерного анализа для обработки данных, полученных при обследовании пациентов с помощью теста Векслера, магнитно-резонансной томографии и анализов крови. Конечной целью является создание инструментария, который облегчит врачам диагностирование заболеваний мозга.

2. Исходные данные. Для проведения исследования Санкт-Петербургским НИПНИ имени В. М. Бехтерева были предоставлены анонимизированные данные обследований пациентов, включающие данные теста Векслера и анализы крови (227 человек), а также данные МРТ обследований головного мозга (136 человек), предварительно обработанные программным пакетом FreeSurfer [4], который позволяет получить характеристики различных долей мозга. Среди пациентов были страдающие некоторыми заболеваниями головного мозга, а также контрольная группа.

3. Постановка задачи. Целью данной работы является поиск возможного разделения пациентов на группы, основываясь на результатах теста Векслера, данных МРТ обследования, а также проверка статистических различий между полученными группами, основываясь на анализах крови. Оценка соответствия разбиения реальным представлениям о пациентах будет проведена нейрофизиологами.

4. Методы. В данном разделе описаны методы, выбранные для анализа данных.

4.1. Понижение размерности. В связи с тем, что количество пациентов довольно мало по сравнению с количеством признаков, для увеличения точности было принято решение использовать понижение размерности признакового пространства. Для этих целей выбран метод главных компонент (РСА), реализованный в библиотеке Scikit-learn [5]. Количество главных компонент выбиралось так, чтобы остаточная дисперсия составляла не менее 85%. Кроме того, в связи с большим количеством данных, анализ МРТ обследований проводился отдельно для каждой доли мозга.

4.2. Кластеризация. Для разбиения на кластеры была выбрана иерархическая кластеризация с расстоянием Уорда. Данный метод позволяет объединять наиболее близкие кластеры, при этом удобно визуализируя их с помощью деревьев (дендрограмм), а также дает возможность ограничить конечное количество кластеров. Использовалась реализация из библиотеки SciPy [6].

4.3. Статистическая значимость. С помощью критерия Данна [7], который учитывает поправку на множественные сравнения, была определена статистическая значимость различий показателей крови в полученных кластерах.

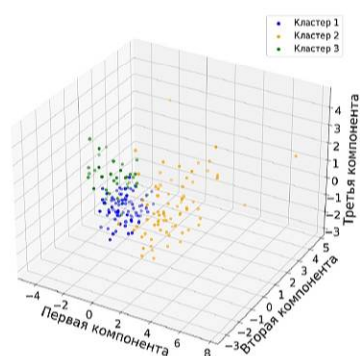


Рис. 1. Разбиение на кластеры по тесту Векслера

5. Результаты. При кластеризации данных теста Векслера были выделены три кластера (рис. 1). Первый представляет смесь различных групп пациентов, а в двух других преимущественно преобладают по представлению врачей либо контрольная группа, либо пациенты с болезнью Альцгеймера (рис. 2). Кроме того, второй и третий кластеры статистически различаются на уровне значимости 0,05 по показателям крови: общий белок и СОД ед./мл. Однако полученное разбиение не показало хорошо различимых групп.

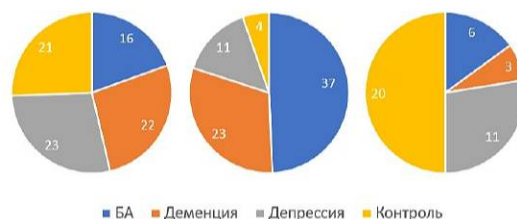


Рис. 2. Распределение пациентов по кластерам (тест Векслера)

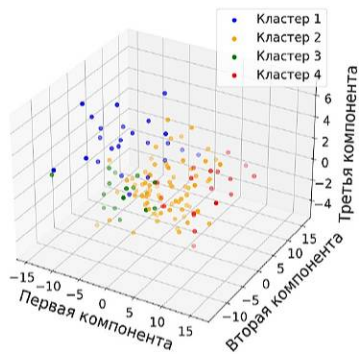


Рис. 3. Разбиение на кластеры по теменной и височной долям

ты с болезнью Альцгеймера, а в четвертом преобладали пациенты из контрольной группы и страдающие депрессией и практически отсутствовали с болезнью Альцгеймера, в двух других кластерах не было заметного преобладания определенной группы (рис. 4). Также первый и четвертый кластер статистически различались на уровне значимости 0,05 по показателям крови: СОД ед./мл и СОД ед./г Нб.

При кластеризации данных по долям мозга в основном получался один большой кластер с несколькими выбросами. Однако при рассмотрении некоторых долей удалось получить потенциально полезный результат. Так, наилучшее разбиение было получено при рассмотрении височной и теменной долей одновременно. Были выделены четыре кластера (рис. 3). В первом по представлению врачей не было пациентов из контрольной группы и преобладали пациенты

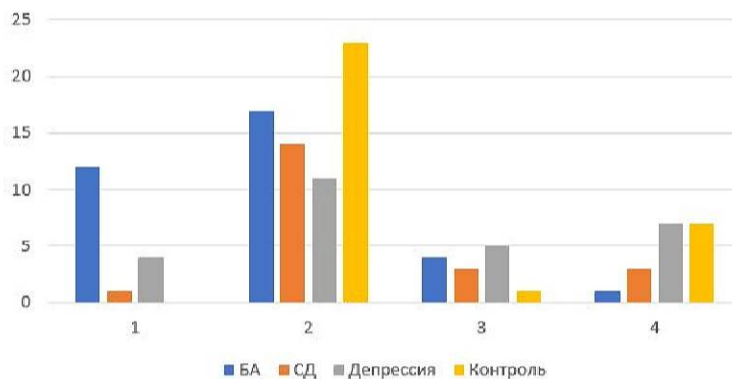


Рис. 4. Распределение пациентов по кластерам (теменная и височная доли)

6. Заключение. Полученные результаты свидетельствуют о возможности применения методов кластерного анализа для решения задачи диагностирования заболеваний мозга. Кроме того, специалистами из НИПНИ им. Бехтерева был отмечен высокий потенциал продолжения исследования.

В дальнейшем планируется добавить к анализу данные электроэнцефалографии пациентов, а также найти наиболее важные показатели для построения системы помощи в диагностике заболеваний мозга.

Литература

1. Patterson C. World Alzheimer report 2018. London: Alzheimer's Disease International, 2018. 48 p.
2. What is Alzheimer's disease? [Электронный ресурс]: URL:<https://www.alz.org/alzheimers-dementia/what-is-alzheimers> (дата обращения: 21.03.2021).
3. Alzheimer's and dementia in Russia [Электронный ресурс]: URL:<https://www.alz.org/ru/dementia-alzheimers-russia.asp> (дата обращения: 21.03.2021).
4. FreeSurfer – An open source software suite for processing and analyzing (human) brain MRI images [Электронный ресурс]: URL:<http://surfer.nmr.mgh.harvard.edu/> (дата обращения: 21.03.2021).
5. Decomposing signals in components [Электронный ресурс]: URL:<https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html> (дата обращения: 21.03.2021).
6. Hierarchical clustering [Электронный ресурс]: URL:<https://docs.scipy.org/doc/scipy/reference/generated/scipy.cluster.hierarchy.linkage.html> (дата обращения: 21.03.2021).
7. Гланц С. Медико-биологическая статистика / пер. с англ. М.: Практика, 1998. 459 с.

УДК 519.23

Бучина Д. А., Гончарова А. Б.

Обработка данных медицинских исследований с использованием псевдорандомизации (PSM)

1. Введение. Согласно информации, предоставляемой ВОЗ, ежегодно в мире регистрируется от 800 тыс. до 1 млн новых пациентов с диагнозом рак молочной железы. Смертность от данного вида онкологического заболевания среди женщин занимает второе место после болезней системы кровообращения. Однако при обнаружении заболевания на первой стадии существует 98% вероятность полного избавления от онкологии. Несмотря на существование современных и информативных методов диагностики, таких как маммография, УЗИ, МРТ, термография, биопсия и т. д., не всегда удается оперативно обнаружить злокачественные новообразования. В связи с чем особое место занимает проблема диагностики рака молочных желез на самых ранних стадиях.

При подозрении на онкологию диагностика часто проводится с помощью КУЗИ и МРТ. Поскольку оба метода подразумевают введение контрастных препаратов, на практике к пациенту применяется лишь один из них.

Целью настоящей работы является изучение метода псевдорандомизации и его применения для оценки эффективности методов КУЗИ и МРТ при диагностике рака молочных желез.

В настоящей работе рассматриваем базу с диагностическими данными женщин, обратившихся в 2017 году с жалобами на боль или уплотнения в молочных железах [1, 2]. В базе данных содержится 453 диагностических записи: 277 получены с помощью метода КУЗИ и 176 – с помощью метода МРТ.

2. Псевдорандомизация (PSM). Впервые понятие propensity score matching было введено Рубином и Розенбаумом в 1983 году [3]. Однако широкое распространение метод получает лишь после 2000 г.

Бучина Дарья Андреевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st086271@student.spbu.ru, тел.: +7(999)714-88-79

Гончарова Анастасия Борисовна – старший преподаватель, Санкт-Петербургский государственный университет; e-mail: a.goncharova@spbu.ru тел.: +7(812)428-47-29

и в данный момент наиболее успешно используется в области доказательной медицины, оперирующей результатами клинических исследований.

PSM – это статистический метод, позволяющий определить идентичных пациентов из двух разных групп и оценить влияние возможных действующих факторов. При использовании метода рассчитываем вероятность попадания каждого наблюдения в основную или контрольную группу на основании набора его характеристик [4, 6]. Благодаря полученным вероятностям можно подобрать группы сопоставления таким образом, чтобы значения соответствия по переменным были сбалансированы между ними. Так будет скомпенсировано неравномерное распределение характеристик, присущее обсервационным исследованиям, и учтена возможная ошибка при оценке различий между группами.

Для propensity score matching не существует общепринятых рекомендаций по выбору модели, также открытым остается вопрос о количестве и природе включаемых в модель характеристик, методах балансировки и расчета воздействия.

3. Алгоритм PSM. Опишем алгоритм псевдорандомизации для решения поставленной задачи.

1. *Построение модели.* На данном этапе проводим отбор переменных, кодирующих характеристики, учитывая теоретические представления о связи этих характеристик как с фактором, так и с исходом исследования. В рамках поставленной задачи отбираются следующие характеристики: id пациента, возраст, менструальный статус, размер новообразования, BIRADS, количество очагов, гистологическое заключение и молекулярный подтип. Переменной фактора принимаем тип метода диагностики, а переменной исхода – гистологическое заключение. Также нормируем каждый столбец со значениями характеристик по максимальному значению.
2. *Расчет условных вероятностей.* Для расчета условных вероятностей применяем логит-модель. Логистическая регрессия – это статистическая модель, которая позволяет предсказать вероятности принадлежности объекта к какому-либо классу по значениям множества характеристик. В качестве зависимой бинарной переменной берем факт принадлежности пациента к основной группе. В качестве независимых переменных – харак-

теристики, потенциально влияющие на исход.

3. *Балансировка.* Для решения поставленной задачи используем метод подбора пар 1:1 при помощи поиска ближайшего соседа (nearest neighbor matching). Для каждого наблюдения из основной группы подбирается наиболее близкое по значению вероятности наблюдение из контрольной группы. Для оценки вклада изучаемого фактора наиболее интересно рассматривать пары, попадающие в зону перекреста значений вероятностей.
4. *Проверка качества баланса.* После подбора пар необходимо проверить согласованность групп по всем характеристикам. Для оценки согласия, помимо визуализации (рис. 1), воспользуемся капшой Коэна (Cohen's kappa) [5].

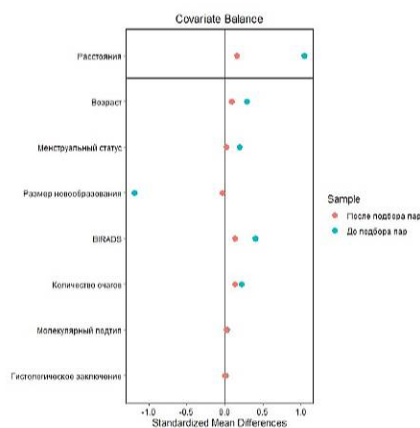


Рис. 1. График ковариантного баланса до и после подбора пар

Каппа Коэна = 0,767; 95% доверительный интервал: 0,656–0,878. Таким образом, по шкале интерпретации имеем существенное согласие групп [5].

5. *Оценка эффективности.* Для оценки и сравнения эффективности методов КУЗИ и МРТ проводим парный анализ на основании PMS и вычисляем основные и вспомогательные характеристики для каждого метода. Также строим ROC-кривую [5].

4. Результаты. В таблицах 1, 2 представлены основные и вспомогательные характеристики методов КУЗИ и МРТ до (453 наблю-

дения) и после подбора пар (134 наблюдения). На рис. 2 приведены ROC-кривые для методов КУЗИ и МРТ после подбора пар.

Таблица 1. Основные и вспомогательные характеристики метода КУЗИ до и после подбора пар

КУЗИ	% (До подбора пар)	95% ДИ (До подбора пар)	% (После подбора пар)	95% ДИ (После подбора пар)
Точность	95,32	92,14–97,49	93,28	87,63–96,88%
Чувствительность	89,02	80,18–94,86	87,04	75,1–94,63%
Специфичность	97,96	94,86–99,44	97,5	91,26–99,7%
ПЦПР	94,81	87,23–98,57	95,92	86,02–99,5%
ПЦОР	95,52	91,67–97,93	91,76	83,77–96,62%

Таблица 2. Основные и вспомогательные характеристики метода МРТ до и после подбора пар

МРТ	% (До подбора пар)	95% ДИ (До подбора пар)	% (После подбора пар)	95% ДИ (После подбора пар)
Точность	88,07	82,34–92,46	91,05	81,52–96,64%
Чувствительность	72,58	59,77–83,15	84,91	72,41–93,25%
Специфичность	96,49	91,26–99,04	95,06	87,84–98,64%
ПЦПР	91,84	80,4–97,73	91,84	80,4–97,73%
ПЦОР	86,61	79,43–92,01	90,59	82,29–95,85%

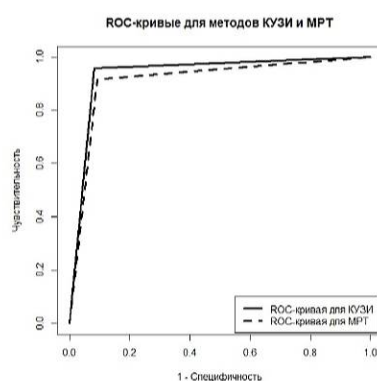


Рис. 2. ROC-кривые для методов КУЗИ и МРТ

5. Заключение. В статье рассмотрен метод псевдорандомизации и описан алгоритм, используемый для оценки эффективности диагностических методов КУЗИ и МРТ. Предложенный алгоритм реализован на языке R. По результатам работы программы видно, что оба метода диагностики показали отличные результаты, однако сравнительная эффективность метода КУЗИ выше.

Литература

1. Бусько Е. А. Мультипараметрическое ультразвуковое исследование с применением соноэластографии и эхоконтрастирования у пациенток с образованиями молочной железы. Свидетельство о регистрации электронного ресурса № 2019620435 от 06.03.2019.
2. Бусько Е. А. Магнитно-резонансная томография с динамическим контрастным усилением у пациенток с патологией молочной железы. Свидетельство о регистрации электронного ресурса № 2019621027 от 17.06.2019.
3. Rosenbaum P., Rubin D. The central role of the propensity score in observational studies for casual effects // *Biometrika*. 1983. P. 41–55.
4. Гржибовский А. М., Иванов С. В., Горбатов М. А. Использование псевдорандомизации (propensity score matching) для устранения систематических различий сравниваемых групп в обсервационных исследованиях с дихотомическим исходом // *Экология человека*. 2016. № 5. С. 50–64.
5. Краско О. Пособие по анализу медико-биологических данных (с примерами в R) [Электронный ресурс]: URL:https://www.researchgate.net/publication/312126731_Posobie_po_analizu_mediko-biologiceskih_dannyh_s_primerami_v_R (дата обращения: 07.11.2020).
6. Москалев А. В., Гладких А. А., Алышевская А. А. Возможность использования метода подбора больных по индексу соответствия (PSM) для устранения систематической ошибки отбора в ретроспективных нейрохирургических исследованиях // *Доказательная медицина*. 2018. С. 52–58.

УДК 001.891.57

Виль М. Ю., Гончарова А. Б.

Применение приложения MATLAB SimBiology для моделирования онкологического заболевания с учетом интерференционной конкуренции

Введение. Математическому моделированию биологических систем посвящено большое количество работ [1–5], причем часть из них рассматривает моделирование онкологических заболеваний [4, 5]. Чтобы оценить адекватность модели и соотнести ее результаты со статистическими данными, удобно использовать программные средства, специально предназначенные для анализа биологических моделей. Приложение SimBiology [6] представляет собой программные инструменты для моделирования, симуляции и анализа динамических систем для фармакокинетики, фармакодинамики и системной биологии. Данное программное обеспечение дает возможность строить модели как в интерактивном режиме (приложение SimBiology Model Builder), так и программно с помощью среды MATLAB непосредственно в командной строке, и предоставляет различные методы анализа моделей (приложение SimBiology Model Analyzer). Целью работы является моделирование онкологического заболевания с учетом интерференционной конкуренции в приложении SimBiology.

Построение модели опухоли в приложении SimBiology Model Builder. Рассмотрим модель, представленную в работе [7]:

$$\begin{cases} \frac{du}{dt} = \mu^u u \left(1 - \frac{u + vz}{K}\right), \\ \frac{dv}{dt} = \mu^v v \left(1 - \frac{u + v}{K}\right), \\ \frac{dz}{dt} = \mu^z uz. \end{cases} \quad (1)$$

Виль Мария Юрьевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st054723@student.spbu.ru, тел.: +7(913)337-75-55

Гончарова Анастасия Борисовна – старший преподаватель, Санкт-Петербургский государственный университет; e-mail: a.goncharova@spbu.ru, тел.: +7(812)428-47-29

Данная модель представляет собой описание процесса роста двух типов клеток (нормальные и злокачественные), который рассматривается как интерференционная конкуренция в функциональном пространстве. Здесь u – количество злокачественных клеток, v – количество нормальных клеток, μ^u , μ^v – удельные скорости их размножения, K – максимальное количество клеток, которое может разместиться на поверхности органа, $z(t)$ – функция, характеризующая факторы роста опухолевых клеток, влияющих на скорость оттеснения нормальных клеток от источников питания и, соответственно, ухода нормальных клеток от конкуренции, μ^z – активность потребления питания.

Реализуем модель (1) в виде блок-схемы в приложении SimBiology Model Builder. Как и в работе [7], примем объем функционального пространства за единицу. Такой выбор параметра K позволяет упростить модель, путем его исключения из построения. Создаем три разновидности. На рис. 1 они представляют собой прямоугольники со скругленными краями: U – злокачественные клетки, V – нормальные клетки и Z – разновидность, соответствующая функции $z(t)$. Далее добавляем параметры μ^u , μ^v , μ^z (в схеме на рис. 1 они скрыты, так как являются константами и никаких событий с ними не происходит).

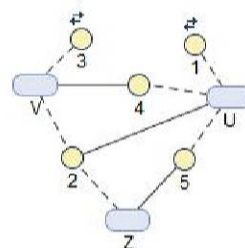


Рис. 1. Блок-схема модели в приложении SimBiology

Теперь перейдем к схематическому описанию взаимодействий между разновидностями – реакциям (на схеме – круги). Реакции в общем виде имеют запись $A + B \rightarrow C$, здесь A , B и C – разновидности. Разновидности в реакциях могут быть реагентами (в использованной выше записи это A и B), продуктами (C), а также и реагентами, и продуктами одновременно. В этом случае разновидность будет записана в реакции как справа, так и слева. Причем она может выходить из реакции как неизменной, так и с коэффициентом. Например, $A + B \rightarrow A$ или $A \rightarrow 2A$. Если разновидность является реагентом, то на схеме линия, соединяющая ее с реакцией, будет сплошной, если продуктом, то стрелкой от реакции к разновидности (в нашей схеме таких нет), если и реагентом, и продуктом

одновременно, то пунктирной.

Из блок-схемы приложение автоматически определяет и записывает вид дифференциальных уравнений. Их вид существенно зависит от формы записи реакции (и, соответственно, ее вида на блок-схеме). Можно коротко описать принципы построения ОДУ из схемы:

- если разновидность входит в реакцию исключительно как реагент, то выражение скорости данной реакции (задается в настройках реакции или записывается автоматически по форме задания реакции при указании кинетического закона (в нашем случае используется значение по умолчанию – MassAction) войдет в уравнение, соответствующее данной разновидности, со знаком «-»;
- если разновидность входит в реакцию только как продукт, то со знаком «+»;
- если же она одновременно является и продуктом, и реагентом, то данная реакция повлияет на вид соответствующего ОДУ только в том случае, если коэффициенты при разновидности справа и слева различны. В таком случае в уравнение для разновидности скорость реакции входит с коэффициентом, равным разности коэффициентов справа и слева.

Опишем реакции (см. таблицу 1), которые создадут в SimBiology систему ОДУ, аналогичную системе (1) (с учетом $K = 1$), причем те члены уравнений, которые реализуют ограничения на емкость среды, также будем описывать с помощью реакций (хотя фактически они ими не являются).

Таблица 1. Соотношение записи реакции и ее записи в модели

Номер реакции	Запись реакции	Скорость реакции
Реакции, влияющие на уравнения для разновидности U :		
1	$U \leftrightarrow 2U$	$\mu^u u - \mu^u u^2$
2	$U + V + Z \rightarrow V + Z$	$\mu^u uvz$
Реакции, влияющие на уравнения для разновидности V :		
3	$V \leftrightarrow 2V$	$\mu^v v - \mu^v v^2$
4	$U + V \rightarrow U$	$\mu^u uv$
Реакции, влияющие на уравнения для разновидности Z :		
5	$U + Z \rightarrow U$	$\mu^z uz$

Блок-схема, реализующая эти реакции, представлена на рис. 1.

Equations	
☐	ODEs
	$d(V)/dt = 1/c*((Mu*V - Mu*V^2) - (Mv*U*V))$
	$d(U)/dt = 1/c*((Mu*U - Mu*U^2) - (Mu*Z*V*U))$
	$d(Z)/dt = 1/c*(-(Mz*U*Z))$

Рис. 2. Система ОДУ, сгенерированная SimBiology

Сверив уравнения, сгенерированные автоматически (рис. 2) с системой (1), убеждаемся в их полном соответствии, так как в нашем случае наличие множителя $\frac{1}{c}$ является следствием технического ограничения о

том, что все разновидности должны находиться в отсеке, поэтому принимается значение $c = 1$.

Далее возьмем из статьи [7] значения параметров и начальные данные: $u = 0,05$; $v = 1$; $z = 1$; $u = v = 0,0038$; $z = 0,05$ и симулируем построенную модель с помощью приложения SimBiology Model Analyzer. Результат представлен на рис. 3. Он полностью совпадает с полученным в статье [7].

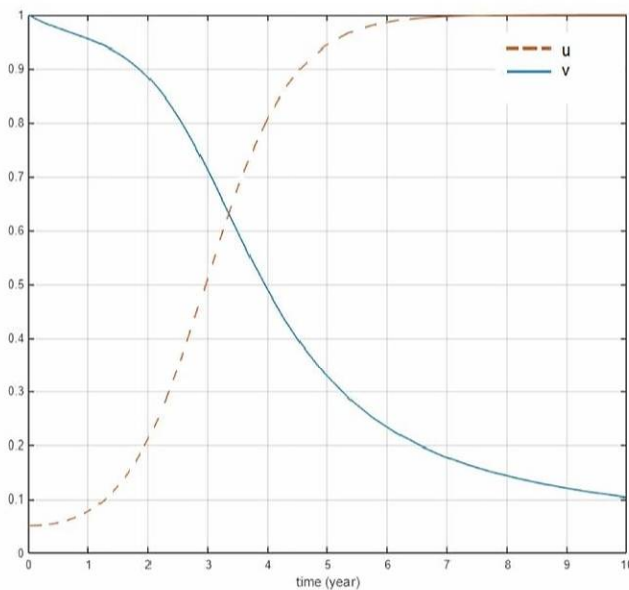


Рис. 3. Зависимость функций u и v от времени t

Заключение. Приложения SimBiology являются эффективным и удобным инструментом для построения и анализа как простых, так и более сложных моделей динамических систем. Их широкий функционал не только позволяет интерактивно моделировать заболевание, но и существенно упростить исследование его течения. В работе приведено сопоставление результатов моделирования с результатами, опубликованными в [7].

Литература

1. Моделирование и оценка состояния медико-эколого-экономических систем / Под ред. Батурина В. А. Новосибирск: Изд-во Сибирского Отделения РАН, 2005. 249 с.
2. Мюррей Д. Д. Математическая биология. М.: Ижевск: Изд-во «Регулярная и хаотическая динамика»: Институт компьютерных исследований. 2009. Т. 1. 776 с.
3. Мюррей Д. Д. Математическая биология. М.: Ижевск: Изд-во «Регулярная и хаотическая динамика»: Институт компьютерных исследований. 2011. Т. 2. 1104 с.
4. Гончарова А. Б., Колпак Е. П. Математическое моделирование лечения новообразования методом иммунотерапии // Современная наука: актуальные проблемы теории и практики. Серия «Естественные и технические науки». 2021. № 1. С. 74–78.
5. Sigal D., Przedborski M., Sivaloganathan D., Kohandel M. Mathematical modelling of cancer stem cell-targeted immunotherapy // Mathematical Biosciences. 2019. No 318. P. 3–28.
6. Документация SimBiology [Электронный ресурс]: URL:https://docs.exponenta.ru/simbio/index.html?s_tid=CRUX_lftnav (дата обращения: 11.03.2021)
7. Гончарова А. Б., Колпак Е. П., Расулова М. М., Абрамова А. В. Математическое моделирование лечения онкологического заболевания // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2020. Т. 16. № 4. С. 437–446.

УДК 004

Ларочкин П. В.

Исследование МРТ изображений головного мозга

Рекомендовано к публикации профессором Котиной Е. Д.

1. Введение. Исследования с помощью магнитно-резонансной томографии позволяют получать анатомические трехмерные изображения органов с высоким разрешением. Разработка программных средств для анализа МРТ изображений является в настоящее время актуальной задачей.

В данной статье исследуются МРТ изображения, в частности, рассматривается их обработка с целью классификации. Классификация происходит на два класса: изображения с опухолевыми образованиями и изображения без образований. Для классификации используется модель нейронной сети.

2. Описание набора данных. В качестве тренировочного набора данных рассматривается датасет из открытого источника [1]. Он содержит 3929 RGB изображений головного мозга и масок опухолевых образований. Примеры изображений представлены на рис. 1, 2.

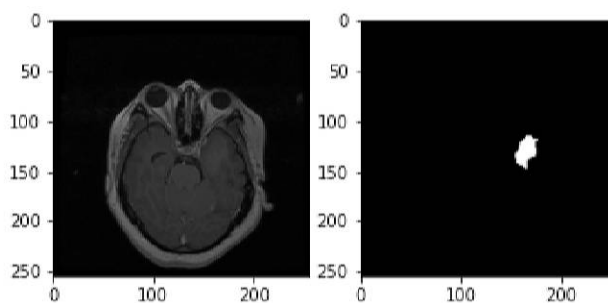


Рис. 1. Изображение с образованием/маска

Ларочкин Петр Викторович – студент, Санкт-Петербургский государственный университет; e-mail: plarochkin@gmail.com, тел.: +7(981)713-70-17

Маски опухолевых образований созданы в ручном режиме. В данной работе они применяются как маркер наличия опухоли.

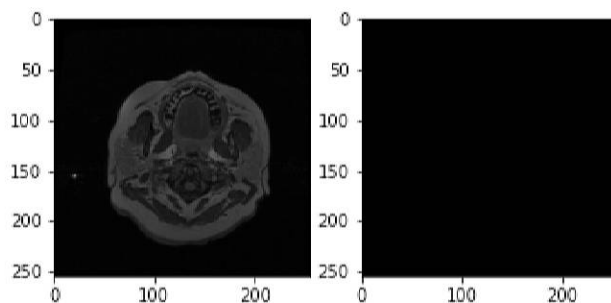


Рис. 2. Изображение без образования/маска

3. Классификатор. Для классификации изображений задействуется модель нейронной сети. Применяется архитектура *VGG16* [2] с добавлением линейного слоя и функции активации *Softmax*. Данная модель хорошо себя зарекомендовала в задачах классификации изображений.

Результатом работы модели является вероятность принадлежности изображения томограммы тому или иному классу. На рис. 3 приведена архитектура модели.

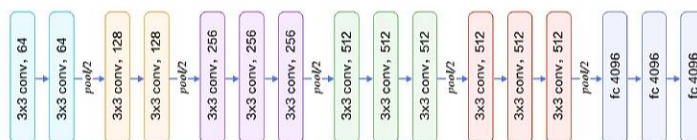


Рис. 3. Архитектура модели

4. Результаты классификации. Так как классы в датасете не сбалансированы ($1/2$), разумно использовать соответствующие метрики для анализа результатов обучения. В данной работе в качестве метрики используется *confusion matrix* и соответствующие ей метрики *recall(sensitivity)*, *precision* и *specificity*. Такие метрики не зависят от сбалансированности классов в выборке (в отличие от *accuracy*) и позволяют объективно оценивать модель. Результаты представлены в таблице.

Таблица. Confusion matrix для изображения с опухолевыми образованиями

	target = 1	target = 0
predict = 1	115	1
predict = 0	8	269

Соответствующая метрика *precision* составила 0,99138, *recall* – 0,93496, *specificity* – 0,99630.

Метрика *precision* интерпретируется как доля предсказаний объектов одного класса, которые действительно являются положительными, от всех положительных предсказаний для этого класса. Метрика *specificity* интерпретируется как доля верных предсказаний негативного класса от всех объектов негативного класса. Метрика *recall* интерпретируется как доля верных предсказаний положительного класса от всех объектов положительного класса [3].

5. Применение. Данная модель может применяться при исследовании томограмм на наличие опухолевых образований. Также данную классификацию можно применять перед сегментацией изображений, т. е. перед выделением областей с опухолевыми образованиями. Такое применение имеет смысл, поскольку при использовании нейронной сети *UNET* для сегментации часто получаются ложноположительные результаты, т. е. строятся маски на изображениях, где образования отсутствуют.

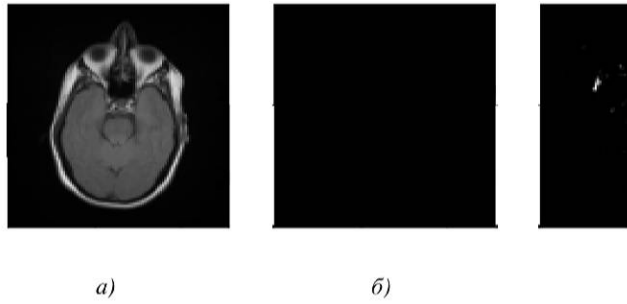


Рис. 4. *a)* томограмма, *б)* ручная маска, показывающая отсутствие образования, *в)* результат работы *UNET*

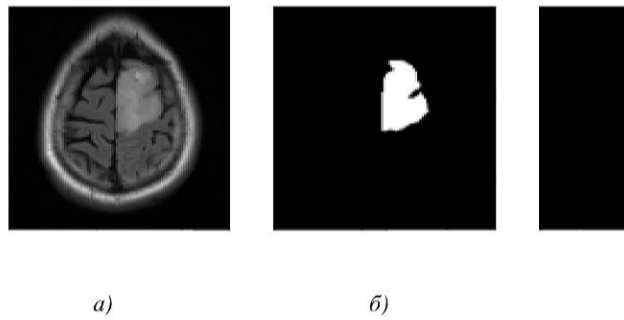


Рис. 5. *a)* томограмма, *б)* ручная маска, показывающая присутствие образования, *в)* результат работы *UNET*

На рис. 4 приводится томограмма с отсутствием образования. На данном изображении томограммы построенный классификатор выдает, что образования нет. Но при этом, в результате сегментации с использованием нейронной сети *UNET* мы получаем область образования.

Однако на изображениях, где образования присутствуют, *UNET* показывает более точные результаты (см. рис. 5).

Применение классификатора перед сегментацией может значительно повысить качество сегментации, заранее исключив снимки, которые не нуждаются в поиске образований.

6. Заключение. В данной работе проведены исследования по построению алгоритма классификации и его возможного применения для улучшения сегментации. Произведена оценка эффективности модели.

Таким образом, автоматическая классификация изображений позволит выделить из больших наборов данных томограммы с наличием опухолевых образований для дальнейшей цифровой обработки. В качестве цифровой обработки может выступать сегментация образований на снимках, а также их дальнейшая классификация на более узкие классы. В настоящее время, когда автоматизация процесса обработки приобретает все большее значение, разработки в данной области являются актуальными.

Литература

1. Kaggle [Электронный ресурс]: URL:<https://www.kaggle.com/linebreak/mateuszbeda/lgg-mri-segmentation> (дата обращения: 20.03.2021).
2. Pytorch [Электронный ресурс]: URL:<https://pytorch.org/docs/linebreak/stable/index.html> (дата обращения: 20.03.2021).
3. Habr [Электронный ресурс]: URL:<https://habr.com/ru/company/linebreak/ods/blog/328372/> (дата обращения: 20.03.2021).

УДК 519.6

Леонова Е. Б., Котина Е. Д.

Построение поля перемещений для последовательностей изображений с использованием дискретных систем

1. Введение. В современной диагностической медицине нельзя обойтись без методов обработки и анализа изображений, поэтому создание и совершенствование таких методов является актуальной задачей. В данной статье представлено развитие модели дискретной оптимизации, предложенной в статье [1], на основе вариации интегрального функционала в задачах управления ансамблями траекторий, представленных в работах [2, 3].

2. Постановка задачи. Рассмотрим дискретную систему

$$y(k+1) = F(k, y(k), u(k)), \quad k = 0, \dots, N-1, \quad (1)$$

с начальным условием $y(0) = y_0$, $y_0 \in M_0CR^m$. Здесь $y(k)$ – m -мерный вектор состояния, $u(k)$ – r -мерный вектор параметров. $F(k, y(k), u(k))$ – m -мерная векторная функция, такая, что при каждом $k \in \{0, 1, \dots, N-1\}$ она определена и непрерывна на множестве $\Omega_y \times U(k)$ по всем своим аргументам $(y(k), u(k))$ вместе с частными производными первого и второго порядка по этим переменным. Множества $\Omega_y \subset R^m$ и $U(k)$, $k = 0, 1, \dots, N-1$, – компактные множества в R^r . Якобиан $J_k = J(k, y(k), u(k)) = \left| \frac{\partial F(k, y(k), u(k))}{\partial y(k)} \right| \neq 0$ при всех изменениях $k, y(k), u(k)$.

Будем полагать, что система (1) задает перемещение точек изображения исследуемого объекта в последовательности кадров. Множество векторов перемещений, построенных в точках изображения, будем называть полем перемещений.

Леонова Екатерина Борисовна – аспирант, Санкт-Петербургский государственный университет; e-mail: st062324@student.spbu.ru, тел.: +7(911)748-46-14

Котина Елена Дмитриевна – профессор, Санкт-Петербургский государственный университет; e-mail: e.kotina@spbu.ru, тел.: +7(812)428-48-68

Пусть $\rho = \rho(k) = \rho(k, y(k))$ яркость изображения на k -м шаге. Также это может быть функция плотности распределения радиофармпрепарата в случае радионуклидных изображений [4].

Положим, что начальное состояние системы описывается компактным в R^m множеством M_0 ненулевой меры с плотностью распределения (яркостью) $\rho_0 = \rho_0(y_0) = \rho(0, y_0)$. Последовательность векторов $\{u(0), \dots, u(N)\}$ обозначим как u . Последовательность векторов $\{y(0), \dots, y(N)\}$ будем называть траекторией движения и обозначать через $y = y(y_0, u)$. Тогда с помощью $y(k) = y(k, y_0, u)$ обозначим состояние траектории на k -м шаге.

Ансамблем траекторий будем называть множество траекторий $y(y_0, u)$, которые соответствуют заданному u и различным начальным состояниям $y_0 \in M_0$. Через $M_{k,u}$ обозначим сечение пучка траекторий на k -м шаге [1].

Функция $\rho(k, y(k))$ при каждом $k \in \{0, \dots, N\}$ определена и непрерывна на множестве $M_{k,u}$ по $y(k)$ вместе с частными производными, обозначим $y_k = y(k)$.

Будем полагать, что плотность $\rho(k, y_k)$ вдоль траекторий системы не изменяется (случай оптического потока [5]):

$$\rho(k+1, y(k+1)) = \rho(k, y_k), \quad k = 0, \dots, N-1,$$

начальное условие $\rho(0, y_0) = \rho_0(y_0)$.

Рассмотрим функционал качества общего вида [1]

$$I(u) = \sum_{k=1}^{N-1} \int_{M_{k,u}} \varphi_k(y_k, \rho(k, y_k), u_k) dy_k + \int_{M_{N,u}} g(y_N, \rho(N, y_N)) dy_N. \quad (2)$$

Здесь $\varphi_k = \varphi_k(y_k, \rho(k, y_k), u_k)$, $k = 1, \dots, N-1$ и $g = g(y_N, \rho(N, y_N))$ – непрерывно-дифференцируемые функции своих аргументов, y_k – переменная интегрирования по множеству $M_{k,u}$, $k = 1, \dots, N$.

Для рассматриваемой системы с соответствующими начальными условиями необходимо найти такое допустимое u , чтобы минимизировать данный функционал качества. Под допустимыми управлениями будем понимать управления, удовлетворяющие условиям

$$\|u(k)\| \leq m^2, \quad k = 0, \dots, N-1, \quad m \in R^1.$$

3. Построение поля перемещений. Рассмотрим последовательность радионуклидных изображений, собранных в динамическом режиме, т. е. известной является плотность распределения радиофармпрепарата на каждом кадре: $\bar{\rho}(k, y(k)), k = 0, \dots, N$, причем $\bar{\rho}(0, y(0)) = \rho_0(y_0)$. Необходимо восстановить функцию $F(k, y(k), u(k))$, в которой неизвестным считаем вектор $u(k)$. Будем искать его, минимизируя частный случай функционала (2):

$$I(u) = \sum_{k=1}^N \int_{M_{k,u}} (\rho(k, y_k) - \bar{\rho}(k, y_k))^2 dy_k.$$

Определив вектор параметров $u(k)$, можно построить поле перемещений в соответствии с уравнением (1).

Далее рассмотрим только один шаг процесса $N = 1$. Система (1) примет вид

$$y(1) = F(0, y(0), u(0)).$$

Обозначим $y(0) = y, u(0) = u$, а $F(0, y, u) = F(y, u)$.

Рассмотрим линейную вектор-функцию

$$F(y, u) = Ay + B, \quad (3)$$

где y – двумерный вектор,

$$A = \begin{pmatrix} \cos u_1 & u_2 \sin u_1 \\ \frac{-1}{u_2} \sin u_1 & \cos u_1 \end{pmatrix}, \quad B = \begin{pmatrix} u_3 \\ u_4 \end{pmatrix}.$$

Тогда вектор искомых параметров $u = (u_1, u_2, u_3, u_4)$. Для данного случая выпишем представление вариации функционала

$$\delta I(u) = \int_{M_0} p^T(1) \frac{\partial F(y, u)}{\partial u} dy \Delta u,$$

где $p^T(1) = \frac{\partial g(1)}{\partial y(1)}$,

$$\frac{\partial F(y, u)}{\partial u} = \begin{pmatrix} -y^1 \sin u_1 + y^2 u_2 \cos u_1 & y^2 \sin u_1 & 1 & 0 \\ \frac{-y^1}{u_2} \cos u_1 - y^2 \sin u_1 & \frac{-y^1}{u_2^2} \sin u_1 & 0 & 1 \end{pmatrix}.$$

Обозначим $(p_1, p_2) = p^T(1)$. Получаем

$$\begin{aligned} \frac{\partial I}{\partial u_1} &= \int_{M_0} p_1 (y^2 u_2 \cos u_1 - y^1 \sin u_1) - p_2 \left(y^1 \frac{-1}{u_2} \cos u_1 - y^2 \sin u_1 \right) dy, \\ \frac{\partial I}{\partial u_2} &= \int_{M_0} \left(p_1 y^2 \sin u_1 + p_2 y^1 \frac{-1}{u_2} \sin u_1 \right) dy, \\ \frac{\partial I}{\partial u_3} &= \int_{M_0} p_1 dy, \\ \frac{\partial I}{\partial u_4} &= \int_{M_0} p_2 dy. \end{aligned} \quad (4)$$

На основе данного выражения градиента функционала (4) можно строить различные направленные методы оптимизации, используя для нахождения вектора параметров итерационную формулу

$$u^{i+1} = u^i - \alpha \text{grad } I(u^i),$$

где α – параметр метода градиентного спуска.

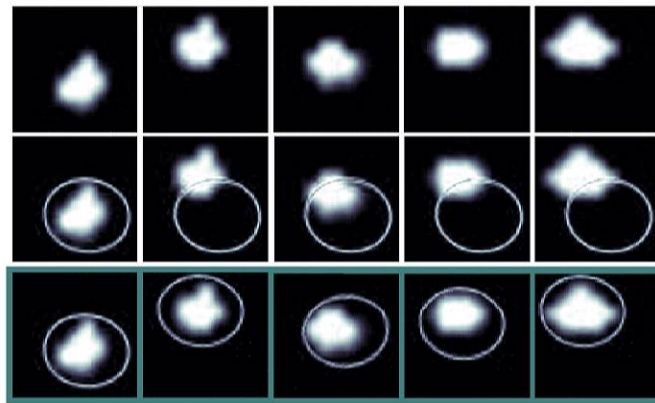


Рис. 1. Вверху – последовательность кадров, в середине – контур области интереса без коррекции, внизу – скорректированное положение контура на изображениях

4. Применение для радионуклидных изображений. На основе полученных выражений (3), (4) в среде MATLAB был реализован алгоритм построения поля перемещений и соответствующей коррекции положения контура. На рис. 1, в третьем ряду представлено скорректированное положение контура области интереса на изображениях. Данная задача является актуальной для автоматизации процесса обработки многокадровых динамических исследований для дальнейшего корректного построения соответствующих динамических кривых.

5. Заключение. Рассмотрен случай оптического потока при построении поля перемещений на основе модели дискретной оптимизации, предложенной в [1]. Разработан алгоритм коррекции положения контуров на изображениях при обработке планарных динамических радионуклидных исследований. Результаты работы могут быть использованы как для коррекции движения на радионуклидных изображениях, так и для анализа различных изображений.

Литература

1. Котина Е. Д., Леонова Е. Б., Плоских В. А. Обработка радионуклидных изображений с использованием дискретных систем // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 4. С. 544–554.
2. Овсянников Д. А. Моделирование и оптимизация динамики пучков заряженных частиц. Л.: Изд-во Ленингр. ун-та, 1990. 312 с.
3. Kotina E. D. Discrete optimization problem in beam dynamics. // Nuclear Instruments and Methods in Physics Research. 2006. Vol. 558. Iss. 1. P. 292–294.
4. Kotina E. D., Pasechnaya G. A. Optical flow-based approach for the contour detection in radionuclide images processing // Cybernetics and physics. 2014. Vol. 3. No 2. P. 62–65.
5. Horn B. K. P., Schunck B. G. Determining optical flow // Artificial intelligence. 1981. Vol. 17. No 11. P. 185–203.

УДК 51.76

Моисеева В. Д.

Анализ одной мультивиральной эпидемической модели

Рекомендовано к публикации доцентом Губар Е. А.

1. Введение. В настоящее время эпидемические модели являются подходящим инструментом для описания распространения биологических вирусов в популяциях людей или животных, информации в социальных сетях и вредоносных программ в компьютерных сетях.

В качестве базовой модели для описания процессов распространения эпидемии в различных средах используется модель susceptible-infected-recovered (SIR) [1], в которой S – это доля восприимчивых агентов, которые не инфицированы, но могут заразиться, I – это доля инфицированных (тех, кто уже был инфицирован и может передать инфекцию восприимчивым агентам), R – доля вылечившихся (иммунных).

В данной работе рассматривается модификация классической модели, в которой предполагается, что существуют два различных эпидемических процесса: распространение информации и распространение двух типов вирусов в популяции. Подробно она будет описана далее в разделе 3.

Одним из важнейших показателей распространения эпидемии является базовое репродуктивное число, обозначаемое R_0 , – коэффициент, отражающий среднее количество людей, которые могут быть заражены одним инфицированным человеком в популяции, не имеющей иммунитета [2]. Если $R_0 > 1$, то эпидемический процесс развивается, иначе – идет на спад. Значение R_0 зависит от длительности инфекционного периода, вероятности инфицирования при контакте восприимчивого человека с инфекционным и частоты контактов.

Целью данной работы является анализ числа R_0 для модели SWIRS в зависимости от числа контактов между людьми.

Моисеева Вероника Денисовна – студент, Санкт-Петербургский государственный университет; e-mail: nika-moiseeva@list.ru, тел.: +7(996)765-06-87

2. Вычисление базового репродуктивного числа. В литературе известно много способов для вычисления R_0 , здесь представлен один из них – метод матрицы следующего поколения [3]. Он используется для моделей распространения инфекционных заболеваний, в которых всех агентов можно разделить на группы по состояниям, в которых они находятся. Группы, которые состоят из инфицированных агентов, назовем инфицированными (I). Пусть $x_i, i = 1, \dots, m$ – число инфицированных агентов в i -й инфицированной группе в момент времени t . За $\tilde{F}_i(x)$ обозначим поток инфицирования – компоненту системы, отвечающую за появления новых инфекций в группе i , а за $\tilde{V}_i(x)$ – поток перемещений агентов между группой i и всеми другими группами. Получаем

$$\frac{dx_i}{dt} = \tilde{F}_i(x) - \tilde{V}_i(x). \quad (1)$$

Пусть x_0 – точка равновесия системы (1) в случае отсутствия инфицированных групп, т. е. ее правая часть равна нулю (обозначается как DFE – disease-free equilibrium state) – это состояние системы, в котором все инфицированные группы пусты. Определим следующие матрицы:

$$F = \left[\frac{\partial \tilde{F}_i(x_0)}{\partial x_j} \right], \quad V = \left[\frac{\partial \tilde{V}_i(x_0)}{\partial x_j} \right], \quad 1 \leq i, \quad j \leq m. \quad (2)$$

Матрица F соответствует появлению новых инфицированных агентов в группах, матрица M – переходу этих агентов в другие группы.

Матрицей следующего поколения называется матрица

$$G = FV^{-1}. \quad (3)$$

В уравнении (3) (i, j) -элемент – это ожидаемое число вторичных инфекций в группе i , произведенных инфицированным индивидуумом, введенным в группу j . Наибольшее собственное число этой матрицы и есть искомое R_0 .

3. Вычисление R_0 для модели SWIRS. Определим эпидемиологическую модель SWIRS [4]: S (susceptible) – восприимчивые к инфекции, W (warned) – получившие информацию о возможном распространении вирусов и готовые использовать специальные средства защиты, I_1 и I_2 (infected) – инфицированные различными типами вирусов V_1 и V_2 [5], R (recovered) – иммунные. На конце стоит состояние S , потому что вылечившиеся агенты могут снова становиться восприимчивыми к вирусу, так как пожизненный иммунитет к вирусам не вырабатывается. Особенностью модели является то, что вирус первого типа с какой-то вероятностью может мутировать в вирус второго типа.

Эпидемиологический процесс моделируется системой нелинейных дифференциальных уравнений. Общее количество агентов в популяции в течение всего процесса остается постоянным и равно $n_S + n_W + n_{V_1} + n_{V_2} + n_R = N$, где n_i – количество агентов в группе i в определенный момент времени. Обозначим доли восприимчивых, предупрежденных, инфицированных и выздоровевших индивидуумов следующим образом:

$$S(t) = \frac{n_S(t)}{N}, \quad W(t) = \frac{n_W(t)}{N}, \quad I_1(t) = \frac{n_{V_1}(t)}{N}, \\ I_2(t) = \frac{n_{V_2}(t)}{N}, \quad R(t) = \frac{n_R(t)}{N}.$$

Все величины имеют значения от 0 до 1, и выполняется закон сохранения: $S(t) + W(t) + I_1(t) + I_2(t) + R(t) = 1$.

В начале эпидемии, в момент времени $t = 0$, большинство индивидуумов находится в восприимчивом состоянии, и небольшая часть заражена различными типами вирусов. Следовательно, начальные состояния равны: $0 < S(0)$; $0 < W(0)$; $0 < I_1(0)$; $0 < I_2(0)$; $R(0) = 1 - S(0) - W(0) - I_1(0) - I_2(0)$.

Система, описывающая модель, выглядит следующим образом:

$$\begin{cases} \dot{S} = -kWS - \beta_1^S SI_1 - \beta_2^S SI_2 + \gamma R - u_3 S, \\ \dot{W} = kW S - \beta_1^W WI_1 - \beta_2^W WI_2 + u_3 S - \sigma_3 W, \\ \dot{I}_1 = \beta_1^S SI_1 + \beta_1^W WI_1 - \varepsilon I_1 I_2 - \sigma_1 I_1 - u_1 I_1, \\ \dot{I}_2 = \beta_2^S SI_2 + \beta_2^W WI_2 + \varepsilon I_1 I_2 - \sigma_2 I_2 - u_2 I_2, \\ \dot{R} = \sigma_1 I_1 + u_1 I_1 + \sigma_2 I_2 + u_2 I_2 + \sigma_3 W - \gamma R. \end{cases} \quad (4)$$

Переход некоторых агентов из информированного состояния W сразу в иммунное R объясняется тем, что агент может вакцинироваться и получить иммунитет без болезни.

После потери иммунитета агент возвращается в восприимчивое состояние S , т. к. предполагается, что старая информация о вирусе уже не актуальна и для перехода в W необходимо получить новую.

В данной модели коэффициенты имеют следующий смысл:

- β_1^S и β_2^S – вероятности распространения инфекции для восприимчивых агентов, $\beta_1^S = \alpha_1 m$, $\beta_2^S = \alpha_2 m$, где α_i – вероятность передачи инфекции за один контакт, m – среднее количество контактов в день [6];
- β_1^W и β_2^W – вероятности распространения инфекции для предупрежденных агентов, $\beta_1^W = \xi_1 m$, $\beta_2^W = \xi_2 m$, где ξ_i – вероятность передачи инфекции за один контакт, m – среднее количество контактов в день;
- σ_1 и σ_2 – скорости восстановления зараженных агентов, $\sigma_i = \frac{1}{T_i}$, $i = 1, 2$, где T – продолжительность болезни, спровоцированной конкретным вирусом;
- ε – вероятность перехода агентов из группы, зараженной вирусом V_1 , в группу, зараженных вирусом V_2 ;
- σ_3 – вероятность вакцинации информированных агентов;
- u_1 , u_2 и u_3 – коэффициенты управления; в данной работе идет анализ модели с нулевым управлением;
- γ – вероятность перехода в восприимчивое состояние из иммунного;
- k – вероятность получить информацию о вирусе.

Все параметры могут принимать значения в диапазоне $[0, 1]$.

Перейдем к вычислению базового репродуктивного числа, используя метод, описанный в третьем разделе. Необходимо определить состояние системы, в котором все инфицированные группы пусты (disease-free equilibrium state): $(S(0), 0, 0, 0, R(0))$.

Далее, используя формулу (2), найдем матрицы

$$F = \begin{bmatrix} \beta_1^S S + \beta_1^W kS & 0 \\ 0 & \beta_2^S S + \beta_2^W kS \end{bmatrix} \text{ и } V = \begin{bmatrix} \sigma_1 + u_1 & \varepsilon \\ -\varepsilon & \sigma_2 + u_2 \end{bmatrix}.$$

Тогда матрица следующего поколения

$$G = FV^{-1} = \begin{bmatrix} \frac{(\beta_1^S S + \beta_1^W kS)(\sigma_2 + u_2)}{w} & \frac{-\varepsilon^2(\beta_1^S S + \beta_1^W kS)}{w} \\ \frac{\varepsilon^2(\beta_2^S S + \beta_2^W kS)}{w} & \frac{(\beta_2^S S + \beta_2^W kS)(\sigma_1 + u_1)}{w} \end{bmatrix}, \quad (5)$$

где $w = -\varepsilon^2 - (\sigma_2 + u_2)(\sigma_1 + u_1)$.

Наибольшее собственное число матрицы G – это искомое базовое репродуктивное число

$$R_0 = \frac{\left(\frac{(\beta_1^S S + \beta_1^W kS)(\sigma_2 + u_2)}{w} + \frac{(\beta_2^S S + \beta_2^W kS)(\sigma_1 + u_1)}{w} \right) + \sqrt{D}}{2}, \quad (6)$$

где D – дискриминант квадратного уравнения, полученного в расчетах:

$$D = \left(\frac{(\beta_1^S S + \beta_1^W kS)(\sigma_2 + u_2)}{w} + \frac{(\beta_2^S S + \beta_2^W kS)(\sigma_1 + u_1)}{w} \right)^2 - 4 \frac{(\beta_1^S S + \beta_1^W kS)(\beta_2^S S + \beta_2^W kS)((\sigma_1 + u_1)(\sigma_2 + u_2) + \varepsilon^2)}{w^2}.$$

4. Численный эксперимент. Выясним зависимость R_0 от числа контактов в день. В данной модели, опираясь на реальные эпидемии, продолжительности болезни возьмем $T_1 = 14, T_2 = 10$. Тогда $\sigma_1 = \frac{1}{14}$ и $\sigma_2 = \frac{1}{10}$. Коэффициенты управления примем $u_1 = u_2 = u_3 = 0$. Коэффициент перехода первого вируса во второй зададим $\varepsilon = 0,003$. Вероятность информирования $k = 0,1$. Вероятность перехода в восприимчивое состояние из иммунного $\gamma = 0,1$.

Вероятности распространения инфекции для восприимчивых и информированных в зависимости от типа вируса за один контакт в нашей модели имеют следующие значения: $\alpha_1 = 0,005, \alpha_2 = 0,025$ и $\xi_1 = 0,001, \xi_2 = 0,01$. Напомним, что $\beta_1^S = \alpha_1 t, \beta_2^S = \alpha_2 t$ и $\beta_1^W = \xi_1 t, \beta_2^W = \xi_2 t$, где t – среднее количество контактов в день. Подставив значения коэффициентов в формулу (6) и проведя необ-

ходимые расчеты в среде MATLAB, получаем зависимость R_0 от числа контактов, которая приведена в таблице. Полученные значения для R_0 показывают, что чем меньше агенты контактируют друг с другом, тем медленнее распространяется инфекция. Эксперимент показал, что при заданных параметрах эпидемия пойдет на спад, если число контактов $m < 5$. При $m = 5$ имеем $R_0 = 1,0981$.

Таблица 1. Зависимость R_0 от числа контактов и ε

m	R_0	ε	R_0
3	0,6589	0,001	0,6599
7	1,5374	0,004	0,658
10	2,1962	0,007	0,6539
15	3,2943	0,01	0,6475

Также в таблице показана зависимость R_0 от коэффициента ε – вероятности перехода агентов из группы, инфицированной первым типом вируса, в группу, инфицированную вторым. Инфекционный период второго вируса меньше, поэтому с увеличением ε базовое репродуктивное число уменьшается. В расчетах, представленных в таблице, число контактов $m = 4$, но эксперимент показал, что данная тенденция справедлива и для другого числа контактов.

5. Заключение. В статье описана модель SWIRS, в которой помимо распространения двух типов вирусов, есть также распространение информационного потока, что дает возможность анализировать более точно базовое репродуктивное число R_0 , которое показывает, будет эпидемический процесс развиваться дальше или же пойдет на спад. Также выявлена зависимость значений R_0 от числа контактов в день и коэффициента перехода первого вируса во второй при конкретных условиях эпидемии.

Литература

1. Kermack W.O., McKendrick A.G. A contribution to the mathematical theory of epidemics // The Royal Society. 1927. Vol. 115. No 772. P. 700–721.
2. Dietz K. The estimation of the basic reproduction number for infectious diseases // Statistical Methods in Medical Research. 1993. Vol. 2. No 1. P. 23–41.

3. Van den Driessche P. Reproduction numbers of infectious disease models // *Infectious Disease Modelling*. 2017. No 2. P. 288–303.
4. Taynitskiy V., Gubar E., Fedyanin D. et al. Optimal control in the multi-virus epidemic model under information spreading // *IFAC-PapersOnLine System*. 2020. Vol. 53. No 2. P. 6650–6655.
5. Gubar E., Taynitskiy V., Zhu Q. Optimal control of heterogeneous mutating viruses // *Games*. 2018. Vol 9. No 4. P. 103–122.
6. Pastor-Satorras R., Castellano C., Van Mieghem P. et al. Epidemic processes in complex networks // *Reviews of modern physics*. 2015. Vol. 87. No 3. P. 925–972.

УДК 004.932.1

Петрова О. Г.

Сегментация МРТ изображений позвоночника с использованием сверточных нейронных сетей

Рекомендовано к публикации профессором Котиной Е. Д.

1. Введение. Обработка медицинских изображений – трудоемкий процесс, требующий предельной внимательности и профессионализма от специалиста, а также определенных временных затрат. Во многих случаях быстрая и своевременная диагностика может в значительной мере облегчить и ускорить процесс выздоровления пациента. В последнее время очень активно идет разработка и создание средств для автоматизации сегментации медицинских изображений, использование которых позволит ускорить процесс постановки диагноза и повысить его точность. Также подобные программы могут обеспечить более раннюю диагностику, обнаружив физические отклонения малых размеров, которые врач еще не может заметить. Методов, которые могут использоваться в задачах сегментации, множество. К ним относят классические методы компьютерного зрения [1], методы машинного обучения, нейронные сети [2, 3].

В статье описывается обработка МРТ изображений позвоночника с использованием сверточных нейронных сетей. Целью является разработка метода автоматической сегментации, который позволит выделить на изображении каждый позвонок и позвоночный отросток.

2. Реализация. Для обучения использовались изображения позвоночника в сагиттальной проекции, полученные из открытого набора SpineSagT2Wdataset3. Обучающая и тестовая выборки состоят из 195 и 14 файлов в формате NIFTY соответственно.

2.1. Архитектура модели. Для реализации выбрана сверточная нейронная сеть, основанная на архитектуре U-net [4]. В классическую модель были добавлены дополнительные слои так, чтобы вектор признаков на первом и предпоследнем слое имел 32 компонента. Всего полученная сеть имеет 33 сверточных слоя.

Петрова Олеся Геннадьевна – студент, Санкт-Петербургский государственный университет; e-mail: olesyapetrova21@mail.ru, тел.: +7(981)847-61-30

2.2. Предобработка изображений. Изначально все изображения были преобразованы из формата NIfTI в png, а размер изменен на 512×512 для ускорения обучения. Для повышения качества сегментации области изображения, которые не представляют интереса, были затемнены. Бинарные маски построены специалистами вручную. Значение пикселя, равное единице, означает его принадлежность телу позвонка или позвоночному отростку, равное нулю – фону. На рис. 1 представлен пример изображения и его маски.

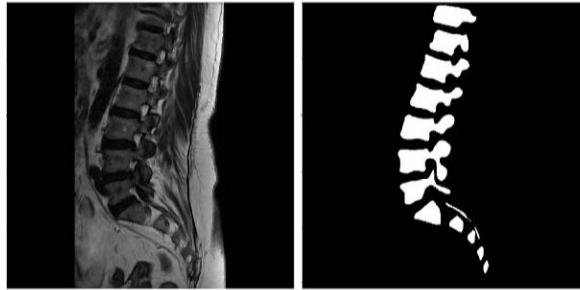


Рис. 1. Пример исходного изображения и его маски

Следующим этапом предобработки является построение матрицы смежности уровней серого (GLSM) [5]. Основная идея метода заключается в построении матрицы, каждый элемент которой равен числу случаев, когда пиксель с яркостью i находится в заданной окрестности $(\Delta x, \Delta y)$ с пикселем яркости j . Каждый элемент матрицы смежности вычисляется по формуле

$$G_{(\Delta x, \Delta y)}(i, j) = \sum_{y=0}^{h-\Delta y} \sum_{x=0}^{w-\Delta x} F,$$

где

$$F = \begin{cases} 1, & \text{если } I(x, y) = i \text{ и } I(x + \Delta x, y + \Delta y) = j, \\ 0, & \text{иначе.} \end{cases}$$

Здесь w, h – ширина и высота исходного изображения, $I(x, y)$ – яркость пикселя с координатами (x, y) .

В зависимости от выбранных значений Δx и Δy можно построить разные матрицы смежности. Обычно используются $G_{(1,0)}$, $G_{(1,1)}$,

$G_{(0,1)}$ и $G_{(-1,-1)}$, каждая из которых нормируется:

$$P_{(\Delta x, \Delta y)} = \frac{G_{(\Delta x, \Delta y)}(i, j)}{(w - \Delta x)(h - \Delta y)}.$$

На основе полученных матриц можно вычислить текстурные признаки Харалика. Для обработки изображений выбран фильтр на основе контраста

$$Contrast = \sum_{i=1}^{N-1} \sum_{j=1}^{N-1} (i - j)^2 P_{(i, j)},$$

где N – размерность матрицы $P_{(\Delta x, \Delta y)}$.

На рис. 2 представлен пример обработки изображения.

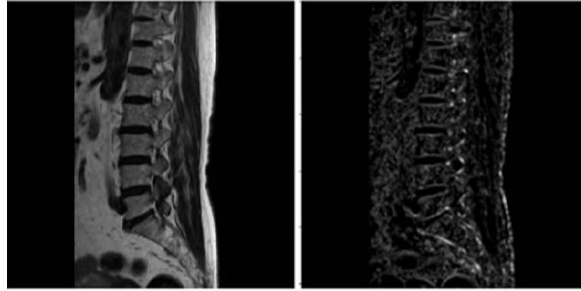


Рис. 2. Результат обработки изображения

2.3. Метрики. В качестве метрики для обучения сети и оценки качества сегментации выбран коэффициент Дайса, который в случае бинарных изображений вычисляется по формуле

$$D(X, Y) = \frac{2 \sum_{i, j=1}^N p_{ij} g_{ij} + 1}{\sum_{i, j=1}^N p_{ij} + \sum_{i, j=1}^N g_{ij} + 1},$$

где p_{ij} – значение пикселя (i, j) изображения P , а g_{ij} – значение пикселя (i, j) изображения G . В качестве функции потерь была выбрана ошибка коэффициента Дайса

$$l(X, Y) = 1 - D(X, Y).$$

3. Результаты. Нейронная сеть была реализована на языке Python с использованием библиотек Keras и TensorFlow. В качестве метода оптимизации выбран RMSprop. Начальная скорость обучения была принята равной 0,001, а размер батча – равным 10. Обучающая, валидационная и тестовая выборки содержат 1722, 516 и 222 изображения соответственно.

Обучение проводилось на стационарном компьютере со следующими характеристиками: ЦПУ: Intel(R) Core (TM) i5-7600K CPU; ОЗУ: 16 ГБ; ГПУ: NVIDIA GeForce GTX 1070 Ti, 8 Гб.

Значения функции потерь на обучающей выборке составило 0,07, на тестовой – 0,086. Коэффициент Дайса для обучающей и тестовой выборок равен 0,93 и 0,914 соответственно. Обучение проводилось на протяжении 60 эпох, время обучения составило 1 час 53 минуты 24 секунды.

Пример работы нейронной сети представлен на рис. 3.

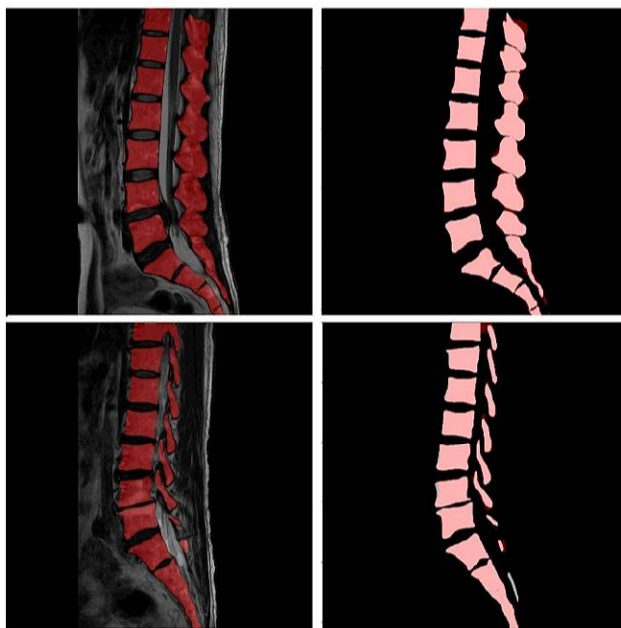


Рис. 3. Результат работы нейронной сети

4. Заключение. Представлена архитектура сверточной сети на основе U-net для сегментации МРТ изображений позвоночника. В качестве предварительной обработки выбран метод, основанный на матрице смежности уровней серого. На тестовой выборке была достигнута точность 0,914 с использованием предобработки и 0,907 – без. В дальнейшем для повышения точности сегментации планируется рассмотреть другие методы предобработки изображений.

Литература

1. Georg P., Saalfeld S., Serowy S., Tonnies K. Vertebral body segmentation in wide range clinical routine spine MRI data // Computer Methods and Programs in Biomedicine. 2018. No 155. P. 93–99.
2. Lu J-T., Pedemonte S., Bizzo B. et al. Deep spine: automated lumbar vertebral segmentation, disc-level designation, and spinal stenosis grading using deep learning [Электронный ресурс]: URL:<https://arxiv.org/pdf/1807.10215v1.pdf> (дата обращения: 10.03.2021).
3. Huang J., Shen H., Wu J. et al. Spine Explorer: a deep learning based fully automated program for efficient and reliable quantifications of the vertebrae and discs on sagittal lumbar spine MR images // The Spine Journal. 2020. No 4. P. 590–599.
4. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional networks for biomedical image segmentation // Medical Image Computing and Computer-Assisted Intervention. 2015. No 9351. P. 234–241.
5. Haralick R. Statistical and structural approaches to texture // Proceedings of the IEEE. 1979. Vol. 67. No 5. P. 786–804.

УДК 519.251

Тихонова Е. О.

Статистический анализ изменения внутриглазного давления после интравитреальных инъекций

Рекомендовано к публикации доцентом Корниковым В. В.

Введение. В настоящее время для лечения таких заболеваний глаз, как возрастная макулярная дистрофия, диабетический макулярный отек, тромбозы вен и артерий сетчатки, микроаневризмы, активно используется метод интравитреальной инъекции (ИВИ). Это связано с тем, что из известных способов введения лекарственных веществ для доставки их к тканям глаза наибольшая концентрация создается именно при ИВИ [1]. Введение дополнительного объема жидкости или газа приводит к повышению внутриглазного давления (ВГД) [2, 3], что, в свою очередь, может вызывать осложнения [4]. В связи с этим важно знать, возвращается ли ВГД в норму после ИВИ, за какое время и насколько резко это происходит, а также какие биометрические параметры глаза имеют влияние на величину изменения ВГД. Цель представленного исследования – основываясь на данных эксперимента определить динамику изменения ВГД после инъекции, изучить влияние на неё биометрических параметров глаза. Дополнительно выяснить, возможно ли сократить число измерений толщины роговицы, выполняемых в различных её областях при использовании метода ИВИ. Экспериментальные данные представлены Кафедрой офтальмологии Рейнско-Вестфальского технического университета Ахена, ФРГ.

1. Описание и распределение данных. Всего в эксперименте приняло участие 34 пациента в возрасте от 42 до 93 лет, средний возраст составил 71 год и 3 месяца. У 28 участников эксперимента были измерены такие характеристики глаза, как длина переднезадней оси глаза, внутренний радиус роговицы, внешний радиус роговицы, глубина передней камеры. У 27 участников была измерена толщина

Тихонова Елизавета Олеговна – студент, Санкт-Петербургский государственный университет; e-mail: st054761@student.spbu.ru, тел.: +7(981)781-47-41

роговицы. Измерения ВГД у всех пациентов произведены в шесть последовательных моментов времени – за 300 и 30 секунд до инъекции и через 10, 120, 300 и 540 секунд после инъекции. Также была измерена методом оптической когерентной томографии (ОКТ) толщина роговицы на 3 часах и 9 часах в 5 мм от лимба у 23 участников, и ультразвуковым методом толщина роговицы в центре, слева, снизу, справа и сверху у 17 участников. Для корректной работы с данными было определено со статистической значимостью 5%, является ли распределение исследуемых признаков нормальным. Для этого использованы критерии Шапиро–Уилка, χ^2 –Пирсона и Харке–Бера. Распределение большей части признаков удовлетворило гипотезе о нормальности, кроме длины переднезадней оси глаза, ВГД через 120 и через 300 секунд после инъекции, толщины роговицы, измеренной ультразвуком за экватором слева. Это изучение структуры данных дало основание для использования как параметрических, так и непараметрических статистических методов.

2. Анализ динамики изменения ВГД. Для того, чтобы выяснить, изменялось ли внутриглазное давление в ходе эксперимента, формулируем нулевую гипотезу: значения шести последовательных измерений ВГД в среднем не отличаются друг от друга. Для её проверки используем дисперсионный анализ повторных измерений и его непараметрический аналог – критерий Фридмана. В результате в обоих случаях со статистической значимостью 0,1% отвергаем нулевую гипотезу. Затем проводим попарные сравнения с целью определения, в какие моменты времени различия в показателях ВГД статистически значимы. Для попарных сравнений используем критерий Уилкоксона с поправкой Бонферрони. В критерии Уилкоксона производится сравнение состояния больных до и после лечения, или, в нашем случае, в разные моменты времени, т. е. учитывается повторность наблюдений. Поправка Бонферрони используется для преодоления проблемы мультиколлинеарности при множественных сравнениях. Чтобы приближение было более точным, используем также поправку Йейтса на непрерывность. Получаем отсутствие попарных различий только в одной паре, а именно для измерений за 30 секунд до инъекции и через 540 секунд после инъекции. Все остальные группы измерений определяем как различные со статистической значимостью 1%. Наконец, изучаем характер различий ВГД, измеренного в последовательные моменты времени. Для этого вычисляем

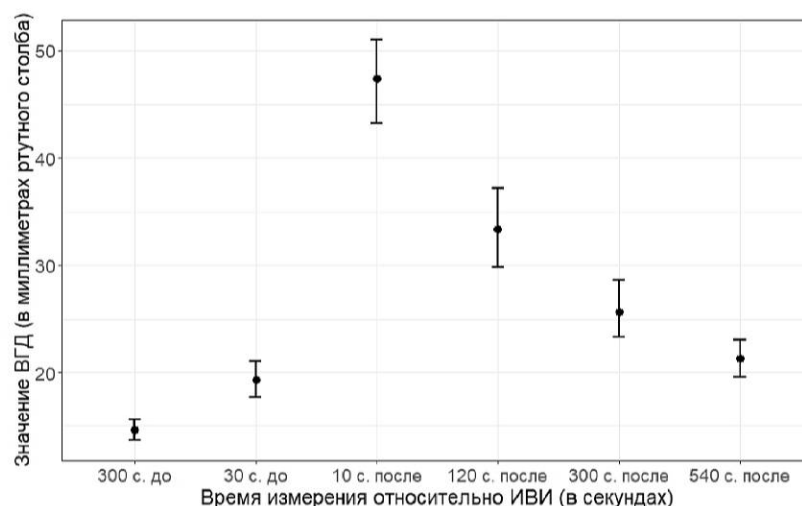


Рис. 1. Значения средних и доверительные интервалы показателей ВГД в зависимости от момента измерения

Как видно на рис. 1, наблюдается резкое повышение ВГД сразу после инъекции и последовательное снижение с возвращением в исходное состояние в течение 540 секунд (9 минут).

3. Регрессионный анализ для параметров глаз. Для выяснения возможности сокращения измерений роговицы, выполненных в различных её областях, проводим корреляционный анализ с использованием непараметрического коэффициента корреляции Спирмена, затем для тех пар переменных, для которых получена высокая корреляция со статистической значимостью 5%, строим линейные регрессионные модели и проверяем достоверность коэффициентов регрессии. Также проверяем условия использования линейной регрессии – нормальность распределения остатков (критерий Шапиро–Уилка), независимость остатков (критерий Дарбина–Уотсона) и гомоскедастичность остатков (критерий Бройша–Пагана). Для всех перечисленных критериев принимаем уровень статистической значимости 5%. В итоге выявляем наличие линейной взаимосвязи между тремя парами параметров глаза, а именно между толщиной роговицы, измеренной ОКТ на 3 часах и на 9 часах в 5 мм от лим-

ба, толщиной роговицы, измеренной ультразвуком за экватором в центре и справа, и между толщиной роговицы, измеренной ультразвуком за экватором слева и сверху.

4. Регрессионный анализ для динамики ВГД. Для того, чтобы понять, какие характеристики глаза влияют на динамику изменения внутриглазного давления, методом корреляционного анализа исследуем взаимосвязь ВГД за 300 и 30 секунд до инъекции со всеми измеренными параметрами глаза. В данном случае не получаем высокой корреляции со статистической значимостью 5% ни для одной пары, это позволяет сделать вывод о том, что ВГД до инъекции не взаимосвязано с характеристиками глаза. Затем вводим два новых показателя динамики изменения ВГД. Первый определим как разность между ВГД через 10 секунд после инъекции и средним показателем ВГД, измеренных за 300 и 30 секунд до инъекции, этот показатель назовём повышением ВГД. Второй определим как разность между ВГД через 10 секунд после инъекции и средним показателем ВГД, измеренных в последующие моменты времени после инъекции. Назовём этот показатель понижением ВГД. Исследуем взаимосвязь новых показателей с параметрами глаза методом корреляционного анализа, используя коэффициент корреляции Спирмена, затем для тех пар переменных, для которых получена высокая корреляция со статистической значимостью 5%, строим линейные регрессионные модели, проверяем значимость коэффициентов регрессии и условия использования линейной регрессии критериями, описанными в пункте 3, со статистической значимостью 5%. В результате получаем, что введенные нами показатели динамики изменения ВГД, как повышение, так и понижение, линейно зависят от глубины передней камеры и длины переднезадней оси глаза, коэффициенты при независимых переменных отрицательные.

Закключение. Проведённое статистическое исследование позволяет сделать вывод о динамике изменения внутриглазного давления. Сразу после интравитреальной инъекции происходит сильное повышение ВГД, далее давление плавно снижается и приходит к исходному состоянию в течение 540 секунд. На величину изменения ВГД влияют такие биометрические параметры глаза, как глубина передней камеры и длина переднезадней оси, чем больше по величине эти показатели, тем меньший наблюдается скачок ВГД после инъекции и тем быстрее уровень ВГД восстанавливается. Данный результат даёт

возможность для более эффективной предоперационной диагностики, взвешивания рисков, предупреждения возможных осложнений. Дополнительно выяснено, что число измерений толщины роговицы, выполненных в различных её областях, может быть сокращено на три. Например, можно проводить измерения ОКТ на 3 часах в 5 мм от лимба, ультразвуком за экватором в центре, слева и снизу. Этот результат позволяет врачам и исследователям в дальнейшем уменьшить количество затрат на проведение операции и постоперационное наблюдение.

Литература

1. Даниличев В. Ф. Обоснование и эффективность применения протеолитических ферментов при патологии глаз (клиникоэкспериментальное исследование): автореф. дис. д-р. мед. наук. Л., 1983. 24 с.
2. Бубнова И. А., Юлова А. Г. Изменение морфометрических параметров ДЗН на фоне острого повышения ВГД после интравитреальных инъекций // Национальный журнал глаукома. 2016. Т. 15. Вып. 2. С. 54–60.
3. Bakri S. J., Pulido J. S, McCannel C. A. et al. Immediate intraocular pressure changes following intravitreal injections of triamcinolone, pegaptanib, and bevacizumab // Eye. 2009. Vol. 23. P. 181–185.
4. Бойко Э. В., Сосновский С. В., Березин Р. Д. и др. Интравитреальные инъекции: теория и практика // Офтальмологические ведомости. 2010. Т. 3. Вып. 2. С. 28–35.

УДК 512.774.3

Хао Ц.

Нахождение проекций квадратичных многообразий

Рекомендовано к публикации профессором Утешевым А. Ю.

1. Введение. Есть квадратичная поверхность в n -мерном пространстве, например эллипсоид. И мы хотим найти его проекции на любую $(n - k)$ -мерную гиперплоскость ($1 \leq k \leq n - 1$) при любой размерности. Например, на произвольную плоскость. Физический смысл этой задачи – определение тени, отбрасываемой поверхностью при освещении ее световым потоком. В данной работе указаны алгоритмы для решения задач о нахождении проекций эллипсоида по направлению. Эти методы можно применить для медицинской визуализации анатомических частей (сердце, позвоночник, кровеносные сосуды и т. д.).

2. Постановка задачи. Пусть в n -мерном пространстве задано уравнение эллипсоида

$$a_{11}x_1^2 + \dots + a_{nn}x_n^2 + 2a_{12}x_1x_2 + 2a_{13}x_1x_3 + \dots + 2a_{n-1,1}x_{n-1}x_1 + 2b_1x_1 + \dots + 2b_nx_n + c = 0,$$

где $x_1, \dots, x_n \in R$. Данное уравнение можно представить следующим образом:

$$X^T AX + 2B^T X + C = 0,$$

где $X \in R^n$, а A, B являются матрицами с постоянными элементами размерности $n \times n$ и $n \times 1$, C – некоторая константа.

3. Алгоритм для нахождения проекции эллипсоида. Преежде запишем уравнение эллипсоида в неявном виде

$$\phi(x_1, \dots, x_n) = 0.$$

Хао Цянь – студент, Санкт-Петербургский государственный университет;
e-mail: st066150@student.spbu.ru, тел.: +7(965)764-77-81

Для нахождения проекции эллипсоида на любую $(n - k)$ -мерную гиперплоскость ($1 \leq k \leq n - 1$) при любой размерности будем рассматривать следующие случаи.

1. При $k = n - 1$ проекция эллипсоида на произвольную координатную ось является отрезком. Чтобы найти длину отрезка на координатную ось $x_{n-1} = 0, \dots, x_1 = 0$ исключаем последний аргумент, и тогда уравнение можно представить в виде

$$\phi(x_1, \dots, x_{n-1}, \varphi(x_1, \dots, x_{n-1})) = 0, \quad \forall x_1, \dots, x_{n-1} \in R.$$

Вычисляем частные производные по x_1, \dots, x_{n-1} и составим систему из следующих уравнений: $\frac{\partial \phi}{\partial x_1} = 0, \dots, \frac{\partial \phi}{\partial x_{n-1}} = 0$. Затем по формулам Крамера [1] найдём выражения для x_1, \dots, x_{n-1} и подставим эти выражения в исходное уравнение. После того, как эти шаги все сделали, можем получить квадратное уравнение $\phi(x_n) = 0$. Потом решаем это уравнение, соответствующее решение и есть длина проекции эллипсоида на координатную ось $x_{n-1} = 0, \dots, x_1 = 0$. Ещё заметим, что можно писать это уравнение в виде определителя, который дальше указан.

Для другого случая, где проекция на произвольную ось, исключаем соответствующий аргумент. После того, как сделали все шаги можем получить промежутки

$$x_1 \in [A_1, B_1], \dots, x_n \in [A_n, B_n].$$

Теорема. *Проекция любого n -мерного эллипсоида на произвольную прямую – это отрезок, концы которого имеют координаты, выражающиеся в виде корней квадратного уравнения с коэффициентами, полиномиально выражающимися через коэффициенты эллипсоида.*

Доказательство. Рассмотрим случай, когда ищем проекцию на координатную ось Ox_n . Очевидно, что можем представить уравнение эллипсоида в неявном виде

$$\phi(x_1, \dots, x_{n-1}, \varphi(x_1, \dots, x_{n-1})) = 0.$$

где $x_n = \varphi(x_1, \dots, x_{n-1})$. Будем считать, что функции $x_n = \varphi(x_1, \dots, x_{n-1})$ и $\psi = \phi(x_1, \dots, x_n)$ дифференцируемы, причём $\phi'_{x_n}(x_1, \dots, x_n) \neq 0$. Тогда по правилу дифференцирования сложной функции имеем

$$\frac{\partial \phi}{\partial x_k} + \frac{\partial \phi}{\partial \psi} \frac{\partial \varphi}{\partial x_k} = 0, \quad k = \overline{1, n-1}.$$

По теореме Ферма [2] знаем, что если функция в какой-то точке достигает своего максимума или минимума, причём в этой точке имеет конечную или бесконечную производную с определённым знаком, то производная в этой точке равна нулю. Таким образом, мы можем предполагать, что $\frac{\partial \varphi}{\partial x_k} = 0, k = \overline{1, n-1}$. Подставляем их в $\frac{\partial \phi}{\partial x_k} + \frac{\partial \phi}{\partial \psi} \frac{\partial \varphi}{\partial x_k} = 0, k = \overline{1, n-1}$, получаем $\frac{\partial \phi}{\partial x_k} = 0, k = \overline{1, n-1}$.

Теперь введём вспомогательную функцию следующим образом:

$$\chi = 2\phi - \left(\frac{\partial \phi}{\partial x_1} x_1 + \dots + \frac{\partial \phi}{\partial x_{n-1}} x_{n-1} \right).$$

В данный момент легко проверить, что $\deg_{x_1} \phi = \dots = \deg_{x_{n-1}} \phi = 1$, тут имеют ввиду, что старшие степени у аргументов x_1, \dots, x_{n-1} будут равными единицами.

Составим систему (пусть $\chi = 0$):

$$\begin{cases} \frac{\partial \phi}{\partial x_k} = 0, & k = \overline{1, n-1}, \\ 2\phi - \left(\frac{\partial \phi}{\partial x_1} x_1 + \dots + \frac{\partial \phi}{\partial x_{n-1}} x_{n-1} \right) = 0, \end{cases}$$

которая является линейной относительно x_1, \dots, x_{n-1} . По формулам Крамера [1] можем выразить x_1, \dots, x_{n-1} и подставить эти значения в $\chi = 0$, и это будет в виде квадратичного полинома по x_n , и этот квадратичный полином будет равен нулю. Пусть дискриминант полученного уравнения будет больше нуля, тогда можем получить два вещественных корня, как раз эти корни являются координатами на соответствующей координатной оси. В результате определитель этой системы будет выглядеть таким образом:

$$\begin{vmatrix} 1 & -2x_n & 1 & 0 & 0 & 0 & \dots & 1 \\ -2x_n & 4c & 0 & 0 & b_1 & b_2 & \dots & b_n \\ 1 & 0 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & b_1 & 0 & 0 & a_{11} & a_{12} & \dots & a_{1n} \\ 0 & b_2 & 0 & 0 & a_{12} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & b_n & 0 & 0 & a_{1n} & a_{2n} & \dots & a_{nn} \end{vmatrix}.$$

Он должен равняться нулю (система совместна).

2. При $k = n - 2$ проекция эллипсоида на плоскость является эллипсом (в частном случае эта проекция будет кругом), чтобы найти уравнение эллипса на гиперплоскости $x_1 = 0, \dots, x_{n-2} = 0$ исключим последние два аргумента, и тогда уравнение можно переписать в следующем виде:

$$\phi(x_1, \dots, x_{n-2}, \varphi(x_1, \dots, x_{n-2})) = 0, \quad \forall x_1, \dots, x_{n-2} \in R.$$

Вычисляем частные производные по x_1, \dots, x_{n-2} и составим систему из уравнений: $\frac{\partial \phi}{\partial x_1} = 0, \dots, \frac{\partial \phi}{\partial x_{n-2}} = 0$, по формулам Крамера [1] найдём выражения для x_1, \dots, x_{n-2} , затем подставим их в исходное уравнение эллипсоида и можем получить квадратное уравнение $\phi(x_{n-1}, x_n) = 0$, а полученное уравнение и есть уравнение эллипса на гиперплоскости $x_1 = 0, \dots, x_{n-2} = 0$. Заметим, что исходное уравнение эллипсоида можно представить в виде $X^T A X + 2B^T X + C = 0$. И у искомого уравнения эллипса тоже такой вид.

3. Частный случай имеет место только тогда, когда $k = n - 1$ и проекция эллипсоида будет отрезком, а для $k = \overline{1, n-2}$ проекция эллипсоида будет уравнением на соответствующую гиперплоскость. Разумеется, чтобы найти проекцию на $(n-k)$ -мерную гиперплоскость нам надо исключить $n - k$ аргументов, и найти частные производные по остальным аргументам и они должны быть равны нулю, затем составим систему из этих уравнений, потом по формулам Крамера [1] найдём выражения для этих аргументов, потом нужно подставить эти выражения в исходное уравнение эллипсоида. В конце концов получаемое выражение и есть уравнение проекции на $(n - k)$ -мерную гиперплоскость и их можно представить в виде $X^T A X + 2B^T X + C = 0$.

Замечание. После того, как нашли эти выражения, которые получены из формул Крамера [1], подставляем их в исходное уравнение эллипсоида и получаем квадратичный полином с некоторым множителем, а часть квадратичного полинома в случае $k = n - 1$ можно выразить через определитель и он должен равняться нулю (система совместна). А для $k = \overline{1, n-2}$ полученный квадратичный полином там зависит от несколько аргументов, тогда можем разложить тот полином по убывающим степеням какого-то аргумента и вычисляем дискриминант, в результате полученный дискриминант тоже можно представить в виде определителя, с учётом исключения разного аргумента, в первой строке и в первом столбце единички, которые

находятся перед b_n могут лежать в разных местах, т. е. могут перед b_1, \dots, b_n .

Пример. Найдём проекции трехмерного эллипсоида на координатную ось и на плоскость. Эллипсоид задан уравнением следующим:

$$7x^2 + 6y^2 + 5z^2 - 4xy - 4yz - 6x - 24y + 18z + 30 = 0.$$

Для нахождения проекции на координатную ось исключаем аргумент z и работаем по алгоритму.

Вычисляем частные производные

$$\frac{\partial \phi}{\partial x} = 14x - 4y - 6, \quad \frac{\partial \phi}{\partial y} = -4x + 12y - 4z - 24.$$

По формулам Крамера [1] получим

$$x = \frac{16z + 168}{152}, \quad y = \frac{56z + 360}{152}.$$

Подставим x и y в исходное уравнение:

$$\phi(z) = \frac{81}{19}z^2 + \frac{162}{19}z - \frac{33}{19} = 0.$$

Решив уравнение, получим ограничения

$$z_{max} = 0,1863420280, \quad z_{min} = -2,186342028.$$

Аналогично делается для $x = \varphi(y, z)$ и $y = \varphi(x, z)$. В конце получим полные ограничения в виде:

$$x \in [0,187; 1,981], \quad y \in [0,861; 3,136], \quad z \in [0,186; -2,186].$$

Теперь ищем проекцию эллипсоида на плоскость Oxy , вычислим

$$\frac{\partial \phi}{\partial z} = 10z - 4y + 18.$$

Пусть $\frac{\partial \phi}{\partial z} = 0$, отсюда $z = \frac{4y-18}{10}$, подставляем это в исходное уравнение, получим

$$7x^2 + \frac{26}{5}y^2 - \frac{84}{5}y - 4xy - 6x + \frac{69}{5} = 0.$$

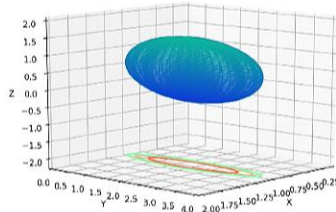


Рис. 1. Проекция эллипсоида

Полученное уравнение будет уравнением эллипса, которую эллипсоид проецировал на плоскость Oxy .

4. Нахождение проекций эллипсоида по направлению.

Рассмотрим теперь уравнение эллипсоида в матричном виде

$$X^T A X + 2B^T X + C = 0.$$

Есть в системе координат параллельные прямые, которые параллельны вектору $\Xi = (\xi_1, \dots, \xi_n)^T$ (вектор Ξ в начале дан), и они выглядят следующим образом (где t является некоторым действительным числом):

$$\begin{cases} x_1 = x_1^0 + t\xi_1, \\ \dots \\ x_n = x_n^0 + t\xi_n. \end{cases}$$

И их можно представить в векторном виде $X = X_0 + t\Xi$. Составим систему из уравнения эллипсоида и параллельных прямых, чтобы найти все пересечения между параллельными прямыми и эллипсоидом

$$\begin{cases} X^T A X + 2B^T X + C = 0, \\ X = X_0 + t\Xi. \end{cases}$$

Вычислим эту систему и разложим это уравнение по убывающим степеням t и оно будет выглядеть следующим образом:

$$t^2(\Xi^T A \Xi) + t(X_0^T A \Xi + \Xi^T A X_0 + 2B^T \Xi) + (X_0^T A X_0 + 2B^T X_0 + C) = 0.$$

Вычисляем дискриминант этого уравнения, получим

$$(X_0^T)^2 A^2 \Xi^2 + (\Xi^T)^2 A^2 X_0^2 + 4(B^T)^2 \Xi^2 + 2X_0^T A \Xi \Xi^T A X_0 + 4X_0^T A \Xi B^T \Xi + 4\Xi^T A X_0 B^T \Xi - 4\Xi^T A \Xi X_0^T A X_0 - 8\Xi^T A \Xi B^T X_0 - 4C = 0.$$

Теперь начнём искать проекцию эллипсоида по направлению на любую $(n - k)$ -мерную гиперплоскость ($1 \leq k \leq n - 1$).

Проекция на координатную ось. Сначала рассмотрим частный случай, это когда ищем проекцию на прямую, соответственно, в этом случае $k = n - 1$, и исключаем соответственные $(n - 1)$ элементов $x_\alpha^0 = 0$, $\alpha \in \overline{1, n}$, в векторе X_0 и подставим его в уравнение дискриминанта и можем получить квадратное уравнение $\phi(t, x_\beta^0) = 0$. Для того чтобы между осью и эллипсоидом были только два пересечения, требуется, чтобы дискриминант (по t) данного уравнения был больше нуля, решение дискриминанта и есть искомая проекция на соответствующую координатную ось.

Пример. Найдём проекцию трехмерного эллипсоида на ось ox по направлению $u = (-1, -1, 0)$. Уравнение эллипсоида задано следующим образом:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}^T \begin{pmatrix} 7 & -2 & 0 \\ -2 & 6 & -2 \\ 0 & -2 & 5 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2 \begin{pmatrix} -3 \\ -12 \\ 9 \end{pmatrix}^T \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 30 = 0.$$

И параллельные прямые по направлению выглядят следующим образом:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x^0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -t \\ -t \\ 0 \end{pmatrix}.$$

Подставляем в уравнение дискриминанта, чтобы вычислить проекцию, получим

$$-152x_0^2 - 384x_0 - 180 = 0.$$

В конце ограничение на ось Ox будет

$$x_0 \in \left[-\frac{24}{19} - \frac{3\sqrt{\frac{33}{2}}}{19}, \frac{3\sqrt{\frac{33}{2}}}{19} - \frac{24}{19} \right].$$

Однако, не всегда можем успешно найти проекцию по направлению на координатные оси, это не значит, что не существует проекции, а значит, что искомая проекция находится на комплексной плоскости.

Проекция на плоскость. В этом случае проекция будет уравнением эллипса, разумеется, что нам надо исключить соответственные $(n - 2)$ элементов $x_\alpha^0 = 0$, $\alpha \in \overline{1, n}$, в векторе X_0 и подставим его в уравнение дискриминанта и можем получить квадратное уравнение $\phi(t, x_{\alpha_1}^0, x_{\alpha_2}^0) = 0$, где α_1 и α_2 являются некоторыми индексами. Полученное уравнение и есть искомое уравнение эллипса.

Пример. Найдём проекцию трехмерного эллипсоида на плоскость $z = 0$ по направлению $u = (-1, -1, 0)$. Уравнение эллипсоида задано следующим образом:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}^T \begin{pmatrix} 7 & -2 & 0 \\ -2 & 6 & -2 \\ 0 & -2 & 5 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 2 \begin{pmatrix} -3 \\ -12 \\ 9 \end{pmatrix}^T \begin{pmatrix} x \\ y \\ z \end{pmatrix} + 30 = 0.$$

И параллельные прямые по направлению выглядят следующим образом:

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} x^0 \\ y^0 \\ 0 \end{pmatrix} + \begin{pmatrix} -t \\ -t \\ 0 \end{pmatrix}.$$

Подставляем в уравнение дискриминанта чтобы вычислить проекцию, получим

$$\begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} -243 & 96 \\ 96 & -152 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + 2 \begin{pmatrix} 18 \\ 192 \end{pmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix} - 180 = 0.$$

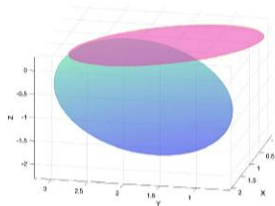


Рис. 2. Проекция эллипсоида по направлению

Проекция на гиперплоскость. Заметим, что проекция на гиперплоскость будет уравнением эллипса, разумеется, что нам надо исключить соответственные элементы $x_\alpha^0 = 0$, $\alpha \in \overline{1, n}$, в векторе X_0 и подставим его в уравнение дискриминанта и можем получить квадратное уравнение $\phi(t, x_{\alpha_1}^0, \dots, x_{\alpha_\gamma}^0) = 0$, где $\alpha_1, \dots, \alpha_\gamma$ являются некоторыми индексами. Полученное уравнение и есть искомое уравнение эллипса.

5. Заключение. Для задачи нахождения проекции эллипсоида (по направлению и без) существуют разные способы, в данной работе использовался алгебраический метод и удалось вычислить эти проекции.

Литература

1. Утешев А. Ю., Калинина Е. А. Лекции по высшей алгебре. Часть II. Учеб. пособие. СПб.: «СОЛО», 2007. 279 с.
2. Кудрявцев Л. Д. Курс математического анализа. Т. 2. М.: Наука, 1988. 576 с.

УДК 51-76

Чубаров Д. А.

Математическое моделирование межпозвонкового диска с использованием численного метода

Рекомендовано к публикации профессором Трегубовым В. П.

1. Описание математической модели. В качестве модели межпозвонкового диска предлагается рассмотреть плоский цилиндр, разделенный на трапецевидные ячейки радиальными и «хордовыми» стенками (рис. 1, 2) [1]. Причем центральная часть, моделирующая пульпозное ядро, исключается из деления и дальнейших расчетов.

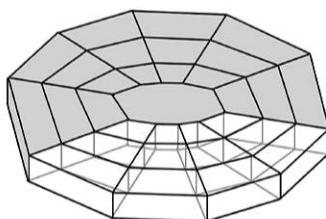


Рис. 1. Принцип деления на ячейки

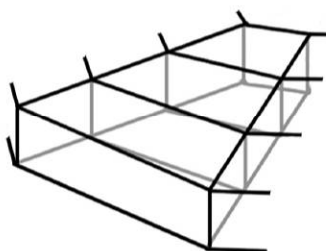


Рис. 2. Радиальный фрагмент модели

Чубаров Дмитрий Александрович – студент, Санкт-Петербургский государственный университет; e-mail: st050417@student.spbu.ru, тел.: +7(999)532-39-07

Ячейки будем считать заполненными несжимаемой жидкостью. С учетом наличия постоянной нагрузки на позвоночный столб и сжатия дисков межпозвоноковыми связками [2], давление P_0 внутри ячеек превосходит атмосферное. Поэтому в начальный момент «хордовые» стенки ячеек выгнуты в противоположную от центра диска сторону. Форма изгиба задается синусоидой, согласно метода Релея:

$$\Delta_{ij} \sin \left(x \frac{\pi}{h_{ij}(\alpha)} \right),$$

где Δ_{ij} и $h_{ij}(\alpha)$ – высота прогиба и высота ячейки $\{i, j\}$ соответственно, i – порядковый номер кольца, j – номер ячейки в кольце (рис. 3).

2. Описание алгоритма. Идея алгоритма заключается в том, что давление внутри ячейки P_{ij} в каждый момент времени полагается пропорциональным высоте прогиба Δ_{ij} .

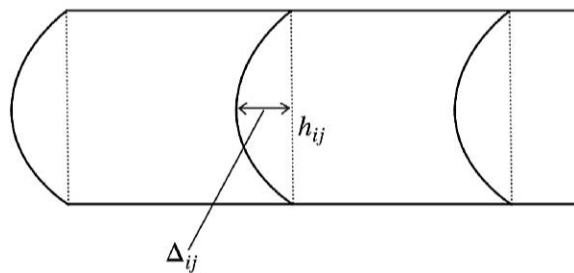


Рис. 3. Высота прогиба и высота ячейки

2.1. Предварительные расчеты. В первую очередь, необходимо рассчитать исходные объемы V_{ij} и высоты прогибов стенок ячеек Δ_{ij} . Так как высота прогиба принята пропорциональной давлению, то

$$\Delta_{ij} = \frac{P_{ij}}{K},$$

где K – коэффициент жесткости стенок.

Объемы ячеек вычисляются согласно формуле

$$V_{ij} = S_{ij} h_{ij}(\alpha) + V_{ij}^p - V_{i-1,j}^p,$$

где

$$S_{ij} = \frac{1}{2} \sin(\phi) (r_{ij}(\alpha)^2 - r_{i-1,j}(\alpha)^2)$$

– площадь верхней грани ячейки, r_{ij} – радиус ячейки (расстояние от центра диска до большего основания грани ячейки), ϕ – шаг радиального деления,

$$V_{ij}^p = a_{ij} \int_0^{h_{ij}} \Delta_{ij} \sin\left(x \frac{\pi}{h_{ij}(\alpha)}\right) dx = a_{ij} \frac{2\Delta_{ij} h_{ij}(\alpha)}{\pi}$$

– объем выпуклости (рис. 4).

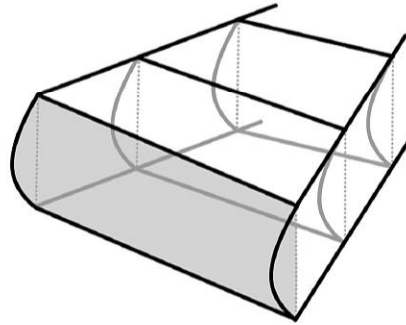


Рис. 4. Объем выпуклости ячейки (окрашенная серым часть)

2.2. Поворот пластин и описание процесса. Модель имеет одну степень свободы – угол между гранями диска и положительным направлением горизонтальной оси. В систему в течение некоторого времени вносится возмущение в виде изменения угла наклона α (рис. 5) от $\alpha_0 = 0$ до заданного предельного значения α_{lim} по линейному закону.

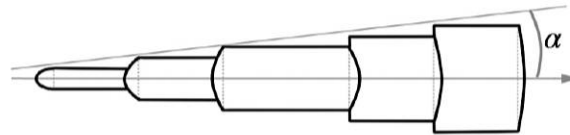


Рис. 5. Вид модели (в сечении) после поворота пластин

Каждый шаг алгоритма можно разбить на этапы:

1. Если время работы алгоритма не превышает T , происходит пересчет угла:

$$\alpha = \alpha_{k+1} = \alpha_k + \frac{\alpha_{lim}}{T}.$$

2. Расчет изменения объемов за счет диффузии:

$$\Delta V_{ij} = D (P_{i-1,j} + P_{i+1,j} - 2P_{ij}) + d (P_{i,j-1} + P_{i,j+1} - 2P_{ij}),$$

где D – коэффициент межкольцевой диффузии, d – коэффициент внутрикольцевой диффузии.

3. Пересчет объемов:

$$V_{ij}^{new} = V_{ij} + \Delta V_{ij}.$$

4. Пересчет Δ_{ij} :

$$\Delta_{ij}^{new} = (V_{ij}^{new} - S_{ij} h_{ij}(\alpha)) \frac{\pi}{2h_{ij}(\alpha)a_{ij}},$$

где a_{ij} – величина большего основания трапеции в верхней грани ячейки.

5. Пересчет давлений:

$$P_{ij}^{new} = K \Delta_{ij}^{new}.$$

Процесс продолжается до тех пор, пока разность давлений между всеми ячейками будет не больше заданного ε .

3. Результаты работы алгоритма. Для численного эксперимента с использованием предложенного алгоритма выбраны следующие параметры модели:

- коэффициент внутрикольцевой диффузии $d = 0,001$;
- коэффициент межкольцевой диффузии $D = 0,0002$;
- коэффициент жесткости стенок ячеек $K = 11,7$;
- исходная толщина диска $H = 7$ мм;
- исходный радиус диска $R = 20$ мм;

- начальное внутреннее давление $P_0 = 101$ кПа;
- количество колец $I = 5$;
- количество ячеек в каждом кольце $J = 25$;
- предельное значение угла наклона $\alpha_{lim} = 10^\circ$;
- время (количество итераций) внесения возмущения $T = 1000$.

На рис. 6, 7 представлены графики динамики давлений внутри двух наугад выбранных ячеек.

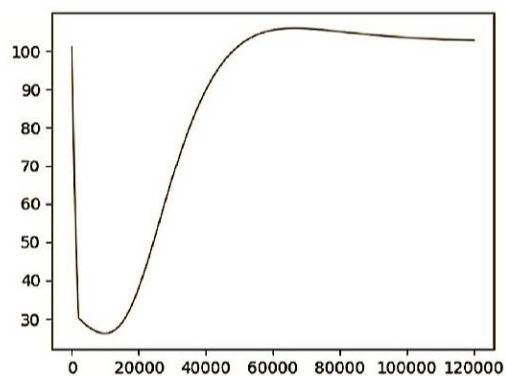


Рис. 6. Динамика давлений в ячейке №1

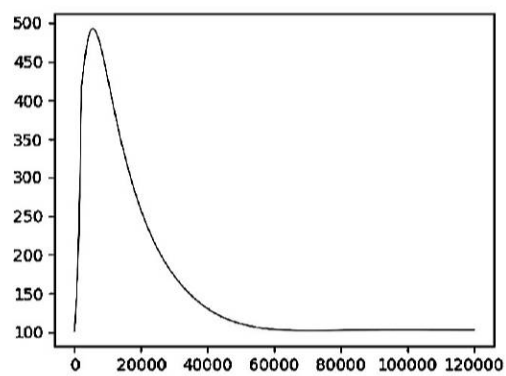
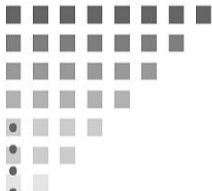


Рис. 7. Динамика давлений в ячейке №2

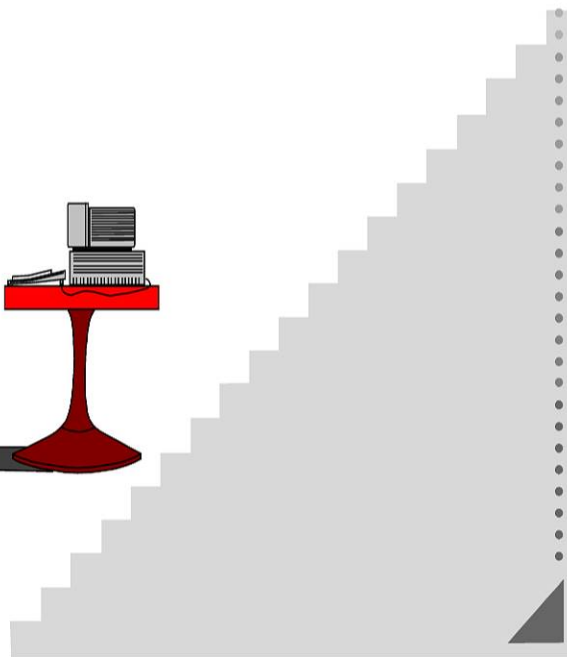
4. Заключение. Построенная модель межпозвонкового диска позволяет вычислить время стабилизации внутреннего давления при динамическом изменении угла наклона пластин. Ключевой ее особенностью является то, что она отражает внутреннюю структуру реального диска и биологические процессы, вызываемые его деформацией.

Литература

1. Капанджи А. И. Позвоночник. Физиология суставов: схемы механики человека с комментариями. Т. 3. М.: Эксмо, 2009. 344 с.
2. Донник А. М., Иванов Д. В., Киреев С. И. и др. Извлечение клинически значимых данных из биомеханического моделирования вариантов хирургического лечения травмы позвоночника при повреждении позвонков ТН10, ТН11 // Известия Саратовского университета. Математика, механика, информатика. Т. 19. 2019. С. 439–453.



*4. Информационные
и компьютерные
технологии*



УДК 004.855.5

Акопян А. С.

Улучшение методов получения выравнивания слов в задаче нейронного машинного перевода

Рекомендовано к публикации доцентом Козыменко В. А.

1. Введение. Задача выравнивания заключается в поиске лексических эквивалентов для параллельного набора предложений и долгое время была одной из фундаментальных задач в области обработки естественного языка. Хотя получение выравнивания в нейронном машинном переводе более не является внутренней подзадачей, в отличие от статистического машинного перевода, в последние годы исследование выравниваний для нейронных моделей стало вновь интересной задачей для исследователей. Выравнивание слов дает ключ к пониманию черного ящика нейронного машинного перевода. Оно применимо во множестве других задач, таких как анализ ошибок перевода [1] или перевод с внешним заданным словарем для исправления перевода редко встречаемых слов [2]. Практическим применением выравниваний может быть перевод документов и веб-сайтов с сохранением ссылок и соблюдением стилей, используемых на сайте.

2. Постановка задачи. Имея предложения на исходном и целевом языках $f_1^J = f_1, \dots, f_s, \dots, f_J$ и $e_1^I = e_1, \dots, e_t, \dots, e_I$ соответственно, необходимо получить выравнивание, также называемое жестким выравниванием и определяемое как множество пар:

$$A = \{(t, s) \mid t \in \{1, \dots, I\}, s \in \{1, \dots, J\}\}.$$

Наличие пары (t, s) во множестве A означает, что слово f_s исходного предложения является лексическим эквивалентом слова e_t из целевого предложения. Для оценки качества выравнивания используется метрика AER (Alignment Error Rate) [3].

3. Схема решения. В данном разделе описаны архитектура и используемой модели и предложенное решение.

Акопян Алексей Степанович – студент, Санкт-Петербургский государственный университет; e-mail: st062283@student.spbu.ru, тел.: +7(965)815-80-19

3.1. Модель Transformer. Сразу после публикации статьи [4], модель Transformer установила новый стандарт качества в области машинного перевода. Архитектура модели Transformer состоит из двух частей: кодирующей и декодирующей, и основана исключительно на механизме внимания, что позволяет эффективно распараллелить вычисления. Обе части состоят из нескольких слоев, каждый из которых содержит подслой множественного внутреннего внимания и подслой прямого распространения, а декодирующая часть модели также имеет подслой множественного кросс-внимания, позволяющий ей выделять слова исходного предложения наиболее значимые при генерации следующего слова. На вход механизму внимания подаются множество Q векторов запросов и множество пар (K, V) векторов ключей и векторов значений. Выход механизма внимания может быть записан в виде

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V,$$

где d_k обозначает размерность векторов ключей. В задаче машинного перевода за Q принимают множество скрытых состояний декодирующей части, а за K и V – множество выходов кодирующей части. Вместо использования одного блока внимания авторы решили использовать множественное внимание с количеством n внутренних блоков, которое может быть представлено в следующей форме:

$$\text{MultiHead}(Q, K, V) = \text{concat}(h_1, \dots, h_n)W^O,$$

где

$$h_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V).$$

Элементы матриц линейных преобразований $W^O \in \mathbb{R}^{d \times d}$, $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d \times \frac{d}{n}}$ являются обучаемыми параметрами модели.

3.2. Усредненное внимание. При выведении выравниваний для нейронных моделей бóльший интерес представляют не выходы механизма внимания, а матрицы весов, часто называемые активациями внимания,

$$W_{l,i} = \text{softmax}\left(\frac{QW_i^Q(KW_i^K)^T}{\sqrt{d_k}}\right),$$

где l обозначает номер слоя, а i – номер внутреннего блока множественного внимания. Наиболее простой способ получения мягкого выравнивания S из активаций блоков внимания состоит в их усреднении по блокам и по слоям. Такую матрицу можно интерпретировать как вероятностное распределение выравнивания над всеми исходными словами для каждого целевого слова. Получить жесткое выравнивание A можно путем выбора для каждого целевого слова индекса элемента с наибольшим значением в соответствующей ему строке матрицы S . Несмотря на то, что этот метод является простым в реализации, он не дает результат требуемого качества. Главная проблема выведения выравнивания из активаций механизмов внимания заключается в том, что данная матрица описывает отношения между скрытыми состояниями декодирующей части и выходами кодирующей, а не словами исходного и целевого предложений. В работах [5, 6] выравнивания получали путем усреднения активаций блоков внимания на одном определенном слое. В данной работе предложено помимо выбора слоя делать выбор блока множественного внимания и будет показано, что выбор блока в некоторых случаях может улучшить качество выравнивания. Такое предположение может быть аргументировано тем, что при обучении минимизируется функция потерь, соответствующая задаче перевода. Поэтому необязательно, что выравнивание, полученное в результате усреднения активаций блоков внимания, даст лучше результат, чем выравнивание, извлеченное из активаций одного блока. Вполне возможно, что при решении задачи перевода один из блоков взял на себя роль выравнивающего элемента в модели. Для подтверждения гипотезы предложено сравнить качество выравнивания для трех методов в случае усреднения по блокам одного слоя и в случае выбора лучшего блока.

3.3. Критерий выбора блока внимания. Для выбора блока внимания предлагается применять критерий, используемый в работе [6] для выбора слоя. Имея прямую $M_{f \rightarrow e}$ и обратную $M_{e \rightarrow f}$ обученные модели перевода, предполагается, что выравнивания, полученные данными моделями, будут совпадать в некоторой степени. Поэтому для выбора лучшего блока внимания предлагается посчитать оценки AER на валидационной выборке для каждого блока прямой модели, в качестве правильных ответов приняв выравнивания, полученные от блоков обратной модели. При этом стоит также оценивать и усредненное выравнивание, чтобы удостовериться, что выбор блока действительно улучшает качество. В итоге получается

$((n + 1) \times (n + 1))$ -матрица оценок, из которой необходимо выбрать элемент с наименьшим значением. Номер строки и номер столбца этого элемента будут определять лучший блок внимания в прямой и обратной моделях соответственно.

4. Анализ результатов. Для сравнения способов получения выравнивания, следуя статье [1], с помощью пакета fairseq-пу [7] обучены две нейронные модели перевода с немецкого на английский и обратно с конфигурациями `transformer_iwslt_de_en`. Обучающие и тестовые данные выбраны, следуя работе [8]. Предоставляемый авторами статьи код [9] использован для подготовки данных и при симметризации выравниваний [10], полученных от двух моделей. Набор обучающих данных включает 1,9 миллиона пар предложений на немецком и английском языках. За отсутствием валидационной выборки отложена 1 тысяча случайных пар предложений. Для оценки качества выравнивания использовался набор из 508 пар предложений, который также содержит выравнивания, выполненные людьми-экспертами. В данной работе сравнивались выравнивания, полученные для прямого и обратного направлений, а также выравнивания, полученные в результате симметризации двух направлений.

4.1. Метод NAIVE-ATT. В работе [5] указано, что усреднение по блокам предпоследнего слоя дает лучше выравнивание, чем в случае усреднения по слоям и по блокам. Из результатов, представленных в таблице 1, видно, что выбор блока не всегда улучшает результат. Оценка AER была исправлена только для выравнивания, полученного прямой моделью.

Таблица 1. Оценки AER на тестовой выборке для метода NAIVE-ATT

NAIVE-ATT	de→en	en→de	bidir
mean	37,1	48,2	31,3
best	36,1	56,0	34,9

4.2. Метод SHIFT-ATT. В статье [6] отмечено, что интерпретация элемента $S_{t,s}$ как меры близости между входным словом f_s и предыдущим сгенерированным целевым словом e_{t-1} показывает результат лучше, особенно на нижних слоях. Исходя из результатов, приведенных в таблице 2, видно, что выбор лучшего блока в данном методе существенно улучшил качество выравнивания во всех трех случаях. Причем, как видно из таблицы 5, хорошо справляется с

выравниванием лишь один блок из четырех, в то время как другие три вносят шум в матрицу выравнивания.

Таблица 2. Оценки AER на тестовой выборке для метода SHIFT-ATT

SHIFT-ATT	de→en	en→de	bidir
mean	25,9	28,0	19,7
best	17,9	23,2	17,0

Результаты, полученные в данной статье, несколько отличаются от результатов, приведенных в [6], что может быть обосновано различиями в объеме словаря и выбранных гиперпараметрах при обучении.

4.3. Метод SHIFT-AET. В статье [6] также предложен метод SHIFT-AET (Alignment-Enhanced Transformer), который заключается в присоединении к основной модели Transformer дополнительного выравнивающего модуля, представляющего собой множественное внимание с несколькими блоками. Выравнивание для этого метода получается путем усреднения активаций блоков, входящих в дополнительный модуль. В качестве функции потерь на этапе обучения использовалась перекрестная энтропия, а правильными ответами считались выравнивания, полученные методом SHIFT-ATT с последующим применением к ним алгоритма симметризации. После сравнения оценок AER, представленных в таблице 3, выявлено, что во всех трех случаях выбор блока ухудшает качество выравнивания. Исходя из этого, можно прийти к выводу, что возможное улучшение, вносимое выбором блока внимания, нивелируется для дополнительного модуля используемой при его обучении функции потерь, специально подобранной для решения задачи выравнивания. Изменение начальных параметров дополнительного модуля (выбор различного количества блоков во множественном внимании, инициализация блоков дополнительного модуля весами блоков внимания основной модели) почти не сказываются на оценке AER.

Таблица 3. Оценки AER на тестовой выборке для метода SHIFT-AET

SHIFT-AET	de→en	en→de	bidir
mean	15,6	19,2	15,5
best	15,8	19,7	15,8

4.4. Критерий выбора блока внимания. Чтобы удостовериться, что предложенный выше критерий действительно выбирает

правильный блок внимания, сначала выбраны номера блоков на основе критерия на валидационной выборке, а затем на основе оценок AER на тестовом множестве.

Таблица 4. Критерий выбора блока внимания

	de→en	mean	1	2	3	4
en→de						
mean		41,4	83,3	85,1	78,9	35,3
1		76,8	85,1	89,5	82,6	76,1
2		81,7	87,8	90,6	91,6	80,0
3		70,1	88,3	90,8	83,8	67,9
4		38,5	83,4	83,9	78,4	30,9

Таблица 5. Оценки AER на тестовой выборке

№ блока	mean	1	2	3	4
en→de	25,9	75,4	77,8	72,5	17,9
de→en	28,0	71,1	76,5	63,8	23,2

Сравнение результатов, приведенных в таблицах 4, 5, подтверждает, что данный критерий способен выбирать лучший блок для получения выравнивания.

5. Заключение. Результаты экспериментов подтверждают, что для получения выравнивания в задаче нейронного машинного перевода помимо выбора слоя необходимо также выбирать блок внимания, из активаций которого будет получено выравнивание.

Литература

1. Ding S., Xu H., Koehn P. Saliency-driven word alignment interpretation for neural machine translation // Proceedings of the Fourth Conference on Machine Translation. 2019. Vol. 1. P. 1–12.
2. Alkhouli T., Bretschner G., Ney H. On the alignment problem in multi-head attention-based neural machine translation // Proceedings of the Third Conference on Machine Translation. 2018. P. 177–185.
3. Och F. J., Ney H. Improved statistical alignment models // Proceedings of the 38th Annual Meeting on Association for Computational Linguistics. 2000. P. 440–447.

4. Vaswani A., Shazeer N., Parmar N. et al. Attention is all you need // Advances in Neural Information Processing Systems. 2017. Vol 30. P. 6000–6010.
5. Garg S., Peitz S., Nallasamy U., Paulik M. Jointly learning to align and translate with transformer models // Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2019. P. 4453–4462.
6. Chen Y., Liu Y., Chen G. et al. Accurate word alignment induction from neural machine translation // Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing and the 10th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). 2020. P. 566–576.
7. GitHub. Fairseq-py [Электронный ресурс]: URL:<https://github.com/pytorch/fairseq> (дата обращения 20.03.2021).
8. Zenkel T., Wuebker J., DeNero J. Adding interpretable attention to neural translation models improves word alignment [Электронный ресурс]: URL:<https://arxiv.org/abs/1901.11359> (дата обращения: 20.03.2021).
9. GitHub. Alignment-scripts [Электронный ресурс]: URL:<https://github.com/lilt/alignment-scripts> (дата обращения 20.03.2021).
10. Koehn P., Axelrod A., Mayne A. B. et al. Edinburgh system description for the 2005 IWSLT speech translation evaluation // International Workshop on Spoken Language Translation (IWSLT). 2005. P. 68–75.

УДК 004.056.53

Афанасов А. К., Цой А. И.

Извлечение данных Android-приложения WhatsApp

1. Введение. Экспертам в области цифровой криминалистики необходимы инструменты для извлечения данных из различных источников. В частности, в качестве таких источников могут выступать Android-устройства и облака. Компания «Белкасофт» занимается разработкой инструмента (Belkasoft X), одной из задач которого является извлечение данных из облачных хранилищ. Поставлена задача – исследовать возможности извлечения данных приложения WhatsApp для платформы Android.

В данной работе исследован алгоритм генерации ключа расшифровки резервных копий данных приложения WhatsApp платформы Android. Это позволяет реализовать «облачный» метод извлечения данных, в котором не нужен доступ к устройству. Доступ к устройству необходим в случае другого метода – «физического», где применяются, например, понижение версии приложения или создание слепка файловой системы устройства. Приложение WhatsApp создаст зашифрованные резервные копии данных на устройстве и дублирует их в облако (Google Drive для платформы Android). Выгружать данные возможно как при наличии, так и при отсутствии Android-устройства. Для расшифровки данных используется key-файл, хранящийся в приватной области памяти файловой системы Android. Этот файл возможно получить двумя способами: либо выгрузить из устройства, либо сгенерировать. Генерация ключа – это один из этапов «облачного» метода извлечения данных, содержащего следующие шаги:

- выгрузка зашифрованных баз данных из Google Drive. Требуется логин/пароль от учетной записи Google Drive. В рамках настоящей работы логин/пароль являются заранее известными;

Афанасов Артем Константинович – студент, Санкт-Петербургский государственный университет; e-mail: afanassov.artiom@gmail.com; тел. +7(812)926-64-74

Цой Артур Игоревич – инженер-программист, ООО «Белкасофт»; e-mail: at@belkasoft.com; тел. +7(812)926-64-74

- генерация ключа расшифровки путем имитации запросов к серверу WhatsApp. Требуется SMS-код, отправленный на привязанный номер телефона;
- расшифровка базы данных.

В статье [1] можно увидеть, что имеет расшифрованная база данных *msgstore.db* в таблицах *messages* и *chat_list*, содержащих разнообразную информацию для анализа. В актуальной базе в интересующую таблицу *messages* на данный момент входит большее число атрибутов. Полезную информацию содержат следующие:

- *media_caption* (заголовок отправленного сообщения);
- *read_device_timestamp* (время прочтения сообщения на устройстве);
- *played_device_timestamp* (время воспроизведения сообщения);
- *mentioned_jids* (упомянутые в сообщении аккаунты).

Аналогично с таблицей *chat_list*:

- *subject* (название беседы);
- *creation* (время создания);
- *last_read_receipt_send_message_table_id* (порядковый номер записи в таблице *messages*, соответствующей последнему прочитанному сообщению).

2. Обзор. В открытом доступе имеются проекты, решающие следующие задачи:

- WhatsApp-GD-Extractor-Multithread [2] – выгрузка зашифрованных данных из Google Drive без расшифровки;
- WhatsApp-Key-DB-Extractor [3] – извлечение данных и ключа расшифровки из устройства;
- WhatsApp-Струт12-Decrypter [4] – расшифровка баз данных по имеющемуся ключу.

Существует проект, созданный для использования WhatsApp через консоль – *yowsup* [5]. Этот проект был полезен для генерации параметров для запросов к серверу WhatsApp.

Для получения доступа к данным приложения WhatsApp без использования Android-устройства не хватает самого главного – ключа

расшифровки. Однако проектов с открытым доступом, осуществляющих генерацию ключа расшифровки для актуальных версий WhatsApp, не существует (можно найти старый проект по генерации WhatsApp-Key-Generator [6], который уже не актуален и не способен генерировать ключ для расшифровки настоящих баз данных). Поэтому необходимо изучать работу самого приложения, проводя обратную разработку. Для этого использовались следующие инструменты:

- анализатор трафика Burp Suite Community использовался для атаки посредника;
- декомпилятор jadx использовался для реверс-инжиниринга APK WhatsApp;
- библиотека Cydia Substrate использовалась для подмены вызовов методов приложения в iOS-версии WhatsApp;
- система сборки theos использовалась для iOS-версии приложения WhatsApp.

3. Генерация ключа расшифровки базы данных. Возможны следующие способы появления ключа на устройстве: получение ключа в чистом или зашифрованном виде от сервера, генерация ключа на устройстве. Исследование генерации ключа в данной работе показало, что WhatsApp генерирует ключ на устройстве. В начале работы приложения WhatsApp скачивается зашифрованная база данных. Далее ключ расшифровки генерируется на устройстве: часть ключа генерируется самостоятельно, часть ключа берется из зашифрованной базы, часть получается от сервера.

3.1. Реверс-инжиниринг WhatsApp. *Изучение файлов приложения.* Необходимость общения с сервером WhatsApp показало изучение файловой системы Android-устройства с правами суперпользователя: ключ создается после загрузки баз данных из Google Drive.

Вызывают интерес следующие файлы приложения WhatsApp:

- `/data/data/com.whatsapp/`
 - `databases/msgstore.db` (не зашифрованная база данных);
 - `files/key` (ключ расшифровки);
- `.../notPrivateMemory/WhatsApp/Databases/`
 - `msgstore.db.crypt12` (зашифрованная база данных).

Анализ трафика. Атакой посредника с использованием Burp Suite Community был перехвачен ряд HTTPS-запросов. Из-за сложностей расшифровки XMPP-пакетов с помощью анализаторов трафика приложение WhatsApp необходимо было модифицировать с целью логирования информации XMPP-общения между клиентом и сервером.

Перехваченные HTTPS-запросы:

- три запроса на регистрацию клиента на сервере WhatsApp <https://v.whatsapp.net>:
 - `/v2/exist?ENC=EncryptedQueryStringParameters1;`
 - `/v2/code?ENC=EncryptedQueryStringParameters2;`
 - `/v2/register?ENC=EncryptedQueryStringParameters3.`
- запросы на выгрузку базы данных из Google Drive с сервера <https://backup.googleapis.com>:
 - `/v1/clients/wa/backups/digitsOfPhone?mode=restore;`
 - `/v1/clients/wa/backups/digitsOfPhone/files?mode=restore&pageSize=5000.`

Декомпиляция APK. С использованием декомпилятора jadx была исследована структура key-файла. В ходе исследования оказалось, что для расшифровки базы данных достаточно только последнего блока из 32 байт key-файла. В структуру ключа входят: константная метаинформация о файле ключа расшифровки; версия шифрования; версия ключа; байты из зашифрованной базы данных, которые используются для получения последней части ключа по XMPP; последовательность байт из нулей; байты, генерируемые после XMPP-общения с сервером WhatsApp.

При декомпиляции приложения был обнаружен код с названиями версий `.crypt12`, `.crypt13`, `.crypt14`. На данный момент WhatsApp для шифрования использует версию под названием `.crypt12`. В данной версии используется алгоритм шифрования AES-GCM.

Логирование iOS-версии WhatsApp. Для логирования XMPP-общения была модифицирована iOS-версия WhatsApp (модифицировать Android-версию невозможно по причине проверки подписи APK) с использованием Cydia Substrate и theos. Логирование показало, что клиент iOS в ходе XMPP-общения получает от сервера пакет, длина поля данных которого в байтах совпадает с количеством

байт последнего блока key-файла. Также в этих запросах фигурировали данные о версии ключа, совпадающие с теми, что имеются в key-файле. Из этого следует, что этот запрос должен предоставить последние данные для генерации ключа расшифровки. Следующим этапом продолжилось исследование декомпилированного APK WhatsApp с целью поиска генерации параметров XMPP-запроса.

Декомпиляция APK для поиска XMPP-параметров. Декомпилированный код APK показал, что параметры XMPP-запроса на получение данных ключа расшифровки, передаваемые от сервера, генерируются по зашифрованной базе данных. Таким образом, чтобы расшифровать базу данных, необходимо иметь данную зашифрованную базу и имитировать XMPP-запрос к серверу.

3.2. Имитация WhatsApp-клиента. Для того чтобы сгенерировать ключ расшифровки, требуется имитировать HTTPS-запросы на регистрацию (см. рис. 1) и XMPP-запрос на получение данных ключа (см. рис. 2). Для имитации XMPP-запроса сначала необходимо задать и получить идентификационные данные аккаунта. Они генерируются при регистрации на сервере WhatsApp. Способы генерации параметров для регистрации клиента были найдены в проекте *yowsup*.

Имитация клиента для регистрации. Сначала на сервер вместе с запросом на регистрацию отправляется информация о номере телефона, идентификационные параметры аккаунта, служебные параметры (см. проект *yowsup* [5]).

В случае корректно предоставленных данных сервер отправляет SMS-код для регистрации на телефонный номер клиента. Далее клиент для завершения регистрации отправляет полученный код вместе с идентификационной информацией о себе. При завершении регистрации сервер высылает подтверждение и дополнительный идентификационный параметр.

Пример запросов на окончание регистрации с передачей SMS-кода выглядит следующим образом.

- URI-строка для запроса клиента
https://v.whatsapp.net/v2/register
с параметрами *phoneCode, accountPhone, accountInfo, sms*;
- сервер отправляет информацию об успехе регистрации и поле данных, в котором содержится дополнительный идентификационный параметр клиента.

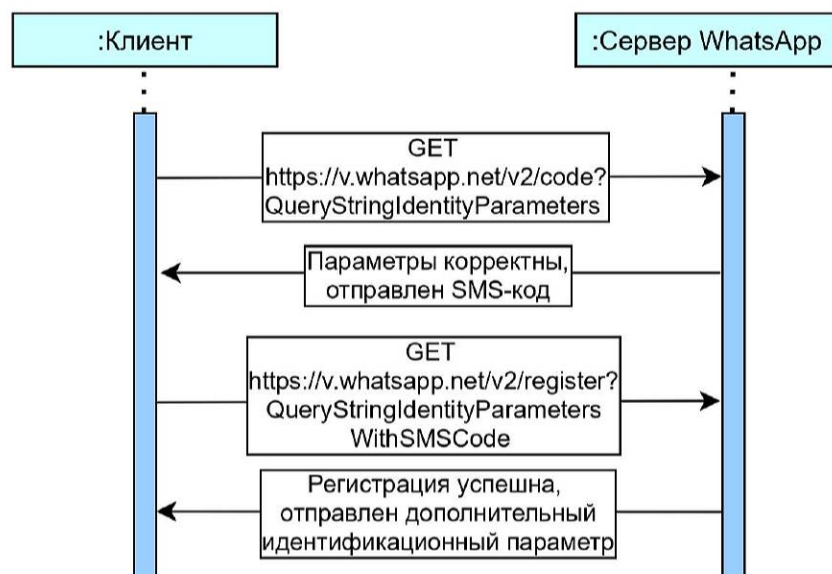


Рис. 1. Сценарий регистрации на сервере WhatsApp по сгенерированным параметрам

Имитация клиента для XMPP-запросов. После регистрации и формирования идентификационных данных клиента следует авторизация на сервере с использованием протокола Noise [7]. При первой авторизации на сервере клиент получает еще один идентификационный параметр. После авторизации производится XMPP-запрос для получения последнего блока key-файла от сервера. В итоге полученные данные от сервера являются последним фрагментом ключа расшифровки.

Алгоритм «облачного» извлечения.

1. Выгрузить зашифрованную базу данных из Google Drive.
2. Сформировать идентификационные данные клиента, используемые для регистрации и запроса на получение данных ключа.
3. Отправить запрос на получение SMS-кода для регистрации.

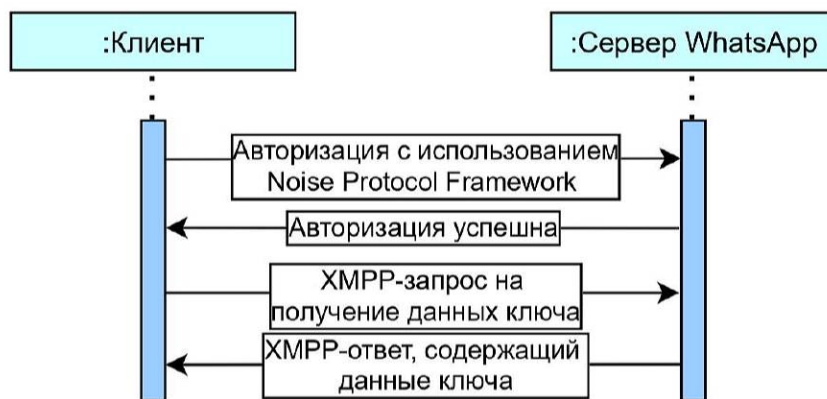


Рис. 2. Сценарий получения данных ключа шифрования по XMPR

4. Отправить запрос на регистрацию по SMS-коду и сохранить новые идентификационные данные от сервера.
5. Авторизоваться на сервере, используя протокол Noise, и сохранить новые идентификационные данные от сервера.
6. Отправить XMPR-запрос на получение данных ключа расшифровки.
7. Расшифровать базу данных.

4. Апробация. Выполнена следующая процедура проверки «облачного» метода извлечения данных.

1. Посредством проекта WhatsApp-GD-Extractor-Multithread извлечена зашифрованная база данных *msgstore.db.crypt12*. Необходимые входные данные – логин/пароль от Google Drive и номер телефона, связанного с учетной записью.
2. Путем имитации клиента сгенерирован ключ расшифровки. Необходимые входные данные – SMS-код.
3. База данных *msgstore.db.crypt12* расшифрована с использованием сгенерированного ключа расшифровки и WhatsApp-Crypt12-Decrypter.

5. Результаты. Исследована генерация ключа расшифровки резервных копий данных Android-приложения WhatsApp. Руководствуясь описанным алгоритмом и имея логин/пароль от учетной

записи Google Drive, а также доступ к SMS-коду, становится возможным реализовать автоматизированный инструмент цифровой криминалистики по извлечению резервных копий Android-версии WhatsApp без участия устройства. Предложенные решения могут быть использованы в том случае, когда WhatsApp изменит версию шифрования (например, на *.crypt13* или *.crypt14*, которые присутствуют в декомпилированном коде). Также основные идеи исследования применимы для обратной разработки других мобильных приложений.

Литература

1. Anglano C. Forensic analysis of WhatsApp Messenger on Android smartphones // Digital Investigation. 2014. Vol. 11. № 3. P. 201–213.
2. YuriCosta. WhatsApp Google Drive Extractor [Электронный ресурс]: URL:<https://github.com/YuriCosta/WhatsApp-GD-Extractor-Multithread> (дата обращения: 21.03.2021).
3. Rotbuf. WhatsApp Key/DB Extractor [Электронный ресурс]: URL:<https://github.com/rotbuf/WhatsApp-Key-DB-Extractor> (дата обращения: 21.03.2021).
4. EliteAndroidApps. WhatsApp Crypt12 Database Decrypter [Электронный ресурс]: URL:<https://github.com/EliteAndroidApps/WhatsApp-Crypt12-Decrypter> (дата обращения: 21.03.2021).
5. Tgalal. yowsup [Электронный ресурс]: URL:<https://github.com/tgalal/yowsup> (дата обращения: 21.03.2021).
6. EliteAndroidApps. WhatsApp KeyGenerator [Электронный ресурс]: URL:<https://github.com/EliteAndroidApps/WhatsApp-Key-Generator> (дата обращения: 21.03.2021).
7. Trevor Perrin. The noise protocol framework [Электронный ресурс]: URL:<https://noiseprotocol.org/> (дата обращения: 23.03.2021).

УДК 004.852

Гладышев Т. Ю.

Реконструкция трехмерных объектов из одинарных изображений при помощи глубинных нейронных сетей

Рекомендовано к публикации доцентом Козыниченко В. А.

1. Введение. Человеку достаточно одного взгляда на фотографию какого-либо объекта, чтобы довольно точно представить этот объект в объеме. Но что именно позволяет нам реконструировать предметы в голове с такой легкостью? Ответ прост – опыт. Когда мы видим какую-либо вещь, мы сразу вспоминаем огромное количество других похожих объектов, которые нам доводилось видеть в течение нашей жизни, это и позволяет нам без труда понять, как эта вещь будет выглядеть с другой стороны.

Для того чтобы компьютер мог повторить успех человека в воссоздании объемных объектов, необходимо прибегнуть к помощи глубинных нейронных сетей (DNN, Deep Neural Network). В данной работе рассмотрено одно из самых популярных решений этой задачи на основе DNN, тестируя его на эталонном в этой области наборе данных – Pix3D [1].

2. Набор данных. Pix3D – набор данных, сформированный учеными Массачусетского технологического института. По словам его создателей, набор отличается от предшественников тем, что содержит несравнимо большее число данных (более 10 тысяч изображений против 759 в другом популярном наборе от IKEA), а также включает в себя как реальные, так и синтетические данные.

Набор состоит из 10 069 фотографий преимущественно предметов мебели. Каждым нескольким фотографиям ставится в соответствие одна трехмерная модель. Всего моделей 395, т. е. около 25 фотографий на штуку, что позволяет алгоритму обучаться как на фотографиях с разных ракурсов, так и на предметах, имеющих одинаковую форму, но разный внешний вид (поверхности разного цвета, текстуры и прочее).

Гладышев Тимофей Юрьевич – студент, Санкт-Петербургский государственный университет; e-mail: st051995@student.spbu.ru, тел.: +7(921)885-19-61

Трехмерные модели представлены в наборе в двух видах: матрицы воксельной сетки, а также в классическом для компьютерной графики формате многогранников. Для решения задачи понадобилось использование псевдотрехмерных набросков. Для их получения многогранники были обработаны в бесплатном инструменте для трехмерного моделирования – Blender3D.

3. Постановка задачи. Обозначим за $I = \{I_k, k = 1, \dots, n\}$ – множество изображений объектов. Сформулируем задачу реконструкции как поиск предсказательной функции $f(I)$, возвращающей упорядоченное множество объектов \tilde{X} , являющийся приближением множества неизвестных точных объектов X . Другими словами, задача – минимизация $L(I) = d_{\Theta}(f(I), X)$, где Θ – набор параметров f , а d – расстояние.

В качестве элементов множества I будут выступать RGB изображения в разрешении 256×256 пикселей, которые также можно представить в виде тензоров размерности $3 \times 256 \times 256$. А в качестве восстанавливаемых объектов множества X – сетки вокселей (кубов одинакового размера, объемный аналог пикселя) размерностью $128 \times 128 \times 128$.

4. Метрика. Теперь определимся с метрикой, с помощью которой мы будем оценивать качество работы алгоритма. Сравнение двух тензоров поначалу может показаться задачей вполне тривиальной, однако важно учитывать, что метрика должна максимально соответствовать восприятию форм человеком. В работе [1] предложены три метрики: IoU, Chamfer distance [2], Earth mover’s distance [3]. Оценим работу алгоритма с помощью IoU.

Рассмотрим метрику IoU. Она представляет собой пересечение деленное на объединение (Intersection over Union), также известное в русскоязычной литературе как *коэффициент Жаккара*, является интуитивной метрикой для сравнения бинарных множеств (а воксельная сетка является бинарным множеством, так как на каждом определенном месте воксель либо присутствует, либо отсутствует):

$$IoU(A, B) = \frac{A \cap B}{A \cup B}.$$

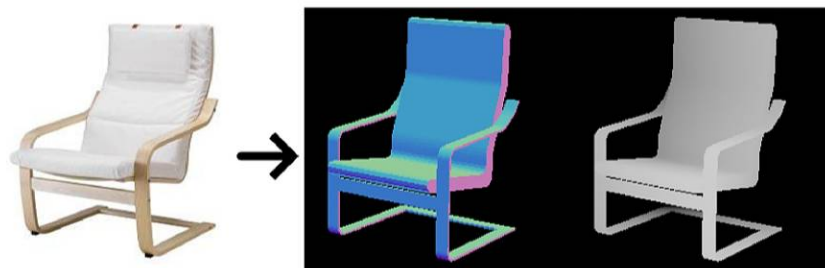


Рис. 1. Пример карт нормалей и глубин

5. Архитектура. Для решения задачи используется нейросетевая модель, основанная на архитектуре MarrNet [4]. Основная идея данного решения заключается в двухступенчатости. Авторы этого подхода ссылаются на работу нейробиолога Дэвида Марра (отсюда и название архитектуры) 1982 года, который утверждал, что в своем восприятии форм человек полагается на так называемые *первоначальные наброски* (primal sketches), на основе которых затем образуются объемные представления.

Предлагается решать задачу в два этапа. На первом этапе из разноцветных фотографий получают так называемые карты нормалей и карты глубин. Карта нормалей – это изображение, в котором цвет пикселя используется как трехмерный вектор для задания угла наклона нормали к поверхности. Карта глубин – черно-белое изображение, на котором яркостью пикселя задается расстояние от поверхности до наблюдателя. Примеры карт нормалей и глубин представлены на рис. 1. Карты нормалей и глубин вместе с силуэтом объекта будем называть *псевдо-трехмерными набросками* (2.5D sketches).

Архитектура первого модуля представляет собой Unet-подобную модель с сетью ResNet-18 [5] в качестве кодировщика, переводящую изображение в 512-мерный вектор скрытых признаков. Затем декодировщик – зеркально отраженный ResNet – строит необходимые псевдо-трехмерные наброски. Будем реализовывать три экземпляра данной архитектуры для предсказания каждого из трех набросков (карта нормалей, карта глубин, силуэт объекта) по отдельности.

На втором этапе, на основе результатов работы первого модуля, предсказываются конечные воксельные сетки. Архитектура второго модуля основана на принципе автокодировщика. В начале аналогич-

ная сеть ResNet-18 переводит наброски в 200-мерный вектор скрытых признаков, затем декодировщик на базе слоев трехмерной транспонированной свертки строит искомые сетки вокселей размерности 128^3 . Подробная схема слоев декодировщика приведена в таблице.

Таблица. Архитектура декодировщика второго модуля

Тип слоя	Параметры
deconv3d	размерность: 200 в 512, ядро: 4?4?4, сдвиг: 1, активация: relu
batchnorm3d	—
deconv3d	размерность: 512 в 256, ядро: 4?4?4, сдвиг: 2, активация: relu
batchnorm3d	—
deconv3d	размерность: 256 в 128, ядро: 4?4?4, сдвиг: 2, активация: relu
batchnorm3d	—
deconv3d	размерность: 128 в 64, ядро: 4?4?4, сдвиг: 2, активация: relu
batchnorm3d	—
deconv3d	размерность: 64 в 32, ядро: 4?4?4, сдвиг: 2, активация: relu
batchnorm3d	—
deconv3d	размерность: 32 в 1, ядро: 4?4?4, сдвиг: 2

6. Обучение. Сети первого модуля обучались по отдельности. В качестве функции погрешностей использовалось среднеквадратическое отклонение. Был взят оптимизатор Adam [6] со скоростью обучения 2×10^{-4} . Количество эпох – 32 для каждой из трех сетей. Сети второго модуля – кодировщик и декодировщик – обучались совместно. За функцию погрешностей была принята бинарная перекрестная энтропия (binary cross entropy), а в качестве оптимизатора взят стохастический градиентный спуск со скоростью обучения 0,1 и импульсом 0,9. Количество эпох – 41. Размер батча во всех случаях – 4. Количество батчей в обучающей выборке – 2189.

7. Результаты. Работа обоих модулей оценивается как подсчет метрик для выходных данных второго модуля. Оценка проводилась на 100 случайным образом взятых изображениях. Итоговые результаты – IoU: 0.2752. Пример реконструкции трехмерного объекта (дивана) приведен на рис. 2.

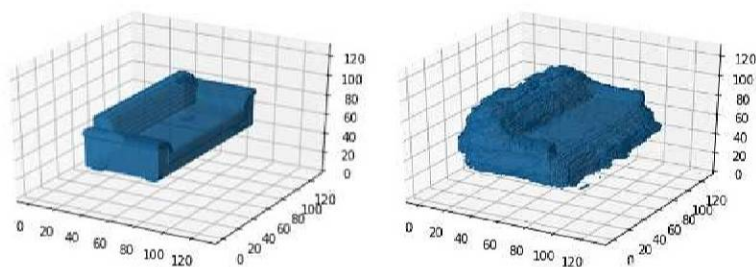


Рис. 2. Пример работы алгоритма

Литература

1. Xingyuan Sun, Wu J., Zhang Z. Pix3D: dataset and methods for single-Image 3D shape modeling // IEEE CVPR. 2018. P. 2974–2983.
2. Kurenkov A., Ji J., Garg A., Mehta V. et al. DeformNet: free-form deformation network for 3D shape reconstruction from a single image // IEEE WACV. 2018. P. 858–866.
3. Rubner Y., Tomasi C., Guibas L. J. The earth mover's distance as a metric for image retrieval // IJCV. 2000. P. 99–121.
4. Wu J., Wang Y., Xue T., Sun X., Freeman B., Tenenbaum J. MarrNet: 3D shape reconstruction via 2.5D sketches // NIPS. 2017. P. 540–550.
5. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition // IEEE CVPR. 2016. P. 770–778.
6. Kingma D. P., Ba J. Adam: a method for stochastic optimization [Электронный ресурс] // ICLR. 2015. URL:<https://arxiv.org/abs/1412.6980> (дата обращения: 18.05.2021).

УДК 004.932

Грашин Д. М.

Определение размера текстуры для реконструированной модели в приложении Agisoft MetaShape

Рекомендовано к публикации профессором Громовой Е. В.

1. Введение. В данной статье рассматривается процесс фотограмметрии [1], в котором с использованием фотографий объекта, снятого с разных ракурсов, можно реконструировать 3D-модель этого объекта. Для этого процесса был выбран программный продукт Agisoft Metashape [2] от компании Agisoft LLC, исходя из сравнительной таблицы по скорости и качеству реконструирования [3]. Также в приложении можно управлять процессом автоматизированно, при помощи скриптов.

После получения реконструированной модели необходимо сгенерировать его текстуру. Под текстурой понимается квадратное изображение, которое является отображением внешней поверхности модели на двумерную плоскость. В данной статье предложен метод для вычисления размера текстуры 3D-модели. Преимущества данного метода по сравнению с существующими заключаются в получении текстуры без потери полезной информации от изображений на соответствующем этапе, а также в практическом применении для автоматизации процессов.

2. Постановка задачи. На этапе реконструирования модели определены данные о положениях камеры в пространстве относительно объекта съемки. Этап текстурирования выглядит следующим образом. В зависимости от того, на какую часть модели направлено определенное положение камеры, генерируется текстура модели, где для каждого плоского треугольника реконструированной модели (полигона) существует соответствующий треугольник на 2D-текстуре. Размер текстуры в приложении Agisoft MetaShape задается только вручную. При малом размере текстуры есть вероятность

Грашин Дмитрий Михайлович – студент, Санкт-Петербургский государственный университет; e-mail: st044391@student.spbu.ru, тел.: +7(921)885-62-50

потерять информацию, но в то же время слишком большой размер увеличит время выполнения алгоритма, не привнося дополнительной полезной информации. Поэтому необходимо разработать алгоритм его поиска. При этом следует опираться не только на информацию о положениях камер, но и на характеристики используемой камеры, такие как фокусное расстояние, размер сенсора, количество пикселей на получаемых изображениях. Также полезными данными будут площадь поверхности объекта в реальных величинах, минимальное расстояние от модели до положений камеры и особенности заполнения текстуры в используемом приложении.

3. Предлагаемый метод. Для вычисления необходимого размера текстуры, которая будет максимально точно отображать поверхность реконструированной 3D-модели без потери информации от исходных изображений объекта, предлагается следующий пошаговый метод.

3.1. Масштабирование модели до реальных размеров. Реконструированная модель масштабируется до своего реального размера при помощи маркеров [4], показанных на рис. 1. Особенность маркеров состоит в том, что можно задать точное расстояние между их центрами и, опираясь на эти данные, масштабировать 3D-модель до реальных размеров. Обозначим расстояние в реальных величинах между центрами маркеров за r , а расстояние в системе координат модели – за l . Тогда коэффициент для масштабирования реконструированной модели

$$s = \frac{r}{l} = \frac{133}{0,214} = 622.$$

Используя встроенные инструменты приложения Agisoft Metashape, описанные в [4], масштабируем 3D-модель до ее реальных размеров.

3.2. Вычисление площади поверхности 3D-модели. Реконструированная модель состоит из большого числа плоских треугольников (полигонов) F_k . Складывая последовательно площади треугольников $S(F_k)$, получим общую площадь поверхности модели:

$$A = \sum_{k=0}^n S(F_k) = 9360 \text{ мм}^2. \quad (1)$$

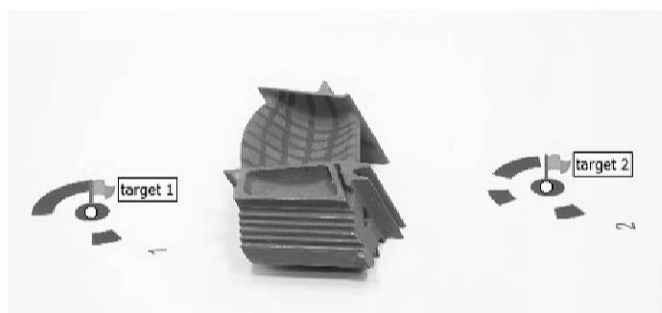


Рис. 1. Пример фотографии объекта, рядом с которым лежат маркеры

3.3. Вычисление полезной части текстуры. Теперь для модели необходимо построить текстуру размера по умолчанию для того, чтобы вычислить коэффициент ее полезной части. Под полезной частью текстуры подразумевается количество пикселей, используемых для отображения модели. Подсчет производится следующим образом: информация о положении текстур для каждого полигона находится в исходном файле 3D-модели. Используя эту информацию, можно перезаписать текстуру, заполняя треугольники белым цветом. Коэффициент полезной части текстуры рассчитаем как отношение количества белых пикселей P_w к общему числу пикселей на получившейся текстуре P :

$$T = \frac{P_w}{P} = \frac{14386000}{25160400} = 0,55. \quad (2)$$

Пример сгенерированной текстуры и результат перезаписывания ее полезной части представлен на рис. 2.

3.4. Вычисление расстояния от камеры до модели. Исходя из информации о положениях камеры, найдем минимальное расстояние между поверхностью 3D-модели и местами съемок. Так как модель состоит из большого числа вершин, которые соединены в треугольники, будем искать минимальное расстояние, перебирая все возможные вершины. Для этого обозначим через P_i – i -ю позицию камеры, а M_j – j -ю вершину модели. Тогда получим минимальное



Рис. 2. Слева – текстура, сгенерированная для 3D-модели. Справа – полезная часть этой текстуры

расстояние

$$d = \min_{i=1,n;j=1,m} distance(P_i, M_j) = 960 \text{ мм.} \quad (3)$$

3.5. Определение соответствия реальных величин пикселю. Далее следует определить количество миллиметров, находящихся в одном пикселе на изображении. Для этого используем формулу для расчета соответствия пикселя и реальных величин (Ground Sample Distance):

$$GSD_w = \frac{S_w d}{F I_w}, \quad (4)$$

где величины F – фокусное расстояние в момент съемки, S_w – ширина сенсора камеры, I_w – ширина изображения в пикселях, являются характеристиками камеры Canon 5D Mark II. Тогда в нашем случае

$$GSD_w = \frac{35,9 \cdot 960}{72 \cdot 6240} = 0,08 \text{ мм/пкс.} \quad (5)$$

Использование в (4) именно минимального расстояния d из (3) обусловлено тем, что при увеличении расстояния значение GSD_w увеличивается, следовательно, будет происходить потеря информации от исходных изображений.

3.6. Вычисление размера текстуры. Используя (1), (2), (5), найдем значение *Texture size* – величины стороны текстуры в пикселях, которая требуется, чтобы не потерять полезную информацию от изображений на этапе текстурирования, по формуле:

$$Texture\ size = \sqrt{\frac{A}{GSD_w^2 T}} = \sqrt{\frac{9360}{0,0064 \cdot 0,55}} = 1630\ \text{пкс.} \quad (6)$$

В формуле (6) из вычисленного значения извлекается корень, так как текстура – это квадратное изображение и для получения параметра нужна только одна из его сторон.

4. Анализ результатов. Для анализа результатов построена текстура для 3D-модели с найденным по формуле (6) параметром размера текстуры. Также построена текстура со значением 500 пикселей, которое было выставлено произвольно, без опоры на какие-либо данные. Результат представлен на рис. 3. Текстура с рассчитанным размером 1630 пикселей на модели обладает резкостью и четкостью линий. В то же время текстура с размером 500 пикселей выглядит менее резкой, размытой и нечеткой.

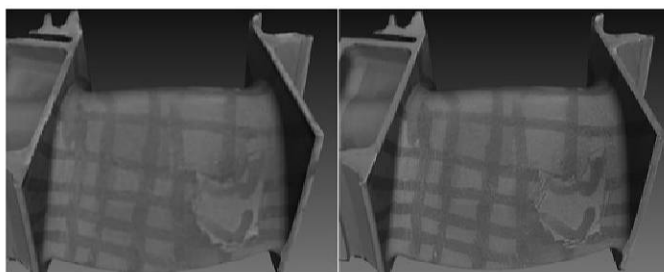


Рис. 3. Слева – пример наложения текстуры размера 500 пикселей на 3D-модель. Справа – пример наложения с текстурой размера 1630 пикселей

5. Заключение. В работе предложен метод определения размера текстуры, опирающийся на параметры съемки объекта и инфор-

мации о нем, который может быть автоматизирован для практического применения. Метод имплементирован на языке программирования Python 3.8 и протестирован в Agisoft Metashape Professional 1.7.2. Главное отличие от подбора вручную, встроенного в используемое приложение, в том, что информация от изображений при перенесении на текстуру модели не потеряется ввиду недостаточно выставленного параметра размера текстуры. Для предложенного метода найдено значение размера текстуры, для результатов проведен анализ и сравнение с исходным подходом.

Литература

1. Stylianidis E. Photogrammetric survey for the recording and documentation of historic buildings // Springer International Publishing. 2020. P. 139–195.
2. Agisoft [Электронный ресурс]: URL: <https://www.agisoft.com> (дата обращения: 21.03.2021).
3. Aati S., Rupnik E., Nejim S. Comparative study of photogrammetry software in industrial field // Revue Francaise de Photogrammetrie et de Teledetection. 2020. No 221. P. 37–48.
4. Coded targets and Scale bars [Электронный ресурс]: URL: <https://agisoft.freshdesk.com/support/solutions/articles/31000148855-coded-targets-and-scale-bars> (дата обращения: 21.03.2021).

УДК 004.4

Жереб В. В.

Реализация алгоритмов RAID-вычислений для процессора ARM

Рекомендовано к публикации профессором Утешевым А. Ю.

1. Введение. Объемы информации, хранимые человеком, растут с каждым годом. Для безопасного хранения данных и предоставления гарантированного доступа к ним применяется технология RAID (Redundant Array of Independent Disks) – массив из нескольких дисков, управляемый контроллером, «отказоустойчивый массив из независимых дисков».

Для обеспечения отказоустойчивости массивов на них записываются и хранятся не только данные, но и некоторая избыточная информация, называемая контрольными суммами, которая позволяет восстанавливать данные в случае их частичной утраты.

Существует множество модификаций RAID. Например, RAID 6 позволяет параллельную запись, поскольку блоки данных и контрольные суммы циклически записываются на все диски массива. В данной модификации возможно восстановление одного или двух утраченных дисков за счет использования двух контрольных сумм, вычисляемых двумя независимыми способами. Так как технология RAID 6 дает много возможностей для восстановления данных, в этой работе рассматривалась именно эта модификация. При работе с RAID 6 выделяется несколько подзадач: расчет контрольных сумм и восстановление утраченных дисков.

Если необходимо обеспечить взаимодействие данных из RAID-массива с другими устройствами в сети, данные поступают к необходимым устройствам через сетевую плату (network interface controller – NIC). Проблема заключается в том, что в такой ситуации велика нагрузка на центральный процессор.

В связи с потребностью уменьшить количество задач, выполняемых центральным процессором, были разработаны SmartNIC (Smart

Жереб Вера Вадимовна – студент, Санкт-Петербургский государственный университет; e-mail: vzhereb9@gmail.ru, тел.: +7(952)392-29-57

network interface controller) – сетевые платы, которые с помощью собственного ARM-процессора принимают на себя задачи обработки, обычно выполняемые центральным процессором [1].

Данная работа позволит уменьшить нагрузку на центральный процессор при помощи вынесения части логики технологии RAID в SmartNIC. Исследование проводилось на базе компании RAIDIX.

2. Используемая терминология. Каждый диск разделяется на блоки, размер которых будет удобен для дальнейшей обработки. Такие блоки называются *стрипами*.

Синдромы – дополнительные стрипы, использующиеся для восстановления данных. Обозначаются P, Q .

Стрипы, предназначенные для хранения данных, обозначаются D_0, \dots, D_{N-1} , где N – количество дисков (два диска с синдромами не учитываются).

Страйп – основная единица обработки данных в системе хранения. Каждый диск разделяется на блоки одинакового размера (стрипы), они нумеруются внутри одного диска, затем блоки с одинаковыми номерами из разных дисков формируют страйп.

3. Математическое описание алгоритма расчета синдромов и восстановления утраченных дисков. В основе алгоритма подсчета синдромов и восстановления утраченных дисков лежит арифметика полей Галуа [2]. Для кодирования было выбрано поле Галуа $GF(2^8)$. Оно позволяет обрабатывать до 256 дисков, для современных нужд такого количества достаточно, притом кодирование при увеличении числа дисков значительно усложняется.

Синдромы P и Q рассчитываются по следующим формулам (умножение на x и сложение производится по двойному модулю: 2, неприводимый многочлен восьмой степени) [3]:

$$P = \sum_{i=0}^{N-1} D_i,$$

$$Q = \sum_{i=0}^{N-1} x^{N-i-1} D_i = ((D_0x + D_1)x + D_2)x + \dots + D_{N-1}.$$

Утраченные блоки страйпа D_α и D_β , такие что $\alpha < \beta$, восстанавливаются по следующим формулам:

$$D_\beta = \frac{\bar{P}_{\alpha,\beta} - \bar{Q}_{\alpha,\beta} x^{\alpha-N+1}}{1 - x^{\alpha-\beta}},$$

$$D_\alpha = \overline{P}_{\alpha,\beta} - D_\beta,$$

где $\overline{P}_{\alpha,\beta}$ и $\overline{Q}_{\alpha,\beta}$ – пересчитанные P и Q для страйпа с утраченными дисками.

Заметим, что скорость чтения и записи данных напрямую зависит от скорости выполнения арифметических операций в полях Галуа. В данной работе оптимизируется выполнение именно этих операций.

4. Описание реализованных алгоритмов. Выполнена реализация трех алгоритмов кэширования. Алгоритмы без векторизации и с векторизацией различаются количеством информации, обрабатываемой за одну единицу времени. Алгоритм компании RAIDIX основан на другом подходе – «побитовом параллелизме» [4].

4.1. Алгоритм без векторизации. В данном алгоритме рассматривается отдельно каждый байт как один элемент поля $GF(2^8)$. Достоинство этого подхода в том, что его возможно применять на устройствах, не поддерживающих векторизацию, но это самый медленный из представленных алгоритмов.

Для всех блоков данных выполняются одинаковые действия. Это позволяет применять различные алгоритмы векторизации.

4.2. Алгоритм с векторизацией. Данный алгоритм рассматривает часть одного блока страйпа как шестнадцать последовательных многочленов из $GF(2^8)$.

4.3. Алгоритм компании RAIDIX. В отличие от предыдущих алгоритмов, в которых страйп разбивается на части, где каждая содержит элементы поля $GF(2^8)$, в данном алгоритме регистры мысленно располагаются «вертикально», и происходит обработка сразу 128 элементов поля.

Подробное описание алгоритма представлено в [4].

Это самый быстрый из рассматриваемых алгоритмов, необходимо всего три операции хог и перестановки для умножения на примитивный элемент поля. Недостаток его в том, что для большего числа контрольных сумм требуется больше векторных регистров, требует поддержки векторизации.

5. Используемые инструменты. Код написан на языке программирования C. В данный язык программирования встроены Intrinsics – это функции, которые компилятор заменяет соответствующей инструкцией или последовательностью инструкций.

Для разработки использовались команды, поддерживающие векторные расширения SSE (для Intel), а также команды, использующие расширения Neon (для ARMv8).

SSE – это векторное расширение для архитектуры Intel, включающее в себя 16 (для 64-битных систем) 128-битных регистров XMM0-XMM15 и набор инструкций для них [5].

Neon – это векторное расширение для архитектуры ARM, комбинированный 64- и 128-битный набор команд. Neon обладает внушительным набором команд, отдельными регистровыми файлами и независимой системой исполнения на аппаратном уровне [6].

6. Тестирование. Ниже представлено описание тестирования.

1. Для выбранного количества страйпов и размера страйпа выделялась область памяти нужного размера, которая впоследствии заполнялась случайными данными. Эта память внутри каждого страйпа разделялась на стрипы, которые эмулировали различные жесткие диски.
2. Запускалась функция расчета синдромов. Вычисленные синдромы сохранялись в соответствующих блоках.
3. Проверялась правильность работы функций восстановления утраченных стрипов:
 - данные страйпа сохранялись в отдельном участке памяти;
 - случайным образом выбирались номера тех стрипов, чье повреждение эмулировалось;
 - эти блоки страйпа заполнялись нулями;
 - вызывалась функция восстановления поврежденных стрипов, которая использовала вычисленные ранее синдромы;
 - проверялось совпадение данных, сохраненных до повреждения, и тех данных, которые получились после выполнения функции восстановления.

Для измерения времени использовался таймер в наносекундах. Для выбранного количества дисков (4, 8, ..., 64) для каждой функции тест проводился 10000 раз, скорость выполнения считалась в мегабайтах в секунду (MB/s). Затем отбрасывалось по 5% самых маленьких и самых больших по величине результатов. Из оставшихся выбиралось среднее арифметическое значение.

Характеристики тестового сервера для SSE (Intel):

ОС: 18.04.2-Ubuntu

CPU: Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz

RAM: 16 GB

Характеристики тестового сервера (SmartNIC Stingray™

PS1100R) для Neon (ARM):

ОС: 18.04-Ubuntu

CPU: ARMv8 Cortex-A72 @ 3.0GHz

RAM: 8 GB

7. Результаты. На рис. 1–6 представлены графики скорости различных RAID-вычислений для каждого из реализованных алгоритмов. Метод без векторизации значительно уступает алгоритмам,

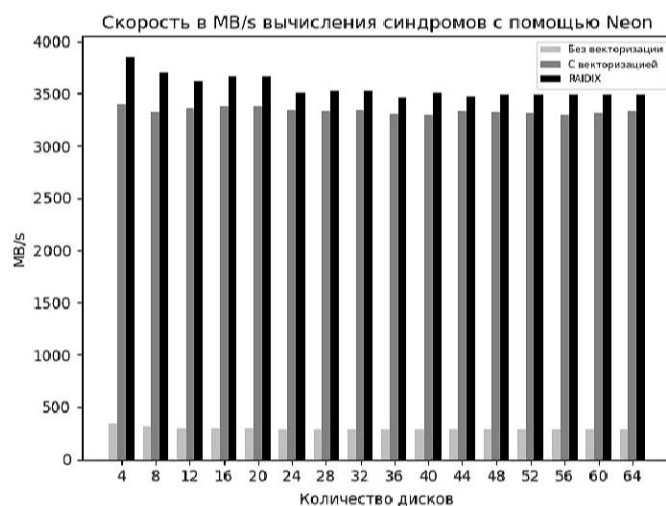


Рис. 1. Вычисление синдромов с помощью Neon

использующим векторизацию. Например, функция вычисления синдромов выполняется алгоритмом без векторизации примерно в 11 раз медленнее, чем алгоритмом с векторизацией, и в 12 раз медленнее, чем алгоритмом RAIDIX с помощью Neon (ARM). Для SSE (Intel) эти значения равны соответственно 15 и 19 раз.

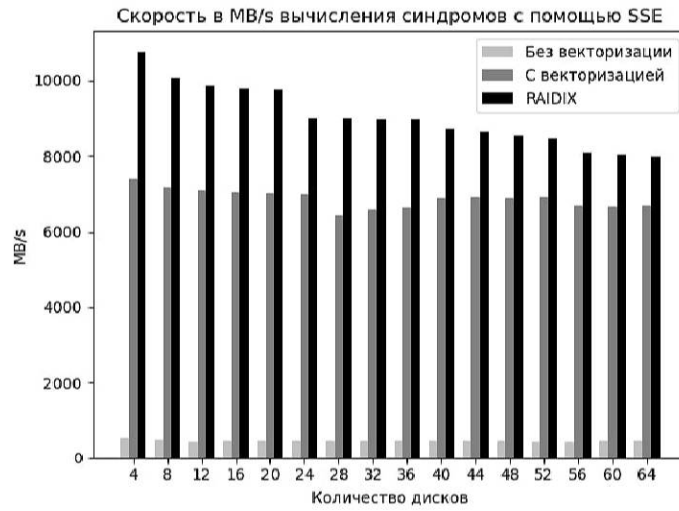


Рис. 2. Вычисление синдромов с помощью SSE

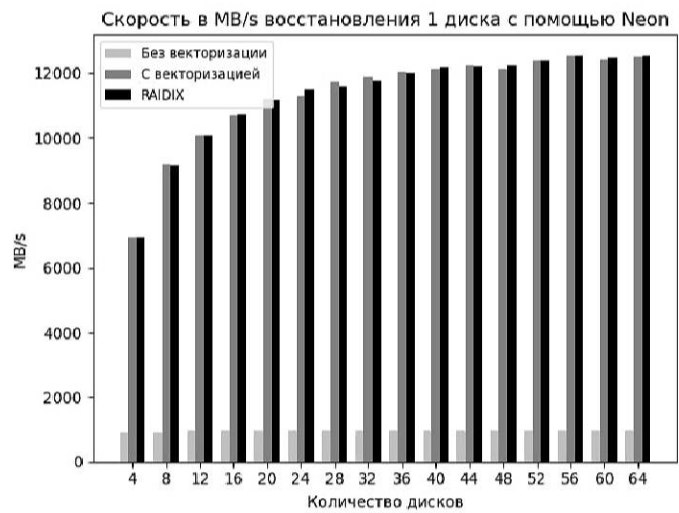


Рис. 3. Восстановление одного поврежденного диска с помощью Neon

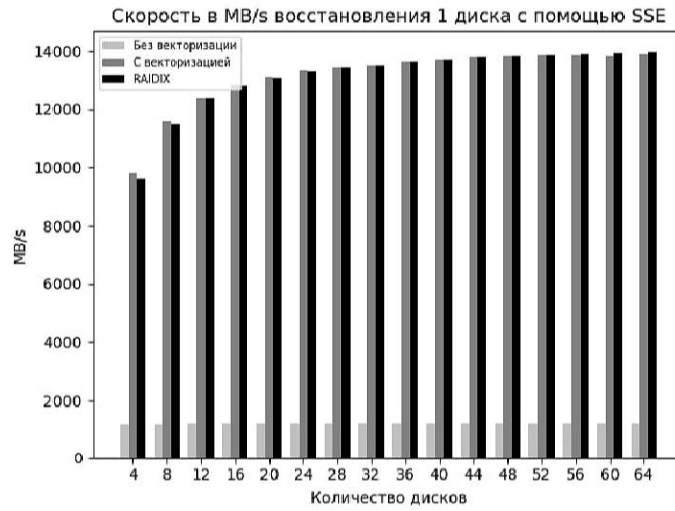


Рис. 4. Восстановление одного поврежденного диска с помощью SSE

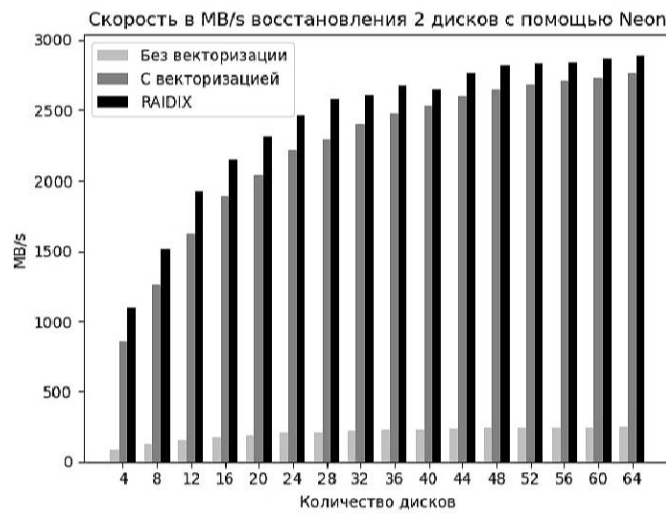


Рис. 5. Восстановление двух поврежденных дисков с помощью Neon

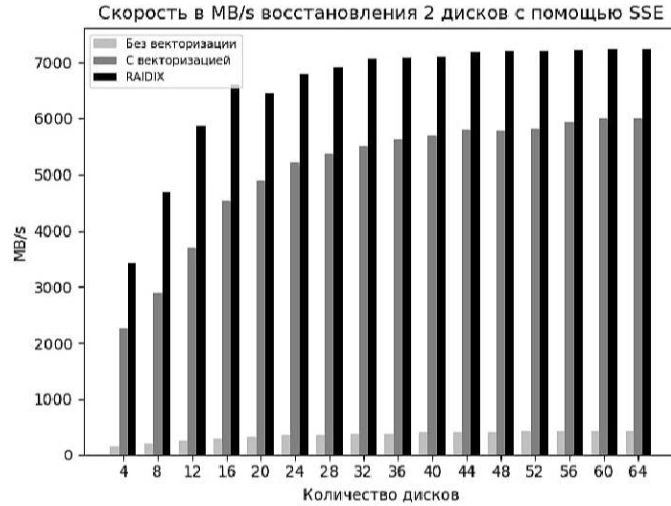


Рис. 6. Восстановление двух поврежденных дисков с помощью SSE

Восстановление одного поврежденного диска алгоритмами с векторизацией и RAIDIX дает почти одинаковые результаты, так как при восстановлении одного диска используется только синдром $P = \sum_{i=0}^{N-1} D_i$, а следовательно, достаточно только операции хог (умножение на x не используется). Таким образом, реализация этих алгоритмов для восстановления одного диска не будет отличаться друг от друга.

Алгоритм компании RAIDIX показал самые лучшие результаты. Например, восстановление двух поврежденных дисков происходит в 12 раз быстрее, чем с помощью алгоритма без векторизации и примерно в 1,1 раза быстрее, чем с помощью алгоритма с векторизацией при использовании Neon. Для SSE эти значения равны соответственно 18 и 1,3 раза.

Так как производительность SmartNIC Stingray™ PS1100R меньше, чем тестового сервера для SSE, скорости RAID-вычислений на архитектуре ARMv8 оказались ниже.

8. Заключение. Реализованы и протестированы три алгоритма: без векторизации, с векторизацией и алгоритм компании RAIDIX. В SmartNIC перенесены алгоритмы вычисления синдромов и восстановления поврежденных дисков. Проведен сравнительный анализ эффективности работы алгоритмов, в ходе которого доказана эффективность алгоритма компании RAIDIX как для процессоров Intel, так и для ARM.

Литература

1. What Is a SmartNIC [Электронный ресурс]: URL: <https://blog.mellanox.com/2018/08/defining-smartnic/> (дата обращения: 21.03.2021).
2. Поля Галуа [Электронный ресурс]: URL: <http://vmath.ru/vf5/gruppe/galois> (дата обращения: 21.03.2021).
3. Федоров А. Р. Способ восстановления записей в запоминающем устройстве, система для его осуществления и машиночитаемый носитель // Патент № RU2448361C2. 2010.
4. Федоров А. Р. Способы кодирования информации для построения программных отказоустойчивых дисковых массивов // Системное программирование. 2012. Т. 7. С. 7–31.
5. Intel Intrinsic Guide [Электронный ресурс]: URL: <https://software.intel.com/sites/landingpage/IntrinsicsGuide/#> (дата обращения: 21.03.2021).
6. Neon Programmer's Guide for Armv8-A [Электронный ресурс]: URL: <https://developer.arm.com/architectures/instruction-sets/simd-isas/neon/neon-programmers-guide-for-armv8-a> (дата обращения: 21.03.2021).

УДК 004.451.622

Ильченко И. И.

Реализация сетевого пространства имен в ОС Embox

Рекомендовано к публикации доцентом Луцивом Д. В.

1. Введение. Виртуализация на уровне операционной системы является активно развивающимся направлением в разработке современных операционных систем. Ее суть заключается в поддержке нескольких изолированных экземпляров некоторых элементов пользовательского окружения. Ядро ОС Linux пошло по пути механизма пространств имен, который виртуализирует определенные ресурсы системы для процессов, принадлежащих данному пространству имен. На момент написания работы в Linux имеется восемь типов пространств имен [1], среди которых нас будет интересовать пространство имен (ПИ) для сетевых ресурсов.

Embox – проект по разработке ОСРВ для встраиваемых систем, одной из основных возможностей которой является возможность запуска ПО для Linux [2]. Для удовлетворения этого требования Embox должен предоставлять то же API для пользовательских приложений, что и Linux. Однако на данный момент Embox не предоставляет возможность создавать пространства имен. В рамках данной работы сделаны существенные продвижения в сторону реализации одного из пространств имен – сетевого, и заложен фундамент для создания пространств имен других типов.

2. Обзор. Виртуализация на уровне операционной системы реализуется теми или иными способами во многих ОС. Поскольку ОСРВ Embox стремится к совместимости пользовательских приложений с Linux, рассмотрим подход этого ядра.

Для достижения целей виртуализации ядро Linux добавляет структуру `nsproхu` с полями, содержащими указатели на структуры каждого из реализованных типов ПИ. Основная структура для задачи в ядре – `task struct` – расширяется полем типа `nsproхu`, как

Ильченко Иван Игоревич – студент, Санкт-Петербургский государственный университет; e-mail: st063119@student.spbu.ru, тел.: +7(911)834-94-65

раз и предоставляя задачам возможность находиться в различных пространствах имен.

Ядро Linux также добавляет новый системный вызов `unshare`, который в контексте механизма ПИ позволяет добавить вызывающую задачу в новые ПИ. Этот системный вызов принимает в качестве единственного параметра комбинацию флагов, каждый из которых соответствует своему типу ПИ. В ходе работы системного вызова создаются указанные ПИ и объект `nsproхu`, ссылающийся на них и на который впоследствии ссылается вызывающая задача, которая тем самым при завершении вызова начинает принадлежать к указанным ПИ.

Помимо системного вызова `unshare`, с поддержкой ПИ в Linux добавляется системный вызов `setns()` с первым параметром – дескриптором уже существующего ПИ (см. ниже). Этот системный вызов присоединяет вызывающую задачу к ПИ, соответствующему переданному дескриптору. Также расширяются флаги, передаваемые в вызов `clone`, который теперь может создавать новую задачу в ПИ тех типов, чьи флаги были переданы.

Информация о принадлежности задачи к тому или иному ПИ находится в виртуальной файловой системе `/proc`. Открытие специальной символической ссылки вида `/proc/PID/ns/nstype`, где `PID` – это уникальный идентификатор задачи, а `nstype` – тип ПИ, как раз и позволяет получить дескриптор ПИ, по которому можно присоединиться к ПИ с помощью описанного выше вызова `setns` [3].

Аналогичный подход к организации ПИ был применен и в ядре `Embox`, однако с собственными решениями в сетевом стеке и добавлением возможности выбора данного механизма за счет системы сборки `Embox – Mybuild` [4]. Однако из-за текущих ограничений работы с файловой системой `/proc`, пользовательское API аналогично Linux, за исключением идентификации ПИ, которые на данный момент имеют глобальный строковый идентификатор.

3. Пространства имен в ядре и окружении пользователя.

3.1. Добавление пространств имен в ядро `Embox`. Основным объектом для реализации ПИ в `Embox` стала структура `nsproхu`, аналогичная для ядра Linux, с ссылкой на единственное реализованное сетевое ПИ и возможностью дальнейшего расширения полей для ПИ новых типов. Структура для задач `task` в `Embox` расширена полем `nsproхu`. Для возможности опционального включения ПИ

в собираемый образ использовался механизм абстрактных модулей системы сборки Mybuild.

3.2. Изменение в сетевом стеке для поддержки пространств имен. Сетевые стеки в операционных системах выстроены по идеям сетевых иерархических моделей, в которых протоколы высших уровней опираются на протоколы нижних уровней. Рассмотрим реализованные изменения в сетевом стеке с этой же точки зрения, начиная с высших уровней и спускаясь к нижним.

Начнем с транспортных протоколов и программного интерфейса сокетов. Каждый сокет ассоциирован с парой адрес и порт, и наличие виртуализации на этом уровне означает возможность иметь одинаковые пары, но в разных ПИ. Для этого в структуру сокета добавляется поле со ссылкой на сетевое пространство имен (СПИ), с которым ассоциирован данный сокет. Информация о принадлежности сокета используется и для отправки сетевого пакета для указания его принадлежности к тому же СПИ, что и сам сокет. Аналогично устроен прием пакета.

Пакет транспортного уровня инкапсулируются в пакет сетевого уровня. При этом в процессе инкапсуляции возникает вопрос о маршрутизации пакета и определении интерфейса, с которого пакет будет отправлен. Для этого используются так называемые таблицы маршрутизации. СПИ позволяют иметь разные таблицы маршрутизации, поэтому для реализации данной возможности структура, отвечающая в ядре за запись в таблице маршрутизации, расширяется полем со ссылкой на владеющее ей СПИ.

За протоколами сетевого уровня следуют протоколы канального уровня. В пределах одной сети на этом уровне решается вопрос сопоставления целевого сетевого адреса с его адресом канального уровня. Для этих целей используется подсистема neighbour, которая поддерживает таблицу с указанными соответствиями между адресами разных уровней. Для каждой записи в этой таблице указывается сетевое устройство, которому она принадлежит, и для всех функций, работающих с модификацией таблицы, также указывается сетевое устройство. Таким образом СПИ на этом уровне поддерживаются сами собой через поддержку СПИ на уровне сетевых устройств, описанной далее.

Работа со сформированными для отправки или полученным пакетом происходит на уровне сетевых устройств, каждое из которых принадлежит своему единственному СПИ. Соответственно каждое

из сетевых устройств ссылается на СПИ, которому оно принадлежит, и добавляется возможность переносить устройство из одного СПИ в другое. С точки зрения прохождения пакета по сетевому стеку при приеме, пакет попадает во вспомогательную структуру буфера сокета, который косвенно через ссылку на устройство ссылается и на его СПИ, что в дальнейшем и позволяет доходить пакету в пользовательское приложение из того же СПИ, что и само устройство.

3.3. Пространства имен в пользовательском окружении. В текущей версии СПИ идентифицируются уникальным строковым именем, и задача может либо присоединиться в уже существующее СПИ с помощью вызова `setns` с аргументом, принимающим имя СПИ, либо создать новое СПИ и добавиться в него с помощью вызова `unshare` со строковым аргументом, которым будет идентифицироваться созданное СПИ.

Для демонстрационного примера была добавлена утилита `psexec`, принимающая в качестве параметров имя СПИ и имя программы с ее аргументами, которая вследствие работы утилиты запустится в указанном СПИ. Так же были добавлена возможность создавать виртуальную пару соединенных Ethernet устройств, которая позволяет пакетам, отправленным с одного устройства, быть полученными на другом устройстве и наоборот, и утилита `addveth` с двумя аргументами для имен СПИ, в которые добавляются концы созданной виртуальной пары устройств. Функциональность описанных утилит аналогична утилите `ip` в Linux [5].

4. Демонстрационный пример. Суть примера состоит в запуске на одной машине двух пользовательских приложений в разных СПИ, передающих друг другу сообщения по протоколу UDP с портом одного и того же номера, что является ключевым моментом, поскольку без СПИ этого нельзя достичь. Отметим, что запуск приложений происходит с помощью утилит, описанных ранее. Пакеты приложений отправляются через виртуальную пару сетевых устройств, каждое из которых находится в соответствующих СПИ указанной пары приложений. Также каждое из устройств предварительно настроено через утилиту `psexec`. Все указанные запуски утилит автоматизированы с помощью добавленного в тестовую конфигурацию скрипта.

5. Заключение. Получены следующие результаты:

1. Разработана общая архитектура пространств имен в операционной системе Embox.
2. Добавлены необходимые изменения в большую часть сетевого стека для реализации сетевых пространств имен.
3. Реализован API пространств имен для пользовательского окружения, аналогичный разработанному в ядре Linux.
4. Выполнено тестирование на демонстрационной конфигурации.

Литература

1. Overview of Linux namespaces [Электронный ресурс]: URL:<https://man7.org/linux/man-pages/man7/namespaces.7.html> (дата обращения: 20.03.2021).
2. Embox – Modular and configurable OS for embedded applications [Электронный ресурс]: URL:<http://embox.github.io> (дата обращения: 20.03.2021).
3. Linux Kernel Networking: Implementation and Theory [Электронный ресурс]: URL:https://www.oreilly.com/library/view/linux-kernel-networking/9781430261964/9781430261964_Ch14.xhtml (дата обращения: 20.03.2021).
4. Mybuild – система сборки для модульных приложений [Электронный ресурс]: URL:<https://habr.com/ru/company/embox/blog/144935/> (дата обращения: 20.03.2021).
5. Process network namespace management [Электронный ресурс]: URL:<https://man7.org/linux/man-pages/man8/ip-netns.8.html> (дата обращения: 20.03.2021).

УДК 004.93

Кванчiani К. Ш.

Повышение качества медицинских изображений в контексте задачи сегментации опухолей

Рекомендовано к публикации доцентом Блекановым И. С.

1. Введение. Сегодня медицинская визуализация является одной из самых развивающихся областей здравоохранения [1]. Основной проблемой медицинской визуализации является сложность сбора необходимого количества качественных данных для наиболее эффективного анализа медицинских изображений методами искусственного интеллекта. В медицине собрать множество изображений хорошего качества – значит не только преодолеть трудности получения доступа к персональным данным пациентов, но и решить проблему негативного воздействия облучения на их здоровье. При снижении дозы агрессивного излучения полученные снимки могут отличаться низкой контрастностью, наличием шумов и размытостью [2].

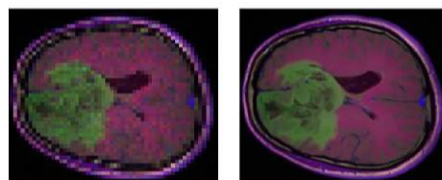


Рис. 1. Пример искусственно полученной пары изображений (LR , HR)

В данной работе рассматриваются автоматизированные техники повышения качества медицинских изображений, с помощью которых оценивается их влияние на эффективность сегментации опухолей. В частности, предлагается алгоритм повышения разрешения на основе генеративно-состязательной сети [3] и искусственно полученных для каждого изображения пар (LR , HR) [4], где LR (Low Resolution) – изображение в низком разрешении, HR (High Resolution) – в высоком разрешении (см. рис. 1). Ставится эксперимент по оценке качества

Кванчiani Карина Шавкановна – студент, Санкт-Петербургский государственный университет; e-mail: kvanchiani_karina@mail.ru, тел.: +7(921)649-65-64

сегментации на сгенерированных изображениях высокого разрешения и на изображениях в низком разрешении.

2. Задача повышения разрешения изображения. Пусть имеются пары изображений (объект, метка): (LR, HR) ($LR = D(HR)$, D – функция деградации). Необходимо построить функцию

$$F(LR) = D^{-1}(LR) + r,$$

где r – остаток реконструкции. Тогда сгенерированное изображение $HRg = F(LR)$.

Классическим методом решения поставленной задачи считается использование сверточных нейронных сетей [4]. В качестве подхода к обучению нейронной сети в данной работе используется генеративно-состязательная сеть [3], необходимая для оценки генеративных моделей через состязательный процесс, в котором одновременно обучаются две нейронные сети: генератор G и дискриминатор D .

В качестве основы модели генератора использовалась классическая архитектура генеративной сети, состоящая из 23 остаточных блоков, каждый из которых состоит из сверточных слоев с последующим применением блока линейной ректификации. Модель дискриминатора представляет собой полностью сверточную нейронную сеть, состоящую из восьми сверточных слоев, к каждому из которых применяется батч нормализация и блок линейной ректификации.

Результатом применения описанной генеративной сети является получение трех тензоров с соответствующими медицинскими изображениями: в первом (LR) – искусственно полученные изображения в низком разрешении (рис. 2а), во втором (HRg) – сгенерированные изображения в высоком разрешении (рис. 2б), в третьем (HR) – изначальные изображения в высоком разрешении (рис. 2с).

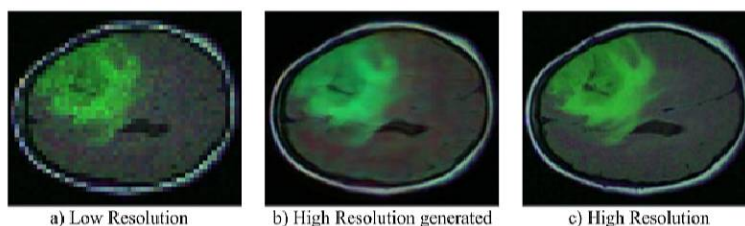


Рис. 2. Пример работы алгоритма повышения качества изображений

3. Задача сегментации. В качестве алгоритма сегментации была использована Unet-образная архитектура нейронной сети. Обучение данной сети производилось на тех же изображениях в высоком разрешении, на которых обучалась и генеративно-состязательная сеть, но пара (объект, метка) теперь представлена в виде $(HR, mask)$, где $mask$ – соответствующая изображению HR маска.

Тестирование обученной нейронной сети производилось на трех наборах полученных после генерации снимков. Снимки из множества HR использовались в качестве эталонных данных для оценки множеств LR и HRg .

4. Эксперимент. Для оценки эффективности использования техник повышения качества медицинских изображений в задаче сегментации опухолей был поставлен эксперимент, в рамках которого на тестовом наборе данных требовалось получить значения метрики качества IoU (Intersection over Union) [5].

4.1. Датасет. В качестве тестового набора данных использовался датасет Brain MRI segmentation [6]. Данный набор содержит 1373 снимка головного мозга и соответствующие им 1373 маски опухолей. Каждое изображение имеет один и тот же размер – 256×256 пикселей.

4.2. Подготовка данных. В медицинской визуализации существуют две основные проблемы – недостаточное количество данных и их качество. Первая проблема решается с помощью искусственного увеличения набора данных путем аугментации, которая положительным образом влияет на устойчивость алгоритма сегментации к разнообразиям схожих изображений. Вторая проблема часто решается за счет применения методов машинного обучения с учителем для задачи повышения разрешения. С этой целью необходимо получить снимки в низком разрешении из имеющихся снимков в высоком разрешении. Для этого в работе предприняты следующие характерные для медицинских снимков трансформации: подвержение уменьшающей дискретизации в четыре раза, размытие снимка и добавление спекл-шума.

В результате применения процесса аугментации набор данных увеличился в два раза: 2746 снимков в высоком разрешении и столько же в низком. Набор данных, состоящий из 2746 пар, был разделен на тренировочную и тестовую выборки, 80% и 20% соответственно. Оптимальный размер батча составил 32 объекта.

4.3. Настройка параметров генеративно-сопоставительной сети. Обучение генератора и дискриминатора основано на методе обратного распространения ошибки с использованием оптимизатора Adam. Значение параметра скорости обучения подбиралось экспериментальным путем, всего было рассмотрено два варианта: (0,0002; 0,0002), (0,008; 0,001), где первое значение – скорость обучения дискриминатора, второе – генератора. Обучение производилось на протяжении 9 эпох. На рис. 3 приведены графики функции потерь для двух описанных случаев. Видно, что второй вариант оказался более стабильным.

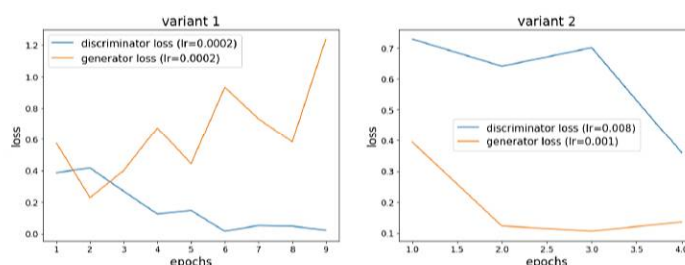


Рис. 3. Графики функции потерь дискриминатора и генератора

4.4. Результаты эксперимента. В таблице приведены значения метрики IoU для двух тестовых наборов данных. На рис. 4 представлен пример визуализации численных результатов (изображение HR , соответствующая ему маска, маски, сгенерированные из изображений LR , HRg и HR соответственно, числа – значения метрики на представленных данных). Видно, что качество сегментации опухоли на отдельном изображении, сгенерированном алгоритмом повышения разрешения, превосходит качество на том же изображении в низком разрешении и, в данном случае, на эталонном изображении.

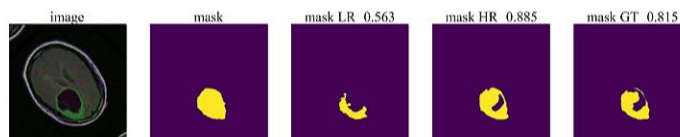


Рис. 4. Пример визуализации численных результатов

Таблица. Значение метрики на тестовых наборах данных

IoU	LR	generated HR	ground truth HR
test 1	0,741	0,77	0,843
test 2	0,739	0,777	0,844

Из таблицы видно, что использование методов повышения разрешения изображений улучшило качество сегментации опухолей в среднем на 3,5% по сравнению с подходом, не учитывающим предобработку данных.

5. Заключение. В работе разработан метод повышения разрешения медицинских изображений, использование которого позволяет повысить качество сегментации опухолей (по метрике IoU) на 3,5%. В дальнейшем планируется применить полученный подход к реальным данным на примере кардиологических снимков.

Литература

1. Botha C. P., Preim B., Kaufman A. et al. From individual to population: challenges in medical visualization // Scientific Visualization. London: Springer, 2014. P. 265–282.
2. Qingsong Y., Pingkun Y., Zhang Y. et al. Low dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss // IEEE Transactions on Medical Imaging. 2018. No 6. P. 1348–1357.
3. Goodfellow I. J., Pouget-Abadie J., Mirza M. et al. Generative adversarial nets // Advances in Neural Information Processing Systems. Berlin: Springer, 2014. P. 2672–2680.
4. Wang Z., Chen J., Hoi S. C. H. Deep learning for image super-resolution: a survey // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2020. P. 1–1.
5. Rezatofighi H., Tsoi N., Gwak J. et al. Generalized intersection over union: a metric and a loss for bounding box regression // IEEE CVPR. 2019. P. 658–666.
6. Buda M. Brain MRI segmentation [Электронный ресурс]: URL: <https://www.kaggle.com/mateuszbeda/lgg-mri-segmentation> (дата обращения: 10.09.2020).

УДК 004.93'12

Князев Н. А.

Определение пола и возраста людей по фотографии лица с использованием нейронных сетей

Рекомендовано к публикации доцентом Козыиченко В. А.

1. Введение. Определение пола и возраста является популярной задачей машинного обучения, и в настоящий момент существуют различные подходы к ее решению [1, 2]. Решение данной задачи имеет множество практических приложений. К примеру, появляется возможность быстрого принятия решения о допуске лиц на мероприятие, имеющее возрастную цензу, или на продажу им некоторых товаров. В области маркетинга и анализа статистики сегментирование посетителей по социально-демографическому признаку позволяет лучше понять состояние рынка и предпочтения пользователей.

В работе предлагается решить задачи определения пола и возраста, используя нейросетевые классификаторы, и сравнить модели.

2. Постановка задачи. Определение пола человека по фотографии относится к задаче бинарной классификации. Предсказание возраста – это регрессионная задача, но мы решим ее как задачу классификации, путем группировки по различным возрастным диапазонам.

В качестве базы используем набор данных Wikipedia (Wiki) [3], а именно вариант набора, содержащий около 62 000 обрезанных фотографий лиц людей. Распределение возрастов и пола представлены на рис. 1. Разделим выборку на обучающую (70%), тестовую (20%) и валидационную (10%). Для оценки качества моделей будем использовать стандартные метрики классификации (accuracy, recall, F-мера). Для многоклассовой модели в качестве данных метрик будет вычисляться среднее значение соответствующих показателей, где конкретный класс будет считаться положительным классом, а все другие – отрицательным.

Князев Никита Андреевич – студент, Санкт-Петербургский государственный университет; e-mail: st064730@student.spbu.ru, тел.: +7(981)791-77-40

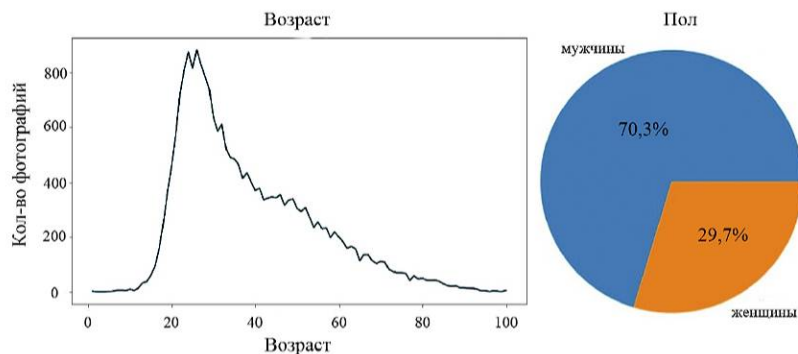


Рис. 1. Распределение пола и возраста Wiki

3. Алгоритм решения. В качестве основы для решения поставленных задач используем предобученную сверточную нейронную сеть, а именно архитектуру VGG-Face [4] (рис. 2), так как данная модель настроена на распознавание лиц.

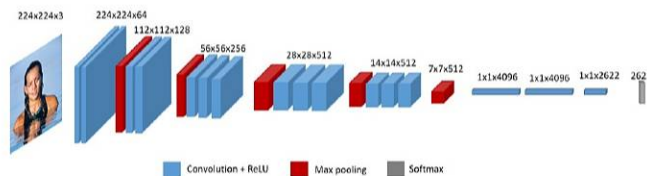


Рис. 2. Архитектура VGG-Face

3.1. Предобработка данных. На некоторых фотографиях Wiki нет людей, или на них изображены несколько человек. Для части людей указан возраст более 100 лет или меньше 0, либо информация о возрасте и вовсе отсутствует. Очистим датасет от подобных фотографий и ограничим диапазон возраста от 0 до 100 лет. Преобразуем фотографии под размер 224 × 224 пикселей.

3.2. Определение пола и возраста одной моделью. Используем один классификатор для распознавания пола и возраста вместе. Зафиксируем веса для первых 18 слоев сети. Также разобьем количество классов в последнем слое свертки на шесть: два для по-

ла и четыре для возраста, используя возрастные периоды Эриксона: 15–30, 31–40, 41–60, 61–80. Зададим параметры обучения: количество эпох – 20, batch size – 256, шаг обучения – 10^{-3} .

3.3. Определение пола. В отдельности распознавание пола довольно простая задача классификации. Прделаем похожие действия, как в пункте 3.2. Однако в последнем слое свертки нам нужно всего два признака для прогнозирования пола. Параметры обучения оставим такими же.

3.4. Определение возраста без разделения по гендерному признаку. Сбалансируем обучающую, тестовую и валидационную выборку таким образом, чтобы мужчин и женщин в них было примерно равное количество. Обучим сеть как указано в пункте 3.2, за исключением количества классов. Для решения данной задачи в выходном слое модели укажем четыре возрастных диапазона. Зададим параметры обучения: количество эпох – 20, batch size – 256, шаг обучения – 10^{-5} . График обучения представлен на рис. 3.

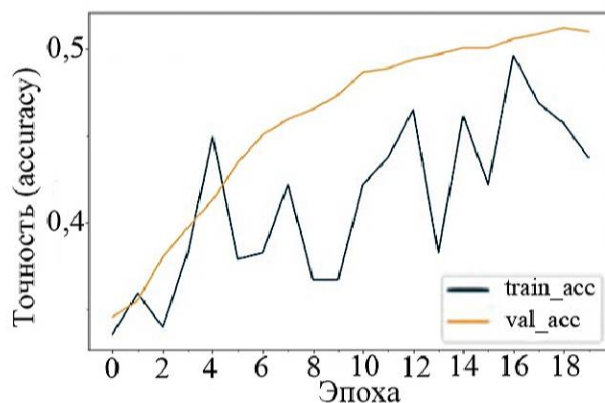


Рис. 3. График обучения модели определения возраста без разделения по гендерному признаку

3.5. Определение возраста с разделением по гендерному признаку. Предполагается, что сначала будет определяться пол, а затем – возраст.

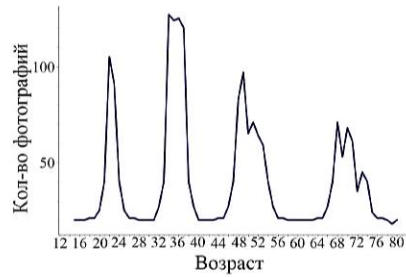


Рис. 4. Модифицированное распределение данных для мужчин

Разделим выборку фотографий на две части: содержащие только мужчин и только женщин. Наборы сбалансируем по количеству изображений для каждого возрастного класса и сделаем распределение по возрастам близким к гауссову (рис. 4). К сожалению, набор Wiki содержит мало фотографий с женщинами, поэтому увеличим датасет путем поворота фотографий влево и вправо на небольшой угол и отражения по горизонтали. Увеличим число эпох до 240 для женщин и 160 для мужчин, остальные параметры обучения оставим неизменными. Графики обучения представлены на рис. 5.

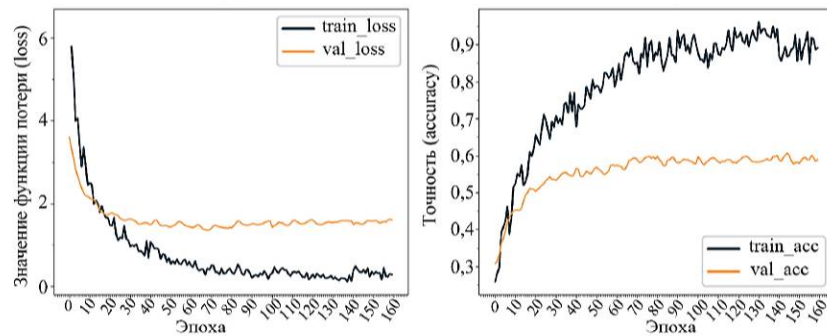


Рис. 5. Графики обучения модели определения возраста, обученной на фотографиях, содержащих только мужчин

4. Реализация. Нейронные сети реализованы на языке Python с использованием фреймворка TensorFlow. Обучение сетей проводилось методом обратного распространения ошибки с оптимизатором

Adam в Amazon с помощью сервиса SageMaker на ml.p2.xlarge.

5. Результаты. Результаты проведенных вычислительных экспериментов, а также размеры тренировочных и тестовых множеств представлены в таблице.

Таблица. Результаты

Модель	Train/Test	Accuracy	Recall	F-мера
Пол и возраст	9000/2600	Пол: 0,95, Возраст: 0,41	Пол: 0,96, Возраст: 0,42	Пол: 0,96, Возраст: 0,41
Пол	9000/2600	0,95	0,96	0,96
Возраст	9000/2600	0,52	0,45	0,45
Возраст, мужчины	1500/1000	0,59	0,6	0,6
Возраст, женщины	2400/800	0,63	0,64	0,63

Можно сделать вывод, что задачи определения пола и возраста эффективней решать по отдельности, предварительно разбив датасет по гендерному признаку и сбалансировав его по количеству фотографий для каждого возрастного класса.

6. Заключение. В работе представлены архитектуры сверточных нейронных сетей для решения задач распознавания пола и возраста человека по фотографии лица разными способами, а также результаты вычислительных экспериментов.

Литература

1. Пакулич Д. В., Якимов С. А., Алямкин С. А. Распознавание возраста по изображению лица с использованием сверточных нейронных сетей // Автометрия. 2019. Т. 55. Вып. 3. С. 52–61.
2. Apparent Age and Gender Prediction in Keras [Электронный ресурс]: URL:<https://sefiks.com/2019/02/13/apparent-age-and-gender-prediction-in-keras/> (дата обращения: 10.02.2021).
3. IMDB-WIKI – 500k+ face images with age and gender labels [Электронный ресурс]: URL:<https://data.vision.ee.ethz.ch/cv1/rrothe/imdb-wiki/> (дата обращения: 10.02.2021).
4. VGG-Face model for Keras [Электронный ресурс]: URL:<https://gist.github.com/EncodeTS/6bbe8cb8bebad7a672f0d872561782d9> (дата обращения: 10.02.2021).

УДК 004.93

Кульминский Д. П., Ефимов В. В.

Semi-supervised сегментация с использованием нейросетей в задаче оценки кондиционности керна

Рекомендовано к публикации доцентом Гришкиным В. М.

1. Введение. Керн – образец горной породы цилиндрической формы. По такому образцу можно получить информацию о геологическом строении недр, вещественном составе горных пород, наличии или отсутствии в них углеводородов. Однако, прежде чем транспортировать образец в кернохранилище или лабораторию, необходимо точно оценить кондиционность керна, т. е. его пригодность для взятия пробы и дальнейшего лабораторного анализа. Некондиционными считаются участки с обилием трещин техногенного (получаются при извлечении из скважины или транспортировке) или естественного происхождения (свойство самой породы) или наличием крупных разломов. Эти участки не участвуют в последующих исследованиях. Соответственно, задача заключалась в разработатке инструмента, выполняющего семантическую бинарную сегментацию кондиционности горной породы, т. е. присваивающего каждому пикселю принадлежность к одному из классов: «кондиционный» или «некондиционный».

Традиционно в задаче семантической сегментации в качестве обучающей выборки используются пары из оригинальных изображений и масок. Последние представляют собой изображение, совпадающее с оригиналом по размерности, каждый пиксель которого имеет метку одного из классов, т. е. имеет место переход из какого-либо цветового пространства во множество меток классов. Обычно маски являются результатом ручной разметки специалистов предметной области. Объем выборок из таких пар варьируется от нескольких сотен до

Кульминский Данил Петрович – студент, Санкт-Петербургский государственный университет; e-mail: st064144@student.spbu.ru, тел.: +7(922)194-01-88

Ефимов Владислав Владимирович – студент, Санкт-Петербургский государственный университет; e-mail: st073936@student.spbu.ru, тел.: +7(911)112-58-89

нескольких тысяч примеров. Ввиду того, что предоставить несколько тысяч масок с достаточно чувствительной разметкой кондиционных и некондиционных сегментов за разумное время не представлялось возможным, вынужденной мерой стало использование вместо детальной попиксельной разметки более грубой площадной, представляющей из себя обобщенную оценку качества на том или ином крупном сегменте породы.

2. Метод сегментации изображений керна. Предложенный подход состоял из следующих этапов (см. рис. 1):

1. По исходному изображению составлялась маска классов пород с использованием обученного классификатора, применяемого на окнах, и SS-CAM (Smooth Score Class Activation Map) обученного отдельного классификатора кондиционности [1].
2. Маска SS-CAM бинаризовалась применением пороговой функции попиксельно в зависимости от классов пород.
3. На полученных «грубых» бинарных масках сегментации обучалась модель семантической сегментации U-Net, позволяющая получить финальный результат сегментации в один этап.

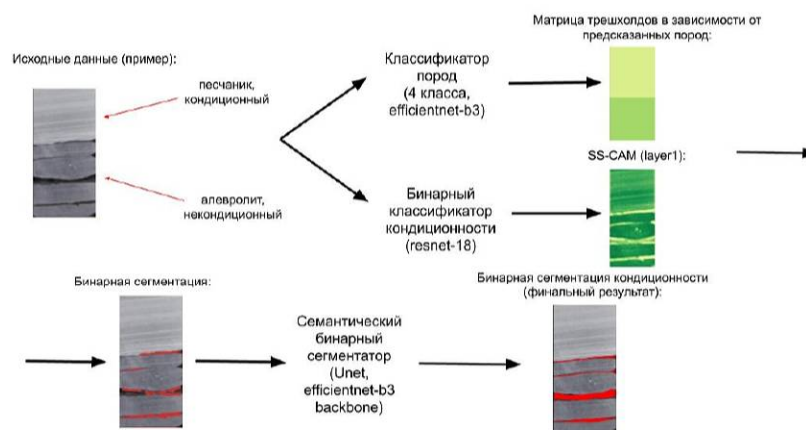


Рис. 1. Схема этапов обучения алгоритма

2.1. Классификатор пород. Задачей классификатора пород было определять следующие классы: «алевролит», «песчаник», «переслаивание пород», «аргиллит». В качестве основы была использована предобученная нейросетевая модель efficientnet-b3 [2].

Предобработка данных для обучения состояла из следующих этапов:

1. Пропорциональное приведение исходного изображения к фиксированной ширине в 256 пикселей.
2. Выбор в качестве сэмплов прямоугольных областей изображения высотой в 512 пикселей и шириной в 256, 50% площади которых принадлежат одному классу породы.
3. Замена черным пикселей, не принадлежащих к целевому классу, т. е. тому, который занимает 50% и более от всей площади сэмпла.

На полученном множестве сэмплов сеть архитектуры efficientnet-b3 дообучалась с использованием алгоритма оптимизации AdamW [3] и кросс-энтропии в качестве функции потерь.

В силу специфики предметной области переиспользование моделей на изображениях иного масштаба крайне нежелательно. Причиной тому является значимость размера вкраплений в качестве визуального признака при определении породы литологических образцов.

В связи с этим в процессе использования модели исходные изображения аналогично предобработке на этапе обучения приводились к ширине в 256 пикселей. Далее изображения разбивались по вертикали на окна высотой в 512 пикселей с пересечением в 60%. Полученные окна использовались в качестве входных данных для модели, на пересечениях вектор вероятностей классов усреднялся.

2.2. SS-CAM бинарного классификатора кондиционности. Поскольку точность разметки не позволяла явно обучить на ней сегментатор кондиционности, для «грубой» сегментации использовалась SS-CAM первого сверточного слоя классификатора кондиционности. Она показывала значимость каждого пикселя в задаче определения класса «кондиционный» или «некондиционный».

Предобработка изображений на этапе обучения была аналогична описанной для классификатора пород. В качестве классификатора использовалась обученная с нуля архитектура resnet-18 [4] с

использованием оптимизатора AdamW и бинарной кросс-энтропией в качестве функции потерь.

На этапе использования SS-CAM классификатора исходные изображения так же приводились к ширине в 256 пикселей. Далее изображения разбивались на окна высотой в 512 пикселей и шириной в 256 с наложением в 60% как по горизонтали, так и по вертикали. Для устранения артефактов крайние 20 пикселей SS-CAM не использовались, для остальных пикселей на пересечениях окон значения соседних SS-CAM усреднялись. Для получения бинарной маски сегментации использовался установленный в зависимости от класса породы порог. На полученной бинарной маске удалялись все сегменты с площадью менее 400 пикселей.

2.3. Нейросеть семантической сегментации. Описанный выше подход является весьма требовательным в плане вычислительных ресурсов – SS-CAM подразумевает итерации по всем фильтрам заданного сверточного слоя для построения оценок важности пикселя исходного изображения для заданного класса. В связи с этим было решено использовать результат работы двух классификаторов для формирования новой обучающей выборки для сегментирующей сети архитектуры U-Net [5], перейдя таким образом к известной задаче бинарной семантической сегментации. Используя пары оригинальных изображений и бинаризованных масок SS-CAM, алгоритм оптимизации AdamW и взвешенную кросс-энтропию в качестве функции потерь (большой штраф за ошибки на пикселях, соответствующих повреждениям, а не куску цельной породы), удалось обучить U-Net распознаванию некондиционных областей на фотографии (метрика сходства предсказаний сегментирующей модели и масок SS-CAM с бинарного классификатора кондиционности, Intersection over union, достигла значения 0,7).

3. Анализ результатов. Предложенный подход позволил добиться решения высокой точности в задаче сегментации без соответствующей разметки. Результат был достигнут благодаря использованию всего объема доступных данных, в том числе разметки пород и информации от экспертов предметной области, позволившей подобрать пороговые значения для алгоритма на первом этапе «грубой» сегментации.

Использование U-Net для построения финального решения существенно повысило скорость обработки изображений.

4. Заключение. В работе продемонстрирован способ решения задачи семантической сегментации в условиях отсутствия детальной попиксельной разметки. Используя технику извлечения карт активаций по классам со сверточных слоев сетей, обученных на классификацию крупных сегментов, возможно либо решить задачу напрямую (после некоторой постобработки), либо на их основе обучить одну из распространенных сегментирующих архитектур.

Литература

1. Haofan Wang, Rakshit Naidu, Joy Michael, Soumya Snigdha Kundu SS-CAM: Smoothed Score-CAM for Sharper Visual Feature Localization // arXiv preprint arXiv:2006.14255. 2020 [Электронный ресурс]: URL:<https://arxiv.org/abs/2006.14255> (дата обращения: 10.10.2020).
2. Mingxing Tan, Quoc V. Le EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks // arXiv preprint arXiv:1905.11946. 2020 [Электронный ресурс]: URL:<https://arxiv.org/abs/1905.11946> (дата обращения: 15.10.2020).
3. Loshchilov I., Hutter F. Decoupled Weight Decay Regularization // arXiv preprint arXiv:1711.05101. 2019 [Электронный ресурс]: URL:<https://arxiv.org/abs/1711.05101> (дата обращения: 14.10.2020).
4. Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun Deep Residual Learning for Image Recognition // arXiv preprint arXiv:1512.03385. 2015 [Электронный ресурс]: URL:<https://arxiv.org/abs/1512.03385> (дата обращения: 05.10.2020).
5. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation // arXiv preprint arXiv:1505.04597. 2015 [Электронный ресурс]: URL:<https://arxiv.org/abs/1505.04597> (дата обращения: 14.09.2020).

УДК 004.415.22

Меньшиков М. А.

Систематизация абстракций для поддержки различных языков программирования и сред в статических анализаторах кода

Рекомендовано к публикации профессором Тереховым А. Н.

1. Введение. Инструменты поддержки написания программ постепенно перерастают свою узкоспециализированную направленность и становятся элементами *комплексных* технологий программирования, для которых характерна возможность работать с программным кодом на любых языках и в любых средах, предоставляя разработчику расширенные возможности по принятию архитектурных решений. Анализаторы программного кода часто поддерживают два и более языков программирования, но, как правило, для них различаются лишь процедуры чтения исходного кода, а различия целевых сред игнорируются. Чтобы достичь хороших результатов при анализе в такой единой технологии, требуются удобные *абстракции*. Под *абстракцией* здесь и далее подразумевается набор структур данных и методов, необходимых для обобщения какого-либо процесса. В данной статье рассмотрена возможность систематизировать абстракции по принципам явности или неявности, необходимости и эффективности. Также приводится правило выбора оптимального набора свойств для подбора абстракций, описание практической пользы от использования подхода, результаты его апробации.

Актуальность работы основывается на текущем состоянии развития технологий программирования, при котором одновременная работа с множеством сред становится необходимостью.

Множество работ посвящено анализу программ на различных языках программирования в общем ключе, и наиболее явное значение с практико-теоретической точки зрения имеет LLVM IR [1], промежуточное представление для компиляции и анализа программ, на основе которого реализованы анализаторы под различные языки программирования. Проект MLIR [2] решает часть проблем LLVM,

Меньшиков Максим Александрович – аспирант, Санкт-Петербургский государственный университет; e-mail: info@menshikov.org, тел.: +7(921)766-41-38

в том числе и вопрос поддержки программ на различных языках. GraalVM [3] – мультиязычный рантайм, предоставляющий возможности по анализу кода. Аналогичной системой является Roslyn [4], который имеет общие анализаторы для различных языков программирования, но с разными трансляционными действиями.

В более теоретическом ключе рассматриваются проблемы композиции семантик в статьях [5, 6], в которых авторы предлагают математическую теорию комбинирования абстрактных интерпретаций существующих языков. Операционная семантика многоязыковых программ рассматривается в [7].

2. Принцип явности и неявности. Абстракции можно явным образом разделить по принципу явности и неявности. Если рассматривать условную схему анализа программ (трансляция, установление взаимосвязей между объектами, непосредственно анализ), то можно заметить, что часть этапов явно прослеживается в ходе анализа, а часть происходит в фоновом режиме. Явность в данном случае зависит от того, насколько объекты или процессы видны в исследуемых программах.

На рис. 1 представлены возможные типы абстракций относительно данного принципа. Рассмотрим деление более подробно.

2.1. Явные трансляционные действия. Под *явными трансляционными действиями* следует понимать процедуры трансляции, которые безусловно должны быть реализованы для целевого языка. К таковым относятся:

- трансляция выражений и управляющих конструкций во внутренние объекты и, при необходимости, обратно в выражения и управляющие конструкции;
- отображение типов во внутренние объекты и обратно. Пусть T_l – совокупность квалифицированных типов, определенная в терминах исходного языка. Тогда T – совокупность типов в терминах анализатора. Отображение $F_T : T_l \rightarrow T$, в общем случае, будет небиективным (в зависимости от реализации оно может быть сюръекцией или инъекцией). В вырожденном случае при реализации под один конкретный язык эти множества могут совпадать ($T_l = T$), порождая биективное отображение.



Рис. 1. Схема абстракций по принципу явности и неявности

В некоторых парсерах могут быть неопределенные структуры (например, `va_list_tag` в Clang). Объявление таких структур тоже можно отнести к явным трансляционным действиям, так как их существование явно обозначено стандартом языка.

На практике эта группа реализуется всеми инструментами в той или иной степени, так как без трансляции невозможно провести дальнейший анализ.

2.2. Явные абстракции для поддержки семантики. Данные абстракции представляют собой сущности, определенные в коде исходной программы. Их хранение в большинстве императивных языков осуществляется в тех или иных пространствах имен (для C++ примером может быть `std`, для Java – полное название модуля, например, `org.example.tool`). Аналогично предыдущей группе, такие абстракции обязаны существовать вследствие невозможности провести анализ программ без них.

2.3. Неявные абстракции для поддержки семантики. В данный подраздел можно включить все операции, которые явным образом не описаны в программах на целевом языке, но применяют-

ся анализаторами, кодогенераторами, оптимизирующими моделями и т. п.

Примерами могут служить операции с примитивными типами данных. К ним относятся правила сдвига битов, проведения элементарных операций (сложность возникает, например, при передаче аргументов при работе с плавающей арифметикой в C/C++).

Фактическая структура типов тоже немаловажна при проведении анализа на низком уровне, а не на уровне полей структур. Выделяются правила упаковки структур, битовых полей, а также наличие или отсутствие выравниваний.

Все компиляторы реализуют эти абстракции, но среди анализаторов они распространены в меньшей степени. Это связано с трудоемкостью работы с низкоуровневым представлением данных, и зачастую разработчики анализаторов ограничиваются поддержкой только уровня полей структур.

2.4. Внешние свойства. Компилятор, настройки языковой среды, семантика среды исполнения, характеристики среды исполнения – все эти детали безусловно влияют на точность и полноту анализа. Эти свойства являются как явными, так и неявными, но в предлагаемой классификации они занимают одно место, так как явные с точки зрения спецификации среды могут быть неявными для пользователя, занимающегося написанием программы.

В качестве примеров влияния внешних свойств можно привести следующие: разрядность центрального процессора целевой архитектуры явно влияет на размер типа указателя в программе; реальная целевая архитектура может отличаться от архитектуры, для которой генерируется объектный код; порядок байтов имеет низкое влияние при отсутствии прямого доступа к памяти или передачи данных, но генерирует большее количество соответствующих ошибок в сетевом коде. Любая используемая операционная система предполагает особую семантику системных вызовов, которая может вызывать определенного рода ошибки при работе.

3. Достижимость идеальной абстракции. Идеальный анализатор программного кода должен максимально абстрагироваться от языковых деталей и особенностей среды, но это трудно достижимо в реальных проектах. Пусть $p \in P$ – единичное свойство языка программирования или среды. Условимся считать P^T множеством свойств языков и сред, поддерживаемых конкретным инструментом.

Тогда P_1, \dots, P_n – множества свойств различных комбинаций языков программирования и сред (т. е. $1, \dots, n$ – комбинации языков и сред). Здесь можно столкнуться с проблемой выбора степени абстракции. С одной стороны, можно выбрать такой набор P^T , который бы поддерживал все свойства языков: $P^T = \{p | p \in P_1 \cup \dots \cup P_n\}$. Для простоты запишем данное объединение как P_{max} . Количество рассматриваемых свойств может быстро расти с увеличением числа языков программирования, и тогда возможность поддержать все свойства выглядит сомнительной как с вычислительной точки зрения, так и с точки зрения времени реализации. С другой стороны, рассмотрение пересечения свойств $P^T = \{p | p \in P_1 \cap \dots \cap P_n\}$ (назовем его P_{min}) приводит лишь к разработке неточных анализаторов кода. Оптимальным решением является выбор абстракции, включающей и пересечение свойств, и некоторый набор свойств из объединения:

$$P^T = \{p | p \in P_{min} \vee p \in P_{max}\}.$$

Возникает вопрос, как именно произвести выбор свойств.

4. Необходимые и эффективные свойства как часть оптимальной абстракции. Несмотря на сложности достижения иде-



Рис. 2. Диаграмма Венна для наборов свойств

альной абстракции, некоторую оптимальную для статического анализатора абстракцию получить все же возможно. Для этого достаточно рассмотреть только две категории свойств: *необходимые* и *эффективные*. Под необходимыми подразумеваются те свойства,

которые пересекаются между языками и средами. Под эффективными свойствами будем подразумевать те, на которые распространяются правила обнаружения ошибок. Можно заметить, что первую группу составляет именно пересечение множеств свойств:

$$P_1^T = \{p | p \in P_{min}\}.$$

Вторую группу представляет набор P_2^T , частично пересекающийся с набором P_{min} :

$$P_2^T = \{p | p \in P_{min} \vee p \in P_{max}\}.$$

Тогда $P^T = P_1^T \cup P_2^T$ – это набор, который можно назвать *оптимальным* по данной классификации, так как он обеспечивает анализатор данными о существенно отличающихся свойствах, а также о некоторых других свойствах, не акцентируя внимание на тех, чья поддержка отсутствует в анализаторе.

В дальнейшем необходимо подобрать такие абстракции, которые позволяют различить численные представления данных свойств между средами и языками программирования. Пусть P_i^T для $i \in [1, n]$ – некоторые численные представления данных свойств для среды (комбинаций языков и сред) i , а f_j – одна из абстракций ($j \in [1, m]$, где m – общее число абстракций). Тогда результат хотя бы одной абстракции должен отличаться для двух разных сред:

$$\forall i_1, i_2 \in [1, n], \quad i_1 \neq i_2, \quad \exists j \in [1, m], \quad f_j(P_{i_1}^T) \neq f_j(P_{i_2}^T).$$

Выполнение данного условия может служить одним из критериев выбора абстракции.

Отношения между множествами свойств проиллюстрированы на рис. 2.

5. Практическое значение разделения абстракций. Предлагаемое разделение абстракций полезно с практической точки зрения. Необходимые абстракции должны реализовываться напрямую в анализаторе кода и иметь максимальную производительность, в то время как строго эффективные абстракции (немного реже используемые) могут реализовываться в составе библиотек и модулей, внешних по отношению к основному модулю.

Явность или неявность абстракций позволяет ограничить количество сведений, доступных пользователю статического анализатора в отчетах о дефектах. Явные объекты появляются вследствие ввода пользователя, что означает необходимость сохранения данных о принадлежности объектов конкретным строкам исходного текста. Неявные объекты важны для анализа, но они могут не иметь явной привязки к исходному коду, следовательно, это позволяет проектировать абстракции с меньшим количеством взаимосвязей и, соответственно, уменьшать вычислительную сложность.

6. Апробация. Неполная реализация идеи выбора абстракций реализована в статическом анализаторе программного кода Equid [8]. Наиболее важные абстракции, реализованные в проекте, приведены в таблице.

Таблица. Абстракции и соответствующие типы свойств

Абстракция	Свойства	Тип
Компилятор	Размеры типов	Явн. трансл. / необх.
Процессор	Размеры типов	Явн. трансл. / необх.
Языковая семантика	Выбор типов выражений, эквивалентность типов	Явн. сем. / необх.
Пространство имен	Способ хранения язык. объектов	Явн. сем. / необх.
База шаблонов типов	Структура типов	Неявн. сем. / необх.
База низкоуровневых типов	Структура типов и операции с ними	Неявн. сем. / необх.
Семантика ОС	Семантика методов, модели объектов и т. п.	Среда / эффект.
Контекстуальные семантики	Контексты исполнения, машины состояний среды и т. п.	Среда / эффект.
Виртуальная машина	Модель потока исполнения	Среда / эффект.
Гипервизор	Модель многопоточной ЭВМ	Среда / эффект.

Под необходимыми абстракциями в проекте подразумеваются трансляционные действия, сравнения типов (осуществляемые при разрешении имен), механизмы выбора типов и т. п. Эффективные свойства, нашедшие отражение в абстракциях: текущее состояние аллокатора памяти, состояния сокетов и файловых дескрипторов и т. п.

Такое разделение абстракций позволяет вынести данные о внешней среде в отдельные файлы или библиотеки, указывает на необходимость оптимизации наиболее загруженных фрагментов. В частности, ведется постоянная работа по оптимизации работы с типами (чтение и разрешение имен) и пространством имен (хранение объектов и их поиск). Также можно выгружать объекты редко используемых эффективных абстракций. При этом моделирование важных свойств программ получается точным и полным.

На практике, своевременное разделение абстракций позволяет быстро прототипировать поддержку новых языков и сред. Добавление поддержки C++ в описываемом проекте было произведено в довольно сжатые сроки и существенно быстрее поддержки C. Следует отметить, что изначально допущенные ошибки архитектуры (например, отсутствие полноценных пространств имен, технически мало пригодных для языка C) привели к серьезному увеличению времени реализации, что подтверждает важность реализации анализаторов в соответствии с классификацией.

Одна из возможных проблем подхода, проявившаяся при апробации, это отсутствие понимания, к какой категории отнести абстракцию. Поиск ответа на данный вопрос для неосновных категорий абстракций может и не нести практической пользы, так как категория может получиться смешанной. В таком случае следует провести ряд экспериментов и установить, какой способ реализации абстракции больше подходит для конкретного случая.

7. Заключение. В работе представлена классификация абстракций, необходимых для поддержки различных императивных языков программирования, по принципам явности или неявности, необходимости и эффективности. Приведен метод выбора оптимального набора свойств для абстракций и объяснено практическое значение описанного разделения. Проведена апробация метода, которая показала преимущества подхода и возможные проблемы при его использовании.

Литература

1. Lattner C., Adve V. LLVM: A compilation framework for lifelong program analysis & transformation // International Symposium on Code Generation and Optimization. IEEE, 2004. P. 75–86.

2. MLIR: A Compiler Infrastructure for the End of Moore’s Law [Электронный ресурс]: URL: <https://arxiv.org/abs/2002.11054> (дата обращения: 13.03.2021).
3. Introduction to GraalVM [Электронный ресурс]: URL: <https://www.graalvm.org/docs/introduction> (дата обращения: 13.03.2021).
4. Roslyn. The .NET Compiler Platform [Электронный ресурс]: URL: <https://github.com/dotnet/roslyn> (дата обращения: 13.03.2021).
5. Buro S., Crole R. L., Mastroeni I. On multi-language abstraction // International Static Analysis Symposium. Cham: Springer, 2020. P. 310–332.
6. Buro S., Mastroeni I. On the multi-language construction // European Symposium on Programming. Cham: Springer, 2019. P. 293–321.
7. Matthews J., Findler R. B. Operational semantics for multi-language programs // ACM SIGPLAN Notices. 2007. Vol. 42. No 1. P. 3–10.
8. Menshikov M. Equid – a static analysis framework for industrial applications // International Conference on Computational Science and Its Applications. Cham: Springer, 2019. P. 677–692.

УДК 004.051

Николаева И. Н.

Минимизация размера образов контейнеров приложений на примере Docker

Рекомендовано к публикации доцентом Ганкевичем И. Г.

1. Введение. Контейнер – это стандартизированная часть программного обеспечения, которая содержит в себе приложение и все его зависимости, чтобы запускать приложение быстро в отдельном окружении от остальных приложений [1]. У каждого запущенного таким образом приложения своя версия корневой файловой системы, списка процессов и списка сетевых устройств. Контейнеры приложений предоставляют безопасный способ запуска приложений без накладных расходов других технологий виртуализации.

Контейнер собирается из образа, который является неизменяемым шаблоном. Образы можно создавать, обновлять или использовать уже готовые. Образы собираются при помощи специальных файлов, в которых описывается, на основе какого базового образа будет создаваться новый образ, какие команды следует запустить внутри образа, какие файлы будут добавлены внутрь контейнера, точку запуска контейнера и многие другие параметры. Хранятся и передаются образы в основном через публичный реестр.

Ввиду того, что распространение и управление контейнерами происходит при помощи образов, остается существенным вопрос уменьшения размера образа для оптимизации занимаемого дискового пространства и пропускной способности сети.

Внутри контейнера работает ограниченное количество заранее известных программ, следовательно, список директорий и файлов, которые они используют, заранее известен. Однако образы корневых файловых систем для контейнеров создаются, как правило, менеджерами пакетов, которые были спроектированы для интерактивного использования человеком, а не для запуска конкретного приложения в изолированной среде. В результате в образы контейнеров входят программы, которые никогда не запускаются приложением

Николаева Ирина Николаевна – студент, Санкт-Петербургский государственный университет; e-mail: st064731@student.spbu.ru, тел.: +7(812)428-47-83

внутри контейнера, и файлы, которые никогда не читаются и не пишутся. Исключения из этого правила бывают, но они крайне редки.

Задача заключается в том, чтобы уменьшить размер образов корневых файловых систем, которые используются в контейнерах приложений. Цель данной работы заключается в том, чтобы определить с помощью встроенных в Linux средств отладки, какие именно файлы используются конкретными приложениями и удалить все остальные файлы из корневой файловой системы контейнера. Это позволит создать образы, которые в основном состоят из

- исполняемых файлов приложений;
- библиотек, которые используются этими исполняемыми файлами;
- набора конфигурационных файлов, используемых этими исполняемыми файлами и библиотеками.

Было произведено исследование материалов, посвященных тематике оптимизации хранения или доставки образов контейнеров. Существует решение, которое при помощи P2P сети уменьшает время доставки образов на более чем 90% [2]. Дедупликация позволяет увеличить переиспользование данных уже имеющихся образов, тем самым сокращая занимаемое дисковое пространство и время доставки [3, 4]. Организация доставки через разделяемую распределенную файловую систему позволяет до 12 раз сократить время на передачу [5]. Все эти подходы используют внешние, не зависящие от образа подходы и решают в основном конкретную задачу оптимизации.

Приложение `chainsaw` действует изнутри образа и с помощью `ptrace` (`ptrace` – системный вызов в системах Unix, позволяющий одному процессу управлять другим) перехватывает системные вызовы, которые работают с файлами. Из аргументов этих системных вызовов извлекает пути к файлам и сохраняет их.

Подобным образом действует приложение CARE, которое аналогично сканирует файловую систему и все элементы и переменные среды сохраняет в архив, с помощью которого можно добиться полного повторения условий, в которых была запущена программа [6]. В случае `chainsaw` используемые файлы не сохраняются в архив, а используются для того, чтобы вычислить неиспользуемые файлы и удалить их.

2. Работа с образами контейнеров на примере платформы Docker. Технология Docker на данный момент является самой распространенной. Для исследования взято приложение `chainsaw`, которое состоит из четырех частей [7]:

- `chainsaw-blacklist` – выводит лист всех файлов (в формате абсолютного пути) в корне файловой системы в файл `blacklist`, при этом исключаются домашние директории пользователей и виртуальные файловые системы (`/dev`, `/sys`, `/proc` и другие);
- `chainsaw-whitelist` – запускает указанную программу и сохраняет файлы, используемые ею, в файл `whitelist`;
- `chainsaw-diff` – выводит в файл `diff` разницу между `blacklist` и `whitelist`;
- `chainsaw-cut` – удаляет все файлы, указанные в `diff`.

2.1. Образ nginx. Веб-сервер и почтовый прокси-сервер `nginx` работает на Unix-подобных операционных системах.

В первых экспериментах было обнаружено, что `chainsaw-whitelist` не обнаруживает файлы, которые:

- отвечают за работу самого контейнера;
- являются исполняемыми файлами приложения;
- используются при работе приложения (для `nginx`, например, это статический `html`).

Поэтому было необходимо добавить в `whitelist` такие файлы и директории, как `/lib/x86_64-linux-gnu/ld-2.28.so`, `/lib64/ld-linux-x86-64.so.2`, `/usr/sbin/nginx` и `/var/run`.

2.2. Образ grafana. Для визуализации временных рядов используется веб-сервис `grafana`.

Были проведены те же действия по добавлению файлов и директорий в `whitelist`; их оказалось несколько больше – `grafana` значительно более объемное и сложное по архитектуре приложение.

Результаты, полученные после выполнения сжатия образа представлены в таблице.

Таблица. Сравнение размеров образов

Образ	Образ из реестра	Полученный образ
<code>nginx</code>	133 MB	8,19 MB
<code>grafana</code>	198,41 MB	148,01 MB

Размер директории (`/usr/share/grafana`), в которой лежит приложение и необходимые для его работы файлы, занимает в оригинальном контейнере 172,5 МВ. Ввиду того, что размер образа в результате стал меньше этого значения, можно сделать вывод, что уменьшение размера образа прошло успешно.

3. Выводы. Проведено исследование работы приложения `chain-saw` на примере двух образов контейнеров. Получено существенное сокращение памяти, занимаемой образом `nginx`, так как внутри контейнера не содержалось больших файлов, необходимых для работы приложения. Ситуация с образом `grafana` оказалась несколько иная: приложение и используемые файлы достаточно объемны, и сокращение памяти оказалось не таким существенным.

Использование данного приложения позволяет оптимизировать путем сокращения размера образа, хранимого в памяти, не только занимаемое дисковое пространство, но и время передачи образа через сеть.

Литература

1. Docker. What is a Container? [Электронный ресурс]: URL: <https://www.docker.com/resources/what-container> (дата обращения: 12.03.21).
2. Wang Kangjin, Yang Yong, Li Ying et al. FID: a faster image distribution system for Docker platform // IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS*W). Tucson, AZ. 2017. P. 191–198.
3. Zhigang Lu, Yuewen Wu, Jiwei Xu, Tao Wang. An acceleration method for Docker image update // IEEE International Conference on Fog Computing (ICFC). Prague, Czech Republic. 2019. P. 15–23.
4. Nannan Zhao, Tarasov V., Hadeel Albahar, Ali Anwar et al. Large-scale analysis of Docker images and performance implications for container storage systems // IEEE Transactions on Parallel and Distributed Systems Submit Manuscript. 2021. Vol. 32. No 4. P. 918–930.

5. Chao Zheng, Rupprecht L., Tarasov V. et al. Wharf: Sharing Docker images in a distributed file system // SoCC '18: Proceedings of the ACM Symposium on Cloud Computing. New York, USA: Association for Computing Machinery. 2018. P. 174–185.
6. CARE [Электронный ресурс]: URL:<https://proot-me.github.io/care/> (дата обращения: 14.03.21).
7. GitHub репозиторий приложения chainsaw [Электронный ресурс]: URL:<https://github.com/LazyDareDevil/chainsaw> (дата обращения: 12.03.21).

УДК 004.891

Павлова Е. Д., Чеботарева А. А., Цепелева Р. В.

Решение задачи автоматизированной разметки подводных животных с помощью каскада нейронных сетей

Рекомендовано к публикации ассистентом Якушкиным О. О.

1. Введение. На сегодняшний день задача подводной сегментации и распознавания морских обитателей методами машинного обучения актуальна для проведения морских исследований в области мониторинга донных сообществ и автоматического картографирования морских ландшафтов. Подсчет подводных популяций необходим для проведения анализа экологического состояния среды.

Для анализа данных по фото- и видеонаблюдениям необходимо иметь размеченные наборы различных данных. Существующие базы данных, описывающие биологическое разнообразие глубоководного морского дна разрозненны и содержат мало структурированной информации. Также проблема создания обширных баз данных заключается в редкости многих донных обитателей, вследствие чего количество изображений определенного вида животных может быть очень мало. Именно поэтому решение задачи сегментации подводных сообществ целесообразно решать с помощью нейронных сетей, решающих класс задач Few Shot Learning [1], особенность которых заключается в том, что с их помощью возможно сегментировать объект, имея всего несколько изображений каждого вида животных и растений. В ходе исследования были подготовлены небольшие наборы данных, содержащих в себе 60 видов морских обитателей, а также проанализированы имеющиеся в свободном доступе датасеты.

В данной работе представлено решение задачи подсчета глубоководных малоподвижных обитателей с помощью нейросетевых технологий.

Павлова Екатерина Денисовна – студент, Санкт-Петербургский государственный университет; e-mail: e.d.pavlova@spbu.ru, тел.: +7(999)224-39-74

Чеботарева Ангелина Андреевна – студент, Санкт-Петербургский государственный университет; e-mail: angel.xe99@gmail.com, тел.: +7(999)224-39-74

Цепелева Рита Вячеславовна – студент, Санкт-Петербургский государственный университет; e-mail: st062153@student.spbu.ru, тел.: +7(999)224-39-74

2. Постановка задачи. Целью работы является построение решения разметки и детектирования малоподвижных подводных обитателей. Для достижения поставленной цели необходимо решить ряд задач, которые в свою очередь можно разбить на два подмножества, представленных на рис. .

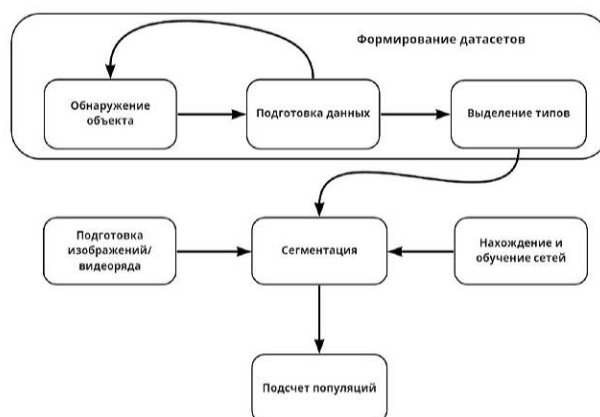


Рис. 1. Поставленные задачи

3. Представленное решение. Предлагаемое решение основано на создании каскада нейронных сетей, каждая из которых способна решать одну из поставленных задач, а именно, улучшение качества изображения, сегментации и подсчета покрытия особями поверхности, возможно достичь поставленной цели, добиваясь наилучшего результата на каждом этапе. Одним из важных критериев выбора компонентов алгоритма было ограничение на объем обучающего набора данных, поэтому в первую очередь внимание было обращено на модели, способные выполнить задачу сегментации без крупномасштабных датасетов, таких как ImageNet и COCO, а выполняющие ту же задачу по нескольким кадрам, т. е. количество обучающих примеров ограничено пятью изображениями. Примером является модель FS-Seg, обученная на базе данных FSS-1000 [2] и описанная в статье [3]. Легкую расширяемость FSS-1000 также можно отнести к положительным характеристикам данного подхода.

С задачей сегментации объектов эффективно [4] справляются сверточные нейронные сети. Для видеопотока обычно такого рода задачи решаются путем раздельной обработки информации о внешнем виде и движении с использованием стандартных 2D сверточных сетей с последующим слиянием двух источников информации. Однако новый подход – DC-Seg [5], превосходящий по скорости и точности накладываемой маски, предлагает осуществлять сегментацию заметных объектов на видео с помощью 3D CNN, которые ранее не были эффективно использованы для решения проблем компьютерного зрения.

4. Подготовка данных. Перед тем как решать задачу сегментации, необходимо создать набор классов разных видов животных и растений. На рис. после обнаружения объекта наступает этап подготовки данных, без которого невозможно перейти к этапу сегментации, так как нейронной сети нужны небольшие наборы данных с целевыми объектами, которые должны быть сегментированы на изображении.

Одна из самых известных открытых баз данных – Fish Recognition Ground-Truth data [6], использованная в проекте Fish4Knowledge для решения задач детектирования и распознавания видов [7], однако она содержит только 15 видов рыб, обитающих около кораллового рифа. Другие же датасеты [8, 9] содержат около 500 видов, но в качестве выделения объектов используются прямоугольники, в связи с чем часто плавники и части тела могут быть обрезаны.

Для решения задачи сегментации различных морских обитателей была создана выборка из шести изображений для каждого вида. Для данного датасета было произведено изменение размера исходного изображения до размера 224×224 пикселя и создана черно-белая маска изображения такого же размера.

Ввиду особенностей видео под водой, мутное изображение – одно из самых часто встречающихся преград для точного наложения маски на объект, поэтому также был опробован метод, когда видеоряд разбивается на кадры, к каждому из которых была применена SISR [10], генеративная super-resolution модель для увеличения четкости кадра.

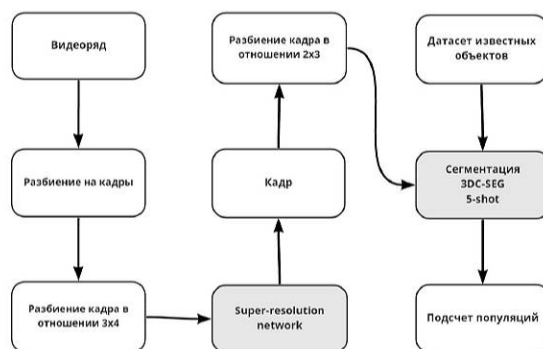


Рис. 2. Этапы решения задач

Полученное решение представлено на рис. . Видеоряд подводной съемки разбивается на кадры, которые в свою очередь проходят предобработку и с помощью нейронной сети увеличивают разрешение изображения. Сегментация объектов из базы данных производится на кадрах, прошедших предобработку. Более подробное описание использованных нейронных сетей представлено в таблице.

Таблица. Рассмотренные нейронные сети

Сеть	Наличие весов	Датасет	Сегментация
3DC-SEG	да	COCO, YouTubeVOS, DAVIS'17	видео сегментация (unsupervised task)
5-shot	да	FSS-1000	FSL сегментация
SIDR	нет	USR-248	Single Image Super-resolution

5. Заключение. С помощью рассмотренной архитектуры можно успешно решать задачу автоматизированной разметки подводных животных. Применение новых алгоритмов улучшения изображений и сегментации подводных объектов для автоматизации мониторинга и подсчета популяций может значительно улучшить качество данных и сократить количество затрачиваемых ресурсов по сравнению с традиционными методами изучения морской флоры и фауны. Полученное решение может быть использовано как основа для создания обширных баз данных для последующего обучения статичных сегментационных моделей.

Литература

1. Dong N., Xing E. Few-shot semantic segmentation with prototype learning // British Machine Vision Conference. 2018. Vol. 3.
2. Azad R. et al. On the texture bias for few-shot CNN segmentation // Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2021. P. 2674–2683.
3. Li X. et al. Fss-1000: A 1000-class dataset for few-shot segmentation // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020. P. 2869–2878.
4. Miao J., Wei Y., Yang Y. Memory aggregation networks for efficient interactive video object segmentation // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020. P. 10366–10375.
5. DC-Seg Making a case for 3D convolutions for object segmentation in videos [Электронный ресурс]: URL: <https://arxiv.org/pdf/2008.11516.pdf> (дата обращения: 24.12.2020).
6. Fish Recognition Ground-Truth data [Электронный ресурс]: URL: <http://groups.inf.ed.ac.uk/f4k/groundtruth/recog> (дата обращения: 24.12.2020).
7. Fish Species Recognition [Электронный ресурс]: URL: <http://www.perceivelab.com/datasets> (дата обращения: 24.12.2020).
8. Ozfish [Электронный ресурс]: URL: <https://aims.github.io/ozfish> (дата обращения: 24.12.2020).
9. Fish Dataset [Электронный ресурс]: URL: <https://wiki.qut.edu.au/display/raq/Fish+Dataset> (дата обращения: 24.12.2020).
10. Islam M. J. et al. Underwater image super-resolution using deep residual multipliers // 2020 IEEE International Conference on Robotics and Automation (ICRA). 2020. P. 900–906.

УДК 004.023

Рущенко В. В.

Разработка алгоритма упорядочивания наложенных контуров

Рекомендовано к публикации доцентом Корховым В. В.

1. Введение. В современном мире нейронные сети имеют обширную область практического применения. Одной из таких областей является промышленный дизайн. На данный момент существует проблема преобразования точечных контуров, сгенерированных нейронными сетями, в векторизованный САД-формат для дальнейшей работы. Для построения контура нужно, как правило, объединить несколько «одинаковых» контуров, отличающихся шумом порядка размера пикселя. Последовательное объединение двух контуров даёт два контурных витка, которые необходимо преобразовать в один. При этом простейшие алгоритмы поиска ближайших соседей могут пропустить часть точек. В данной статье приводится описание алгоритмов, которые основаны на поиске ближайшего соседа и способны решить проблему пропуска точек.

2. Постановка задачи и используемые данные. Целью данной работы является создание программного продукта, решающего задачу построения одинарного контура по заданному двойному. Входные данные представлены в виде набора $n \in \{159, 328\}$ точек, имеющих координаты (x_i, y_i) , $i = 1, \dots, n$. На выходе программа будет выдавать упорядоченный набор из всех входных точек, который будет представлять из себя одинарный контур без самопересечений.

3. Алгоритмы. В данном разделе представлены алгоритмы построения контура по заданным точкам.

3.1. Метод ближайшего соседа. Алгоритм ближайшего соседа [1] – один из первых алгоритмов, использованный для приближённого решения задачи коммивояжёра [2]. Данный метод состоит из следующих шагов:

1. Создать пустой список.

Рущенко Влада Владиславовна – студент, Санкт-Петербургский государственный университет; e-mail: st064148@student.spbu.ru, тел.: +7(916)316-15-62

2. Добавить в него первый элемент входных данных.
3. Рассмотреть последний элемент из списка.
4. Найти не входящий в список элемент, расстояние рассматриваемого элемента до которого будет минимально (здесь и в остальных алгоритмах, описанных в данной статье, для измерения расстояния используется евклидова метрика).
5. Добавить элемент в список.
6. Повторять пункты 3–5, пока в списке не будут присутствовать все входные точки.

В результате работы алгоритм выдаёт список – упорядоченный набор входных данных, представляющий из себя короткий маршрут по заданным точкам. Алгоритм прост в реализации, быстро выполняется, но может выдавать неоптимальные решения. Поэтому его использование в рамках рассматриваемой задачи не гарантирует, что в результате построенный им маршрут будет одинарным контуром.

3.2. Улучшенный метод ближайших соседей с пропуском точек. Чтобы получить одинарный контур, надо остановить алгоритм, когда маршрут будет замыкаться. В итоге улучшенный метод будет отличаться только критерием остановки, т. е. в список будут добавляться элементы, пока ближайший элемент не из списка расположен ближе к рассматриваемому элементу, чем первый элемент списка.

В результате работы алгоритма получается контур, построенный не по всем входным точкам. Изменение начальных точек не решает данную проблему, поэтому надо рассмотреть другие варианты удаления пропусков.

3.3. Улучшенный метод ближайших соседей без пропуска точек. Для включения пропущенных точек в контур можно произвести следующие действия:

1. Взять элемент, не вошедший в список.
2. Найти для него два элемента, сумма расстояний до которых от него минимальна.
3. Вставить элемент между найденными в пункте 2 элементами.
4. Повторять пункты 1–3, пока все элементы не войдут в список.

При выполнении этих шагов после работы алгоритма, описанного

в пункте 3.1, будет получен алгоритм, результатом выполнения которого будет одинарный контур без пропусков точек.

3.4. Добавление точек из второго контура в первый.

Рассмотрим ещё один способ построения контура на основании поиска ближайших соседей. Поскольку на вход подаются частично упорядоченные данные, можно разделить их на два контура и точки из второго контура последовательно добавлять в первый. Для получения контура надо выполнить следующие действия:

1. Найти элемент, после которого начинается второй виток.
2. Создать список, куда будут входить все элементы до элемента, найденного в пункте 1.
3. Рассмотреть элемент не из списка.
4. Найти два идущих подряд элемента из списка, сумма расстояния которых до рассматриваемого элемента минимальна.
5. Вставить элемент в список между элементами, найденными в пункте 4.
6. Повторять пункты 3–5, пока в список не войдут все исходные данные.

3.5. Удаление самопересечений. Как упоминалось ранее, метод ближайшего соседа далеко не всегда строит минимальный маршрут, т. е. при использовании его и алгоритмов, основанных на нём, есть вероятность того, что полученный маршрут будет иметь петли. Для того чтобы алгоритм гарантировал отсутствие самопересечений в маршруте, добавим в него метод, используемый в алгоритме 2-opt [3]. Его можно сформулировать следующим образом:

1. Найти самопересечение (начало петли в точке i , конец в точке k).
2. Создать пустой список.
3. Добавить в него все элементы до i -го включительно.
4. Добавить в список элементы от $(k-1)$ -го до $(i+1)$ -го в обратном порядке.
5. Добавить в список оставшиеся элементы.

При добавлении данного метода в конец алгоритмов, описанных в пунктах 3.3 и 3.4, будут получены новые алгоритмы, выдающие в

результате контур без пропусков и самопересечений.

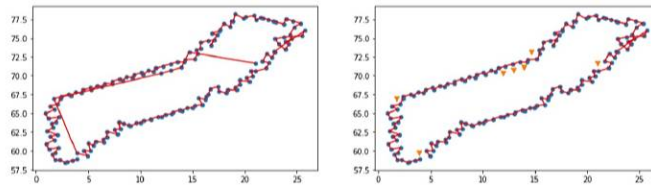


Рис. 1. Результаты применения метода ближайших соседей – слева и улучшенного метода ближайших соседей с пропуском точек – справа

4. Результаты применения алгоритмов. Все описанные выше алгоритмы были применены на входных данных, представлявших из себя двойной контур из 159 точек на плоскости. На рис. 1 показаны результаты выполнения алгоритмов из пунктов 3.1 и 3.2 соответственно. Точками обозначены исходные данные, треугольниками – пропущенные в контуре данные, линиями – построенные контуры. Как и следовало ожидать, оба метода не справились с поставленной задачей: метод ближайшего соседа построил двойной контур, а не одинарный, улучшенный метод ближайших соседей из пункта 3.2 пропустил часть исходных точек.

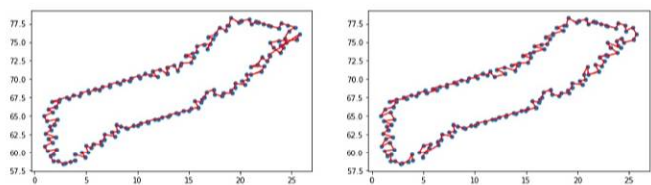


Рис. 2. Результаты применения улучшенного метода ближайших соседей без пропуска точек – слева и метода, основанного на включении точек из второго контура в первый, – справа

На рис. 2 изображены результаты работы алгоритмов, описанных

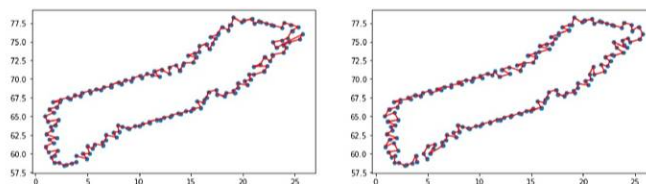


Рис. 3. Результаты применения улучшенного метода ближайших соседей без пропуска точек и петель – слева и улучшенного метода, основанного на включении точек из второго контура в первый и удалении петель, – справа

в пунктах 3.3 и 3.4. Можно заметить, что данные алгоритмы построили маршруты по всем исходным точкам без пропусков, однако полученные маршруты имели петли. Значит, данные алгоритмы не выполнили поставленную задачу.

Новые алгоритмы, полученные при добавлении к алгоритмам из пунктов 3.3 и 3.4 метода, описанного в пункте 3.5, построили контуры без самопересечений и пропуска точек (см. рис. 3).

Таким образом, на основе поиска ближайшего соседа были разработаны два алгоритма, решающих поставленную задачу.

5. Заключение. Проблема, рассматриваемая в данной работе, весьма актуальна. Простые алгоритмы, основанные на поиске ближайшего соседа, способны построить одинарный контур по заданному двойному. В дальнейшем предстоит найти другие пути решения поставленной задачи.

Литература

1. Gutin G., Yeo A., Zverovich A. Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP // *Discrete Applied Mathematics*. 2002. No 117. P. 81–86.
2. Левитин А. В. Алгоритмы. Введение в разработку и анализ. М.: Вильямс, 2006. 576 с.
3. Helsgaun K. General k-opt submoves for the Lin-Kernighan TSP heuristic // *Mathematical Programming Computation*. 2009. No 1. P. 119–163.

УДК 004.054

Савченко А. В.

Сравнение системы прав доступа на основе промежуточных групп пользователей со стандартной системой прав доступа без промежуточных групп

Рекомендовано к публикации доцентом Ганкевичем И. Г.

Введение. В Linux- и UNIX-подобных системах права доступа к файлам и ресурсам операционной системы определяются номером пользователя и номером группы процесса, который пытается получить доступ к ресурсу или файлу. Однако дихотомии пользователей и групп недостаточно для описания сложных взаимосвязей между сотрудниками организации и правами доступа к ресурсам, которые у них есть [1]. Члены одной команды обладают одинаковыми правами с исключениями для конкретных работников (например, доступ к базе данных есть только у ответственного за администрирование, и т. п.). Также, ресурсы связаны между собой отношением агрегации: доступ к вычислительному кластеру означает доступ к каждому узлу этого кластера, а доступ к администрированию базы данных означает доступ к каждой программе, которая необходима для выполнения данной работы. В связи с этим более эффективной схемой организации доступа выступает дихотомия групп пользователей ресурсов и групп самих ресурсов. Группы пользователей агрегируют сотрудников из одной команды, одного подразделения, одного здания, города, страны и т. д., а также другие группы пользователей. Группы ресурсов агрегируют узлы кластера, программы и веб-сервисы, необходимые для выполнения сотрудником определенного вида работы и т. п., а также другие группы ресурсов. Между группами пользователей и группами ресурсов устанавливаются связи, каждая из которых означает получение прав доступа к ресурсам связанной группой пользователей.

Савченко Александр Владимирович – студент, Санкт-Петербургский государственный университет; e-mail: st062158@student.spbu.ru, тел.: +7(812)428-47-83

Рассматриваются системы прав доступа для UNIX-систем. Первая – это система, которая используется по умолчанию в UNIX-системах, вторая – основывается на дихотомии групп пользователей, ресурсов и групп самих ресурсов. В работе сравниваются эти системы по структуре и быстродействию. Для сравнения систем анализируются структуры, в которой они хранят информацию о зависимостях групп пользователей и групп ресурсов.

Анализ структуры. Изначально имеется база данных с промежуточными группами пользователей и ресурсов. В работе рассмотрена только одна таблица из базы данных. Структура представлена в таблице 1.

Таблица 1. Структура базы данных

child_id	parent_id	type
4777	4732	0
...
4732	4225	1
...
4225	4524	2

В строках указываются идентификаторы. Тип связи обозначается в последнем столбце. Тип 0 – связь «Пользователь – Пользователь». Тип 1 – связь «Группа – Пользователь». Тип 2 – связь «Группа – Группа».

Для проведения сравнения составим новую базу данных, в которой не будет промежуточных групп, т. е. всё наследование будет напрямую от ресурсов к пользователю. Структура рассматриваемой таблицы останется той же, кроме типов связей. Так как тип связи только один (Тип 1), столбец «type» можно опустить.

Построив граф зависимостей элементов обеих таблиц баз данных, заметим, что система с промежуточными группами проще в восприятии [2]. Сравнение приведено в таблице 2.

Таблица 2. Сравнение графов

	С промежуточными группами	Без промежуточных групп
Число вершин	110	76
Число граней	206	533

Быстродействие. Так как структура представляет собой базу данных важно проверить, насколько быстро будут выполняться запросы к ней. Для этого были составлены два запроса: 1) «вывести

всех пользователей одной группы» и 2) «вывести все группы одного пользователя». На каждой базе данных запросы выполнялись по 100, 1000 и 10 000 раз. В таблице 3 приведено суммарное время выполнения каждого запроса в секундах.

Таблица 3. Сравнение быстродействия запросов

Число выполнений	С промежуточными группами		Без промежуточных групп	
	Запрос 1	Запрос 2	Запрос 1	Запрос 2
100	4,56	4,68	3,98	3,92
1000	45,76	47,11	34,42	32,11
10 000	467,45	490,92	354,42	318,89

Выводы. Рассмотрев два варианта системы прав доступа: основанную на промежуточных группах и без них. Было выявлено, что несмотря на незначительное увеличение времени на обработку запросов к первой системе, видится, что более простая структура хранения зависимостей является более перспективной при увеличении числа пользователей и различных ресурсов.

Литература

1. Grunbacher A. POSIX Access Control Lists on Linux // Proceedings of the FREENIX Track: 2003 USENIX Annual Technical Conference. 2003. P. 259–272.
2. Просолупов Е. В. Курс лекций по дискретной математике. Ч. 3. Теория алгоритмов и теория графов. СПб.: Изд-во СПбГУ, 2014. 85 с.

УДК 004.93

Сермягин Н. В.

Детектирование людей в видеопотоке с применением детектора движения и сверточной нейронной сети

Рекомендовано к публикации доцентом Гришкиным В. М.

1. Введение. Одной из самых распространенных задач в компьютерном зрении является детектирование людей на изображении или в видеопотоке. Целью детектирования является отслеживание присутствия искомого объекта и определение его местоположения, в случае если присутствие было зафиксировано. Алгоритмы, решающие данную задачу, могут быть применены в самых разных областях и, в частности, широко используются в системах видеонаблюдения. Основными сложностями в данной задаче являются большая разнообразность внешности людей и влияние на них окружения (перекрытие другими объектами, плохое освещение, плохая видимость из-за погоды и др.). Также с ростом количества видеокамер на вычислительную сложность алгоритма накладываются ограничения.

2. Постановка задачи. Необходимо решить задачу обнаружения и выделения людей на кадре из видеопотока – пометить обнаруженных людей обрамляющим окном. Видео берется с камеры, которая является статичной и ведет съемку круглосуточно.

Из существующих методов широко используются алгоритмы, основанные на гистограмме направленных градиентов (Histogram of Oriented Gradients, HOG) и методе опорных векторов (Support Vector Machine, SVM) [1]. Но такой подход плохо справляется со сложными ситуациями. Хорошие результаты в детектировании объектов на изображении показывают алгоритмы, основанные на нейронных сетях, например YOLO или R-CNN [2, 3]. Но для их применения требуется большая вычислительная мощность.

В данной работе предложен алгоритм, использующий для детектирования людей комбинацию из детектора движения и сверточной нейронной сети, состоящий из двух этапов:

Сермягин Никита Владимирович – студент, Санкт-Петербургский государственный университет; e-mail: nikitasermiagin@gmail.com, тел.: +7(919)867-61-44

1. Выделение из кадра с помощью детектора движения областей интереса.
2. Классификация полученных областей с помощью нейронной сети.

3. Детектор движения. Для детектирования движения был выбран алгоритм адаптивного вычитания фона (Adaptive Background Subtraction) [4], так как он не требует большого числа вычислений и хорошо справляется с изменяющимся со временем фоном.

Идея алгоритма заключается в получении фонового изображения путем вычисления среднего значения N последних кадров для последующего вычитания из него текущего кадра и сравнения с некоторым пороговым значением. Но, чтобы не хранить в памяти предыдущие кадры и ускорить вычисления, фон считают рекурсивно по формуле

$$B_{i+1}(x, y) = (1 - \alpha)B_i(x, y) + \alpha I_i(x, y),$$

где α – параметр, определяющий, насколько быстро новая информация заменит старую, а $B_i(x, y)$ и $I_i(x, y)$ – значения пикселей (x, y) i -го фона и кадра соответственно.

Дополнительно, чтобы избавиться от различного шума в видеопотоке, к каждому кадру было применено размытие по Гауссу.

4. Архитектура нейронной сети. Для классификации объектов используется сверточная нейронная сеть, состоящая из трех сверточных и трех полносвязных слоев. В сверточных и полносвязных слоях использовалась функция активации ReLU, а к выходному тензору была применена сигмоидная функция активации. На вход сеть принимает изображение в формате RGB и выдает число в диапазоне от 0 до 1 – вероятность того, что на изображении человек. Такую архитектуру было решено использовать, чтобы минимизировать количество параметров нейронной сети. Более подробно архитектура представлена на рис. 1.

5. Реализация. Алгоритм реализован на языке программирования Python. Для детектора движения использовалась библиотека алгоритмов компьютерного зрения OpenCV и библиотека для работы с многомерными матрицами и массивами NumPy, а для получения областей интереса – метод findContours из OpenCV. Для нейронной

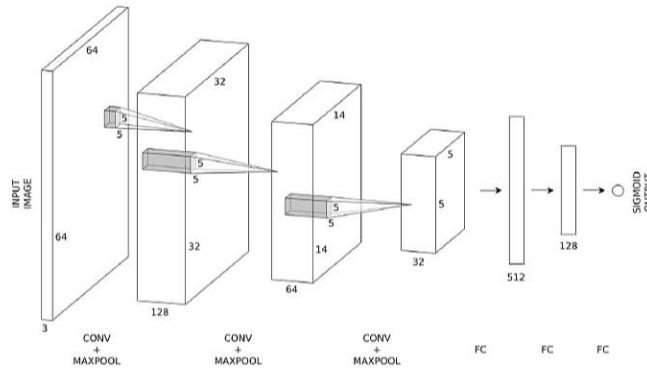


Рис. 1. Архитектура нейронной сети

сети использовалась библиотека PyTorch.

5.1. Подготовка данных. Для обучения и проверки нейронной сети был собран датасет, состоящий из нескольких публичных. Из датасета INRIA Person Dataset [5] получены изображения, содержащие людей, полностью или частично, и изображения фона. А из датасетов Vehicle Image Database [6] и CIFAR-100 [7] – изображения, содержащие транспортные средства или различных животных. Итоговый размер датасета составляет 8700 изображений, половина из которых содержит людей.

5.2. Обучение нейронной сети. Нейронная сеть обучалась методом обратного распространения ошибки с применением оптимизационного алгоритма Adam и функции ошибки Binary Cross Entropy Loss. После 100 эпох обучения точность составила 97,27% по метрике Accuracy.

5.3. Результаты. Реализованный алгоритм работает со средней частотой 44,5 кадра в секунду на машине со следующими характеристиками: Ryzen 2700, 8 ГБ оперативной памяти, NVIDIA RTX 2070 с 8 ГБ видеопамати. Алгоритм справляется с детектированием людей, в частности, если они частично перекрыты или малы относительно всего изображения. Однако иногда, особенно если в видеопотоке большое число движущихся объектов, можно наблюдать краткосрочные ложные срабатывания. Пример работы алгоритма представлен на рис. 2.



Рис. 2. Пример работы алгоритма

6. Заключение. В работе предложен алгоритм для детектирования людей в видеопотоке с использованием детектора движения и сверточной нейронной сети. Приведены результаты, которые показывают, что алгоритм справляется с поставленной задачей, но требует доработки для минимизации числа ложных срабатываний. В качестве решения планируется применить различные архитектуры нейронной сети или добавить в алгоритм этап с отслеживанием найденных объектов, для фильтрации по времени нахождения объекта в видеопотоке.

Литература

1. Said Y., Atri M., Tourki R. Human detection based on integral histograms of oriented gradients and SVM // International Conference on Communications, Computing and Control Applications (CCCA). IEEE Trans. Circuits Syst. 2011. P. 1–5.
2. Redmon J., Divvala S., Girshick R. et al. You only look once: unified, real-time object detection // IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Trans. Circuits Syst. 2016. P. 779–788.
3. Girshick R., Donahue J., Darrell T. et al. Rich feature hierarchies for accurate object detection and semantic segmentation // IEEE Conference on Computer Vision and Pattern Recognition. IEEE Trans. Circuits Syst. 2014. P. 580–587.

4. Xiao-jun Tan, Jun Li, Chunlu Liu. A video-based real-time vehicle detection method by classified background learning // World Trans. Eng. Technol. Edu. 2007. Vol. 6. No 1. P. 189–192.
5. INRIA Person Dataset [Электронный ресурс]: URL:<http://pascal.inrialpes.fr/data/human/> (дата обращения: 19.03.21).
6. Vehicle Image Database [Электронный ресурс]: URL:https://www.gti.ssr.upm.es/data/Vehicle_database.html (дата обращения: 19.03.21).
7. The CIFAR-100 Dataset [Электронный ресурс]: URL:https://git-disl.github.io/GTDLBench/datasets/cifar-100_datasets/ (дата обращения: 19.03.21).

УДК 004.023

Сорокина С. О., Михайлова Е. Г.

Оптимизация гиперпараметров алгоритмов градиентного бустинга

1. Введение. На сегодняшний день алгоритмы градиентного бустинга являются достаточно распространенными методами решения задач машинного обучения [1]. Эти алгоритмы весьма распространены, в том числе и благодаря высоким результатам в соревнованиях на платформе kaggle.com [2]. Задача оптимизации гиперпараметров алгоритмов требует перебора большого количества возможных комбинаций значений гиперпараметров с последующей оценкой моделей, построение которых является вычислительно трудоемким [3, 4]. В связи с тем, что оптимизация гиперпараметров требует много вычислительных ресурсов, задача поиска новых методов оптимизации является важным направлением современных исследований. Так как значения некоторых гиперпараметров могут быть взаимосвязаны, то обычно принято оптимизировать гиперпараметры совместно. В рамках этой статьи решена задача упрощения оптимизации гиперпараметров алгоритмов градиентного бустинга за счет разбиения множества гиперпараметров на группы.

2. Проведенные исследования. Метод заключается в следующем: вместо того, чтобы оптимизировать гиперпараметры совместно, они разбиваются на группы и оптимизация проводится последовательно в рамках каждой группы. Для того чтобы понять, является ли этот метод состоятельным, было проведено сравнение его производительности с популярными на сегодняшний день методами оптимизации гиперпараметров алгоритмов. Затем был проведен эксперимент для исследования влияния ранней остановки на оценку качества и количество итераций поиска для разных методов оптимизации.

Сорокина Светлана Олеговна – студент, Санкт-Петербургский государственный университет; e-mail: st049892@student.spbu.ru, тел.: +7(921)570-83-47

Михайлова Елена Георгиевна – доцент, Санкт-Петербургский государственный университет, Университет ИТМО; e-mail: e.mikhaylova@spbu.ru, тел.: +7(921)997-97-91

2.1. Первый эксперимент: состоятельность метода. Для выявления состоятельности метода, суть которого заключается в разбиении гиперпараметров на группы, в рамках первого эксперимента проведено сравнение методов оптимизации с разбиением на группы и без разбиения.

В рамках эксперимента рассматриваются следующие гиперпараметры алгоритмов градиентного бустинга, которые разделены на группы по направленности на основании статьи [5]:

1. Основные гиперпараметры деревьев: *max_depth* и *min_child_weight*.
2. Гиперпараметры регуляризации: *alpha* и *lambda*.
3. Гиперпараметры, отвечающие за подвыборки при построении деревьев: *subsample* и *colsample_bytree*.

Эксперименты проводились на наборе данных Facebook Comment Volume Dataset, который содержит информацию о постах социальной сети Facebook. На основании данных проводится предсказание количества комментариев для конкретного поста. Это является задачей регрессии (предсказание непрерывных значений на основании исторических данных). Поэтому для оценки качества использовалась метрика R -квадрат, рассчитанная на трех фолдах кросс-валидации и отражающая, насколько итоговая модель работает лучше, чем модель, равная константному значению. Если значение метрики равно нулю, то модель не лучше константы, если значение метрики близко к единице, то модель почти идеальна. В рамках решения этой задачи рассмотрен алгоритм XGBoost, который реализует градиентный бустинг.

Метод сравнивается с самыми распространенными на сегодняшний день методами оптимизации [6]: поиск по сетке и случайный поиск. Для задания сетки поиска каждому гиперпараметру ставились в соответствие три возможных значения, поэтому всего было выполнено 729 итераций поиска. Для метода с разбиением на группы было задано 243 итерации поиска в каждой группе. Для методов, использующих случайный поиск, было произведено 10 запусков для построения доверительных интервалов оценки. Эксперимент проводился в среде Google Colaboratory с использованием ускорителя Tesla K80.

Для проведения анализа полученных результатов рассмотрены

графики улучшения достигнутой оценки качества при увеличении количества итераций поиска (рис. 1). На рис. 1 изображены графики

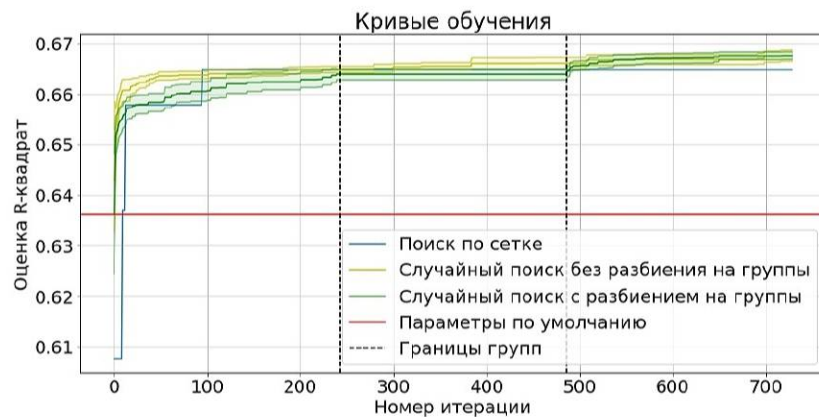


Рис. 1. Графики оценки R -квадрат разных методов оптимизации гиперпараметров XGBoost на данных Facebook Comment Volume Dataset

оценок R -квадрат для методов поиска, рассматриваемых в рамках эксперимента. На горизонтальной оси отмечаются номера итераций поиска, по вертикальной оси — наилучшая оценка качества, достигнутая на текущей итерации. Для методов, использующих случайный поиск, изображены 95% доверительные интервалы для среднего значения в предположении, что данные распределены нормально.

Из рис. 1 видно, что случайный поиск с разбиением на группы в итоге работает не хуже, чем случайный поиск без разбиения на группы: наблюдается пересечение доверительных интервалов на последних итерациях поиска. Стоит отметить, что для случайного поиска с разбиением на группы во второй группе не происходит улучшения оценки качества на рассматриваемом наборе данных. Поэтому введение раннего критерия остановки в данном случае может помочь значительно сократить число итераций поиска.

Помимо этого, в рамках эксперимента были рассмотрены связи между значениями гиперпараметров и итоговой оценкой качества. Внутри первой и третьей группы наблюдается взаимосвязь между

значениями гиперпараметров и итоговой оценкой. Это означает, что эти группы были выделены верно: при разделении пар гиперпараметров на отдельные группы алгоритм может не сойтись к оптимальным значениям.

2.2. Второй эксперимент: влияние критерия останова.

Для оценки влияния раннего критерия останова на итоговую оценку качества и количество итераций поиска был проведен еще один эксперимент, который выполнялся при тех же условиях и настройках, что и первый. В рамках эксперимента рассмотрены критерии останова 50, 25 и 10 итераций без улучшения максимально достигнутой оценки качества. Для метода, использующего разбиение на группы, было введено следующее перераспределение числа итераций между группами:

1. Для первой группы зафиксировано максимальное возможное количество итераций, равное 243.
2. Пусть I_1 – номер итерации на которой был завершен поиск в рамках первой группы, где $I_1 \leq 243$. Тогда для второй группы выделяется $\frac{729-I_1}{2}$ итераций поиска, т. е. половина оставшегося лимита итераций.
3. Предположим, что при поиске во второй группе было выполнено I_2 итераций, где $I_2 \leq \frac{729-I_1}{2}$. Тогда для поиска в третьей группе выделяется $729 - (I_1 + I_2)$ итераций.

После выполнения второго эксперимента получены результаты оптимизации при разных критериях останова, представленные в таблице.

Введены обозначения: RS – random search, случайный поиск без разбиения на группы; GRS – grouped random search, случайный поиск с разбиением на группы; Grid – grid search, поиск по сетке. Для оценок качества указаны среднее значение и стандартное отклонение для 10 запусков с округлением до четвертого знака после запятой, разделенные знаком «±». Для числа итераций округление производится до целых, а также указано максимальное число итераций, сделанных для каждого критерия останова.

Из таблицы видно, что метод с разбиением на группы для любого критерия останова сходится к примерно одинаковым оценкам качества. Введение критерия останова увеличивает среднюю оценку качества при поиске с разбиением на группы. Это достигается за счет перераспределения итераций поиска между группами.

Таблица. Значения оценки качества и числа итераций при проведении оптимизации с разными критериями остановки

Критерий остановки	Метод поиска	Оценка R -квадрат	Число итераций
–	RS	$0,6676 \pm 0,0018$	729
–	GRS	$0,6676 \pm 0,0012$	729
–	Grid	0,6648	729
50	RS	$0,6636 \pm 0,0012$	76 ± 23 max : 113
50	GRS	$0,6683 \pm 0,0014$	448 ± 103 max : 729
25	RS	$0,6629 \pm 0,0015$	43 ± 19 max : 88
25	GRS	$0,6688 \pm 0,0014$	517 ± 119 max : 729
10	RS	$0,6618 \pm 0,0025$	18 ± 7 max : 37
10	GRS	$0,6683 \pm 0,0012$	522 ± 124 max : 685

Для случайного поиска без разбиения на группы число итераций очень быстро сокращается, но при этом наблюдается падение оценки качества: она становится хуже, чем при поиске по сетке. Отличие оценок качества между методами оптимизации в этом эксперименте видно только в третьем знаке после запятой.

Можно сказать, что метод с разбиением на группы весьма устойчив при введении критерия ранней остановки: даже при остановке после 10 итераций без улучшения максимально достигнутой оценки качества метод сходится к значениям оценки качества не хуже, чем в эксперименте без критерия остановки.

2.3. Другие эксперименты. Дополнительно были проведены аналогичные эксперименты для байесовской оптимизации. Все выводы, полученные для случайного поиска с разбиением на группы и без, справедливы и для байесовской оптимизации, отличия наблюдались лишь в значениях полученных оценок качества и количестве итераций до выполнения критериев остановки.

3. Заключение. Эксперименты показывают, что в некоторых случаях методы с разбиением на группы показывают результаты не хуже, чем методы без разбиения. Также методы поиска, использующие разбиение на группы, более устойчивы при использовании критериев остановки. Введение критериев остановки позволяет сократить число итераций поиска без значительных изменений в итоговой

оценке качества в случае методов с разбиением на группы. В данный момент проводятся эксперименты и на других наборах данных. Также выполняется отладка кода распределенной реализации этого метода с помощью технологий Scala и Spark. С программным кодом экспериментов можно ознакомиться в [7].

Литература

1. Anghel A., Papandreou N., Parnell T. et al. Benchmarking and optimization of gradient boosting decision tree algorithms [Электронный ресурс]: URL:<https://arxiv.org/abs/1809.04559> (дата обращения: 18.03.2021).
2. Awesome XGBoost [Электронный ресурс]: URL:<https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions> (дата обращения: 16.03.2021).
3. Wu J., Chen S, Chen X. RPR-BP: a deep reinforcement learning method for automatic hyperparameter optimization // 2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019. P. 1–8.
4. Sommer J., Sarigiannis D., Parnell T. Learning to Tune XGBoost with XGBoost [Электронный ресурс]: URL:<https://arxiv.org/abs/1909.07218> (дата обращения: 18.03.2021).
5. Complete Guide to Parameter Tuning in XGBoost with codes in Python [Электронный ресурс]: URL:<https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/> (дата обращения: 18.03.2021).
6. Szwabe A. Kernel and acquisition function setup for bayesian optimization of gradient boosting hyperparameters // Intelligent Information and Database Systems. Cham: Springer International Publishing, 2018. P. 297–306.
7. Репозиторий исследования [Электронный ресурс]: URL:<https://github.com/RisaMagpie/GradBoostOptim> (дата обращения: 14.03.2021).

УДК 004.75

Терещенко Д. В.

Программный интерфейс для отказоустойчивых распределённых вычислений на Python

Рекомендовано к публикации доцентом Ганкевичем И. Г.

Введение. Сегодня всё чаще встречаются задачи, когда вычислительных ресурсов одного компьютера не хватает. На помощь приходят распределённые вычисления, требующие, в свою очередь, использование специализированного программного интерфейса. Выход из строя одного из узлов в распределённых системах является нормальной ситуацией, поэтому данный интерфейс в таком случае должен предоставлять механизмы для сохранения работоспособности системы в целом.

Экосистема языка программирования Python включает в себя множество инструментов для обработки и анализа данных, однако подобного рода решений не так много [1], а те, что есть, либо слишком узкоспециализированные, либо не дают полной отказоустойчивости. К примеру, одно из популярных решений Ray [2], как пишут сами разработчики, уязвимо к отказу главного узла.

В полной мере эта проблема решена в новом C++ фреймворке Subordination [3], причём без использования контрольных точек, дампов памяти и других интенсивных операций ввода-вывода, которые часто применяются в таких инструментах. Таким образом, возникла необходимость в реализации интерфейса к этому фреймворку, что позволило бы использовать все его преимущества на языке более высокого уровня Python.

Методы. Для достижения поставленной цели необходима низкоуровневая интеграция с исходным фреймворком, что может дать только разработка расширения интерпретатора Python. Для лучшей совместимости с C++ была использована де-факто эталонная реализация языка Python – cPython [4]. Она, в свою очередь, поставляется вместе с Python/C API [5] – инструментом, с помощью которого

Терещенко Дмитрий Владиславович – студент, Санкт-Петербургский государственный университет; e-mail: st064145@student.spbu.ru, тел.: +7(812)428-47-83

Работа выполнена при финансовой поддержке гранта Президента Российской Федерации для молодых кандидатов наук №МК-383.2020.9

разработаны все примитивы и методы, необходимые для функционирования всего фреймворка.

Получившийся интерфейс, как и исходная система Subordination, основан на так называемых управляющих объектах (*kernel*) – объектах в языке программирования, содержащих данные, которые необходимо обработать, и код для их обработки (см. рис. 1).

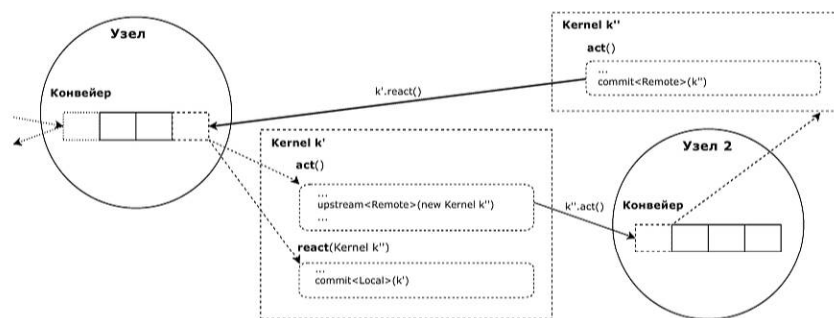


Рис. 1. Упрощенная схема выполнения

В методе *act* некоторая задача либо последовательно вычисляется, либо разбивается на подзадачи (представленные другим набором управляющих объектов). В методе *react* подчиненные объекты, завершившие работу, обрабатываются их родителем. Процедура взаимодействия объектов представляет собой аналог стека вызова функций, но для распределенных систем. Для того чтобы содержащийся в объектах код был исполнен, они отправляются на *конвейер* – очередь из управляющих объектов.

С целью замера производительности и проверки всех сценариев отказоустойчивости была реализована задача пакетной обработки по восстановлению частотно-направленного спектра морского волнения и вычислению дисперсии (область под графиком спектра).

Данные задачи представляют собой следующее. Каждая станция хранит записи частотно-направленного спектра морского волнения, закодированных в пяти переменных. Каждая запись содержит дату измерения и дискретные значения спектра для каждой из частот. Каждый файл содержит записи только одной из станций за промежуток времени, равный одному году.

Данные из каждой пятерки файлов с помощью отдельного управ-

ляющего объекта объединялись в один кортеж и соединялись по дате измерения. При этом обрабатывались все пропуски и дублирования. Затем в дочерних объектах вычислялась дисперсия по формуле [6]. После расчёта всех дисперсий работа программы завершалась.

Результаты. Сначала проводилось тестирование на производительность без сбоев, как для C++, так и для Python реализации. Количество задействованных узлов в кластере изменялось с 1 до 6. Тест повторялся три раза, затем полученные результаты усреднялись. На вход подавались данные из NDBC [7] за год и за 10 лет (см. таблицу). Результат представлен на рис. 2.

Таблица. Свойства набора данных NDBC

Свойства	За год	За 10 лет
размер данных	43МБ	512МБ
размер распакованных данных	238МБ	2851МБ
количество станций	25	25
общее количество спектров	118 491	1 637 809

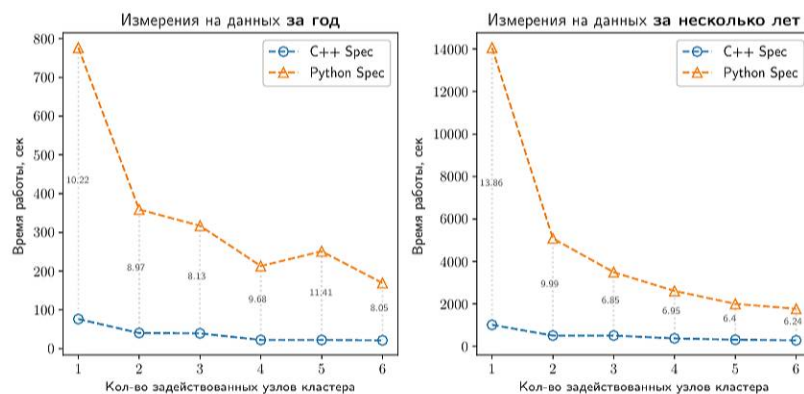


Рис. 2. Замер производительности C++ и Python реализации без сбоев

Далее с целью повторить сценарии из статьи [3] для Python реализации на данных за год таким же образом были произведены замеры производительности с симулированием сбоя узла, на котором находится главный управляющий объект, его копия или подчинённый объект, а также выход из строя всех узлов в сравнении с выполнением программы без сбоев (см. рис. 3).

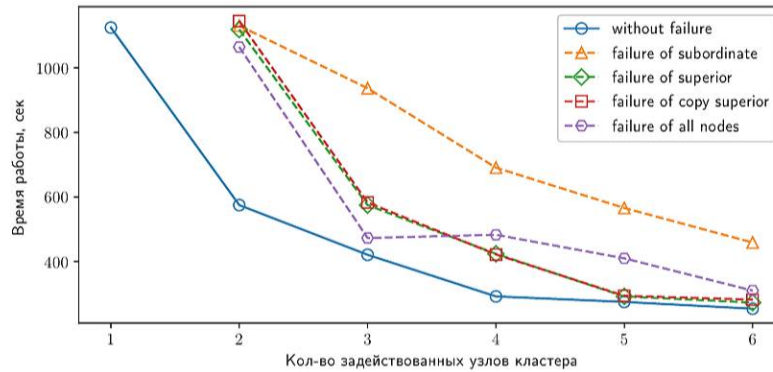


Рис. 3. Замер производительности Python реализации со сбоями

Обсуждение. С увеличением количества задействованных узлов наблюдается прирост производительности, при этом динамика у обеих реализаций совпадает. Всё это показывает корректность работы интерфейса.

Замедление Python реализации было ожидаемым. Во-первых, это связано с тем, что в cPython имеется GIL (Global Interpreter Lock) [8], позволяющий ему работать только в однопоточном режиме. Во-вторых, при переносе управляющих объектов с узла на узел для сериализации/десериализации используется библиотека Pickle [9], внутри которой задействуются не быстрые операции интроспекции. Кроме того, сами вычисления Python не так оптимизированы, как их аналоги в C++. С ростом количества данных и задействованных узлов коэффициент замедления в среднем уменьшается с 12 до 6.

Случай со сбоями узлов с главным управляющим объектом или его копией показал, что это приводит к снижению производительности до производительности кластера без отказавшего узла. Сбой узлов с подчинёнными объектами даёт тот же эффект, но с большими накладными расходами из-за их количества. При выходе из строя всех узлов добавляется некоторое время восстановления.

Заключение. Разработан высокоуровневый интерфейс на языке Python для нового фреймворка отказоустойчивых распределённых вычислений Subordination. Результаты проведения тестирова-

ния показали корректность работы интерфейса, прирост производительности при увеличении количества узлов в кластере, а также возможность обработки различных сценариев сбоя узлов за приемлемое время. В дальнейшем планируется реализовать возможность использования полученного интерфейса для построения распределённых веб-сервисов.

Литература

1. Parallel Processing and Multiprocessing in Python [Электронный ресурс]: URL:<https://wiki.python.org/moin/ParallelProcessing> (дата обращения: 20.11.2020).
2. A Gentle Introduction to Ray [Электронный ресурс]: URL:<https://docs.ray.io/en/master/ray-overview/index.html> (дата обращения: 25.01.2021).
3. Gankevich I., Tipikin Y., Gaiduchok V. Subordination: Cluster management without distributed consensus // In International Conference on High Performance Computing Simulation (HPCS). 2015. P. 639–642.
4. cPython [Электронный ресурс]: URL:<https://github.com/python/cpython> (дата обращения 17.10.2020).
5. Python/C API Reference Manual [Электронный ресурс]: URL:<https://docs.python.org/3/c-api/index.html> (дата обращения: 05.10.2020)
6. Earle M. D. Nondirectional and Directional Wave Data Analysis Procedures. Louisiana: Neptune Sciences, Inc, 1996. 37 p.
7. National Data Buoy Center [Электронный ресурс]: URL:<http://www.ndbc.noaa.gov/dwa.shtml> (дата обращения: 07.12.2020).
8. Python Global Interpreter Lock [Электронный ресурс]: URL:<https://tproger.ru/translations/global-interpreter-lock-guide/> (дата обращения: 12.01.2021).
9. Pickle – Python object serialization [Электронный ресурс]: URL:<https://docs.python.org/3/library/pickle.html> (дата обращения: 11.10.2020).

УДК 004.75

Фараджов А. М.

Сравнительный анализ DLT решений на базе алгоритма PBFT

Рекомендовано к публикации профессором Богдановым А. В.

1. Введение. На данный момент существует множество алгоритмов консенсуса. Многие из них предполагают дополнительные издержки для участников (Proof-of-Work, Proof-of-Stake, Proof-of-Elapsed-Time и др.). Однако существует ряд алгоритмов, не накладывающих подобных ограничений. Как правило, эти алгоритмы базируются на решении задачи византийских генералов [1], наиболее известным из них является Practical Byzantine Fault Tolerance (PBFT) [2].

В рамках данного исследования проводится сравнение работы двух похожих платформ для формирования распределенных реестров (Distributed Ledger Technology, DLT) на базе PBFT-подобного консенсуса и определяется область их применимости.

2. Формулировка задачи. Необходимо сравнить эффективность работы двух платформ на базе PBFT-подобного консенсуса с точки зрения быстродействия и нагрузки на сеть.

Первая, Hyperledger Sawtooth [3] – корпоративное решение с открытым исходным кодом, имеющее в своем составе несколько готовых алгоритмов консенсуса, в частности, оригинальный алгоритм PBFT [4]. Вторая, BGX/DGT – решение на базе Hyperledger Sawtooth со встроенным алгоритмом F-BFT, который является двухуровневой модификацией PBFT.

3. Методика. Сравнение эффективности проводилось в рамках четырех различных тестов, выполненных для сетей, состоящих из 6, 12, 15, 18 и 21 узлов. Выбор такого количества узлов обусловлен стандартной конфигурацией BGX/DGT, ее изменение в данной версии может повлиять на результаты тестирования.

Фараджов Анар Мурватович – студент, Санкт-Петербургский государственный университет e-mail: st069744@student.spbu.ru тел.: +7(996)769-56-18

В силу многоуровневости сети, в BGX/DGT наращивание количества узлов производилось путем увеличения числа задействованных кластеров: для 6 узлов – один кластер на 6 узлов, для 12 – два кластера по 6, для 15 – два кластера по 6 и один по 3, для 18 – два кластера по 6 и два по 3, для 21 – два по 6 и три по 3 узла. Стоит обратить внимание на тот факт, что конфигурация для всех тестов остается неизменной: первый и второй кластеры состоят из 6 узлов, третий, четвертый и пятый – из трех узлов.

В случае Sawtooth, в процессе его работы строится одноранговая сеть, так что разделение на кластеры условно и далее используется для унификации. Так, например, 11-й узел будет 5-м узлом 2-го кластера, а 19-й – 1-м узлом 5-го кластера.

В рамках данного исследования платформы сравнивались по двум критериям – средний суммарный объем трафика на транзакцию, измеренный на валидаторах узлов и среднее количество принятых всеми узлами транзакций в секунду.

4. Алгоритм F-BFT. Консенсус, реализованный в BGX/DGT, можно условно разделить на две фазы. Во время первой фазы блок принимается внутри кластера, инициирующего транзакцию, для этого используется алгоритм PBFT. После того как блок принят в кластере, решение о принятии блока в блокчейн распространяется по всей сети с помощью алгоритма арбитража.

5. Тестовая среда. В данном исследовании использовались BGX/DGT версии Kawartha(0.4) [5] и Sawtooth PBFT версии 1.2.6 [6]. Типовой узел для обеих платформ состоит из нескольких служб:

- валидатор;
- модуль REST-API;
- процессор тестовых транзакций;
- процессор настроек.

В качестве процессора транзакций для Sawtooth использовался процессор `intkey`, для BGX/DGT – `bgt`. Каждый сервис запускался и изолировался средствами `Docker` [7] и `Docker Compose` [8]. Все измерения производились с помощью `Docker`, `vnStat`, а также встроенными средствами Sawtooth.

6. Результаты. Далее приведены результаты измерений, сделанных по четырем тестам.

Тест 1. Генерация одиночных транзакций. В рамках этого теста одиночные транзакции поочередно посылались из каждого узла. Каждая следующая транзакция отправлялась только после получения ответа о принятии предыдущей.

Таблица 1. Скорость наполнения реестра, тр./с

Кол-во узлов	Sawtooth PBFT	BGX/DGT
6	0,85	0,62
12	0,8	1,20
15	0,83	1,07
18	0,62	1,09
21	0,87	1,18

Таблица 2. Средний объем трафика, кб/тр.

Кол-во узлов	Sawtooth PBFT	BGX/DGT
6	81,2	266
12	168	162,7
15	212,8	207
18	293,1	234,9
21	340	206

По результатам данного теста можно видеть (см. таблицы 1 и 2), что в Sawtooth PBFT, в отличие от BGX/DGT, объем трафика значительно возрастает при увеличении числа задействованных узлов.

Тест 2. Генерация потока транзакций. В рамках этого теста из второго узла каждого задействованного кластера посылался поток из 100 транзакций. Внутри потока каждая следующая транзакция отправлялась только после получения ответа о принятии предыдущей в реестр.

Таблица 3. Скорость наполнения реестра, тр./с

Кол-во узлов	Sawtooth PBFT	BGX/DGT
6	0,74	1,23
12	1,85	1,49
15	2,43	1,72
18	4,54	1,09
21	5,55	0,81

Таблица 4. Средний объем трафика, кб/тр.

Кол-во узлов	Sawtooth PBFT	BGX/DGT
6	66	180,8
12	109,2	185
15	121,6	142,66
18	108,8	209
21	115	232,7

Следует обратить внимание на тот факт, что при выполнении теста в BGX/DGT для 12, 15, 18 и 21 узлов, отношение числа занесенных в реестр транзакций к посланным оказалось равно 0,57, 0,49, 0,36 и 0,28 соответственно. В остальных случаях это соотношение равно 1.

Из результатов видно (см. таблицы 3 и 4), что Sawtooth PBFT по обоим показателям выдерживает повышение нагрузки лучше, чем это делает BGX/DGT.

Тест 3. Генерация одновременных транзакций для единичных узлов разных кластеров. В рамках этого теста из второго узла каждого задействованного кластера посылались серия одновременных транзакций. В результате в реестр BGX/DGT не была занесена ни одна из посланных транзакций, поэтому в таблице 5 приведены результаты только для Sawtooth PBFT.

Таблица 5. Результаты теста для Sawtooth PBFT

Кол-во узлов	Скор. наполнения, тр/с	Объем трафика, кб/тр
12	0,8	109,2
15	3,13	110
18	3,8	123,4
21	4,76	112,6

Тест 4. Генерация одновременных транзакций для всех узлов сети. В рамках этого теста из каждого задействованного узла посылались серия одновременных транзакций. В результате в реестр BGX/DGT не была занесена ни одна из посланных транзакций, поэтому в таблице 6 приведены результаты только для Sawtooth PBFT.

Таблица 6. Результаты теста для Sawtooth PBFT

Кол-во узлов	Скор. наполнения, тр/с	Объем трафика, кб/тр
6	0,87	42,5
12	1,18	101,6
15	2,78	75,8
18	3,45	103,8
21	3,85	129,9

Следует заметить, что при выполнении теста для 21 узла отношение количества занесенных в реестр транзакций к посланным составляло 0,95. Для других количеств это отношение равно 1.

7. Заключение. Проведено сравнение двух DLT решений по нескольким критериям и проверено их поведение в ситуациях, которые с высокой вероятностью могут возникнуть в процессе работы системы.

По результатом тестов можно заключить, что BGX/DGT больше подходит для построения больших сетей с низким требованиями к надежности и безопасности, в то время как платформа Sawtooth PBFT может быть использована в небольших сетях с повышенными требованиями к надежности и более высокой нагрузкой.

Литература

1. Lamport L., Shostak R., Pease M. The Byzantine generals problem // ACM Transactions on Programming Languages and Systems. 1982. Vol. 4. No 4. P. 382–401.
2. Castro M., Liskov B. Practical byzantine fault tolerance and proactive recovery // ACM Transactions on Computer Systems. 2002. Vol. 20. No 4. P. 398–461.
3. Hyperledger Sawtooth documentation [Электронный ресурс]: URL:<https://sawtooth.hyperledger.org/docs/core/releases/1.2.6/> (дата обращения: 19.03.2021).
4. Introduction to Sawtooth PBFT [Электронный ресурс]: URL:<https://sawtooth.hyperledger.org/docs/pbft/releases/latest/introduction-to-sawtooth-pbft.html> (дата обращения: 19.03.2021).
5. Репозиторий BGX/DGT [Электронный ресурс]: URL:<https://github.com/DGT-Network/DGT-Kawartha> (дата обращения: 17.03.2021).
6. Репозиторий Hyperledger Sawtooth [Электронный ресурс]: URL:<https://github.com/hyperledger/sawtooth-core> (дата обращения: 15.03.2021).
7. Overview of Docker Engine [Электронный ресурс]: URL:<https://docs.docker.com/engine> (дата обращения: 15.03.2021).
8. Overview of Docker Compose [Электронный ресурс]: URL:<https://docs.docker.com/compose> (дата обращения: 15.03.2021).

УДК 004.582

Цепелева Р. В., Павлова Е. Д., Чеботарева А. А.

Имплементация протокола по кроссчейн-взаимодействию

Рекомендовано к публикации доцентом Корховым В. В.

1. Введение. Сегодня в блокчейне активно продвигается концепция децентрализованных финансов (DeFi [1]), основная задача которой стать альтернативой банковскому сектору и заменить традиционные технологии нынешней финансовой системы протоколами с открытым исходным кодом. Индустрия DeFi задумывалась как единая экосистема, продукты которой могли быть совместимы друг с другом. Но дело в том, что изначально каждая сеть блокчейна создавалась как независимая и автономно работающая. Именно поэтому вопрос объединения сетей блокчейн является актуальным вот уже несколько лет.

2. Постановка задачи. Задача заключается в создании проекта, который сделает возможным передачу цифровых активов из одной сети блокчейна в другую. Для решения задачи планируется написание смарт-контрактов и иного программного кода.

3. Обзор существующих методов и подходов передачи цифровых активов между блокчейн-сетями. На данный момент существуют готовые или относительно готовые проекты по решению проблемы объединения и взаимодействия блокчейнов. Рассмотрим их подробнее.

3.1. Экосистема Polkadot, проекты на Polkadot. Polkadot [2] – это сеть, которая соединяет блокчейны. Она создает пространство, где могут быстро и безопасно обрабатываться и обмениваться данные из разных блокчейнов.

Экосистема Polkadot состоит из трех разных компонентов:

Цепелева Рита Вячеславовна – студент, Санкт-Петербургский государственный университет; e-mail: st062153@student.spbu.ru, тел.: +7(981)698-55-88

Павлова Екатерина Денисовна – студент, Санкт-Петербургский государственный университет; e-mail: e.d.pavlova@spbuniv.ru, тел.: +7(999)224-39-74

Чеботарева Ангелина Андреевна – студент, Санкт-Петербургский государственный университет; e-mail: angel.xe99@gmail.com, тел.: +7(981)773-93-81

1. *Связующая сеть (Relay chain)*. Это центр системы, который обеспечивает обмен транзакциями между цепями. Она также гарантирует консенсус и безопасность.
2. *Парачейны (Parachains)*. Это «параллельные блокчейны», которые осуществляют транзакции и переносят их в исходный блокчейн.
3. *Мосты (Bridges)*. Это особые ссылки на блокчейны со своим собственным консенсусом.

Основной недостаток проектов на Polkadot заключается в безопасности. В них процесс обмена реализован через заморозку токенов, подразумевающую под собой перевод пользователем его средств, которые он хочет обменять, в так называемое хранилище. Оно работает по следующей логике: если токены пришли в хранилище, то их удерживать, а на указанный адрес в целевой сети необходимо начислить эквивалентное их количество. Вопрос безопасности заключается в том, что хранилище представляет собой обычный адрес, а не контракт. Соответственно нет гарантий и защиты пользователей от обмана или технического сбоя.

3.2. Мосты. Блокчейн-мост – это взаимосвязанное звено, которое обеспечивает связь и взаимодействие между двумя системами блокчейнов. Блокчейн-мосты, связывая две сети блокчейнов, помогают децентрализованным приложениям использовать преимущества обеих систем, а не только их хост-платформу. Эти мосты регулируются протоколом *mint-and-burn*. Передача токенов происходит не буквально; скорее, когда токен необходим для передачи из одного блокчейна в другой, он сжигается на первом, а эквивалентный токен чеканится на другом.

Недостаток такого подхода заключается лишь в ограниченности. Один мост в состоянии соединить лишь два блокчейна. Таким образом, говорить о единой экосистеме, использующей лишь один мост, не представляется возможным.

4. Идея решения Wish Swap. Токен Wish имеет формат BEP2 и размещен в блокчейне Binance [3], что ограничивает его применение. Необходимо разработать механизм обмена токенов между сетями Ethereum(ERC20) [4], Binance (BEP2) и Binance Smart Chain (BEP20) [3].

5. Реализация проекта Wish Swap. Рассмотрим детали и инструменты реализации проекта более подробно.

5.1. Контракты токенов. Для того чтобы совершить обмен токена Wish, было принято решение реализовать контракты аналогов токена в сетях Binance Smart Chain (BEP20) и Ethereum (ERC-20) с названиями BWish и WWish, соответственно. Все контракты проекта были написаны на языке Solidity – JavaScript-подобном объектно-ориентированном, кроссплатформенном языке для разработки смарт-контрактов. Контракты токенов, помимо стандартных функций и полей, должны были содержать функции и события `transferToEthereum/transferToBSC` (функция перевода в сеть Ethereum, либо Binance Smart Chain; в зависимости от сети) и `transferToBC` (перевод токенов в Binance Chain), а также функции `mint` и `burn`. Данные функции позволяют выпускать/сжигать токены и доступны только владельцу контракта (в нашем случае бэкенду) для предотвращения неконтролируемого выпуска/сжигания токенов. Помимо контрактов токенов были реализованы бэкенд и сканер на языке Python.

5.2. Перевод из сети Binance Chain. В сети Binance Chain создан адрес для обмена с блокчейнами Ethereum и Binance Smart Chain. Адрес является «swap-контрактом», переводы на который должен ловить сканер. Транзакции в сети Binance содержат поле `Мемо`, которое может содержать дополнительную информацию. Пользователь, желающий совершить обмен токенов Wish из сети Binance Chain, через фронтенд (Django+React) отправляет свои BEP20 Wish токены на данный адрес, заполняя по правилам поле `мемо`. Сканер просматривает операции с токеном BEP20 Wish. При обнаружении транзакции на обменный адрес отправляет ее в бэк посредством RabbitMQ. Бэкенд обращается к контракту токена BEP20 BWish и выпускает токены на адрес пользователя за вычетом установленной комиссии (за перевод предусмотрена комиссия в токенах целевой сети). Токены в сети Binance не могут быть уничтожены и они остаются на обменном адресе, доступ к отправке с обменного адреса имеет только бэкенд. Таким образом, количество токенов на обменном адресе соответствует количеству токенов в сетях Binance Smart Chain и Ethereum.

5.3. Перевод из сетей Binance Smart Chain и Ethereum.

Обмен токенов, осуществляемый из указанных сетей практически аналогичен за исключением того, что здесь пользователи будут взаимодействовать не с адресом, а со смарт-контрактом, а токены, при успешном их зачислении на целевой счет, будут сжигаться.

5.4. Архитектура проекта. Итоговая архитектура проекта изображена на рис. 1.

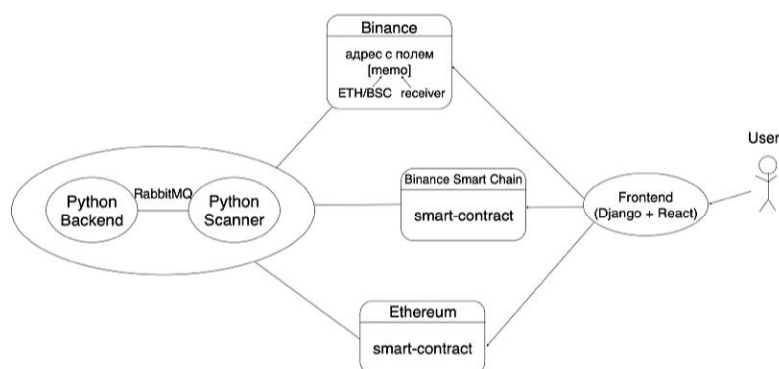


Рис. 1. Архитектура проекта

Шаблон смарт-контракта токена включает функции:

- `transferToBC` и `transferToBSC/transferToEthereum` – перевод токенов в сети Binance Chain и Binance Smart Chain/Ethereum;
- `mint` – функция, необходимая для начисления токенов на определенный адрес;
- `burn` – сжигает токены.

Веб-приложение Django + React выполняет задачи: сбора информации от пользователя и формирования транзакций.

Сканеры сетей Ethereum и Binance Smart Chain и сканер сети Binance Chain обеспечивают сканирование всех транзакций в открытом реестре для отсылки сообщений с подтверждениями на бэкенд через очередь (RabbitMQ).

5.5. Безопасность. В данном проекте вся логика строится на хранении токенов на смарт-контрактах и обычном адресе в сети Binance. Соответственно, для их перераспределения необходимо хра-

нить приватный ключ, что ставит под вопрос безопасность средств пользователей. Если данным ключом завладеет кто-либо, он сможет распоряжаться цифровыми активами. Поэтому мы реализовали шифрование и не храним приватный ключ в чистом виде.

6. Анализ полученных результатов. Сравним полученное решение с аналогичными проектами:

Таблица. Сравнение результатов

	Wish Swap	PolkaSwap	Panama Bridge
Безопасность	да	нет	да
Реализация	да	нет	да
Универсальность	нет	да	нет
Масштабируемость	да	да	нет
Комиссия	1 Wish	0,3%	0,001 BNB

Таким образом видно, что наше решение удовлетворяет требованиям безопасности, а также имеет перспективы развития при добавлении новых блокчейнов.

7. Заключение. Еще несколько лет назад перевод токенов и какой-либо другой информации из одной сети в другую не представлялся возможным. Однако сегодня мы показали, что в данном направлении можно строить безопасные решения.

Литература

1. Zetsche D., Arner D., Buckley R. Blockchain disruption and decentralized finance: the rise of decentralized business models // Journal of Financial Regulation. 2020. Vol. 6. Iss. 2. P. 172–203.
2. Polkadot [Электронный ресурс]: URL:<https://polkadot.network/> (дата обращения: 11.03.2021).
3. Binance [Электронный ресурс]: URL:<https://docs.binance.org> (дата обращения: 11.03.2021).
4. Ethereum [Электронный ресурс]: URL:<https://ethereum.org> (дата обращения: 11.03.2021).

UDC 004.921

Bai M., Li S.

Comparative study of various edge detection operator

Recommended for publication by associate professor Petrosian O. L.

1. Introduction. The edge of the image is one of the most basic features of the image, and is the most significant part of the local intensity change of the image. In essence, the image edge is a response to the discontinuity of local characteristics of the image. Edge detection is a key step in image preprocessing, and the detection results have a direct impact on the later feature extraction and image processing. In order to obtain better edge effect, there are many edge detection algorithms and improved algorithms of existing edge detection operators [1].

2. Principle of edge detection. The edge of the image has two characteristics: direction and amplitude. Edge detection is the measurement and location of gray change. Image edge detection can generally be divided into four steps: filtering, enhancement, detection and positioning.

3. Sobel edge detection operator. The operator is obtained by convolution of the original image with two convolution kernels. The mathematical expression is

$$S = (x, y) = \max \left(\sum_{m=1}^M \sum_{n=1}^N f(m, n) g_1(i - m, j - n), \sum_{m=1}^M \sum_{n=1}^N f(m, n) g_2(i - m, j - n) \right).$$

Sobel edge detection operator is based on the first-order derivative. The calculation method of the operator is as follows:

$$G_x = f(x + 1, y - 1) + 2f(x + 1, y) + f(x + 1, y + 1) - f(x - 1, y - 1) - 2f(x - 1, y) - f(x - 1, y + 1),$$

Bai Ming – undergraduate student, Yanan University;
e-mail: 1579844382@qq.com, phone: +8615591120195

Li Shuai – postgraduate student, Yanan University;
e-mail: ls98818@163.com, phone: +8618700258174

$$G_y = f(x-1, y+1) + 2f(x, y+1) + f(x+1, y+1) - f(x-1, y-1) - 2f(x, y-1) - f(x+1, y-1).$$

The vertical and horizontal templates for Sobel operators are shown in figure 1(a). The horizontal edge of the image can be detected, and (b) the vertical edge of the image can be detected. Basic idea: the use of pixels on the upper and lower, left and right adjacent points of the gray weighted algorithm [2] for image plane convolution. Taking the maximum of two convolutions as the output of the point, an edge amplitude image is obtained.

-1	0	1
-2	0	2
-1	0	1

(a)

-1	-2	-1
0	0	0
1	2	1

(b)

Figure 1. Sobel detection operator

4. Log edge extraction operator. Log detector is a second derivative edge detector, which combines the advantages of Laplacian detector and Gaussian filter. Basic idea: Gaussian operator is used to smooth the original image and reduce noise, then Laplace operator is used to convert the edge points into zero crossing points, and then the edge detection is realized by second derivative zero crossing points. This can overcome the shortcomings of Laplace operator sensitive to noise and reduce the influence of noise. Gaussian function is

$$m(x, y) = e^{-\frac{x^2+y^2}{2\sigma^2}}.$$

Then the log edge detection operator for the continuous function $h(x, y)$ is

$$G(x, y) = -\nabla^2[m(x, y)h(x, y)] = [-\nabla^2]h(x, y) = M(x, y)h(x, y),$$

$$M(x, y) = -\nabla^2 m(x, y) = \frac{\sigma^2 - \gamma^2}{\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}},$$

where σ is standard deviation. $M(x, y)$ is an axisymmetric function. Since this function is very smooth and can reduce the influence of noise, when the edge is blurred or the noise is large, the use of sub-function to detect zero crossing can provide a more reliable edge position [3]. Common Log operators are templates for 5×5 , as shown in figure 2.

0	0	-1	0	0
0	-1	-1	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

(a)

-2	-4	-4	-4	-2
-4	0	8	0	-4
-4	8	24	8	-4
-4	0	8	0	-4
-2	-4	-4	-4	-2

(b)

Figure 2. Log edge detection operator

5. Canny edge detection operator. Because the Laplacian detection operator is sensitive to noise, the effect is not very ideal in processing actual images. Canny operator studies the characteristics of the optimal edge detector: high accuracy, high accuracy, single pixel width. According to these three characteristics, Canny proposed three optimal criteria of the first-order differential filter for edge detection: maximum signal-to-noise ratio criterion, positioning accuracy criterion and single edge response criterion.

(1) Signal-noise criterion

$$SNR = \frac{|\int_{-W}^W F(-x)g(x)dx|}{\sigma\sqrt{\int_{-W}^W g^2(x)dx}},$$

where $F(x)$ is the edge function, $g(x)$ is the impulse response of low-pass filter with W bandwidth, σ is the variance of Gaussian noise.

(2) Positioning accuracy criterion defined as

$$M = \frac{|\int_{-W}^W F'(-x)g'(x)dx|}{\sigma\sqrt{\int_{-W}^W g'^2(x)dx}},$$

where $F'(-x)$ and $g'(x)$ are the first-order derivatives of $F(x)$ and $g(x)$, and M is the measurement of the accuracy of edge positioning. The greater M is, the higher the accuracy is.

(3) Single edge response principle. If a single edge has only one response, the average distance of the zero crossing point of the impulse response derivative of the detector should satisfy:

$$D(f') = \vartheta \sqrt{\frac{\int_{-\infty}^{\infty} g'^2(x)dx}{\int_{-W}^W g''(x)dx}},$$

where $g''(x)$ is the second derivative of $g(x)$, f' is the image after edge

detection. Canny operator is an optimization operator based on first derivative, which integrates filtering, enhancement and detection.

6. Analysis of effect. In order to compare the edge detection effects of several operators, the edge function in Matlab is used to detect the original shoot image and the shoot image after adding salt-pepper noise and Gaussian noise respectively. The test results are shown in figures 3–5.

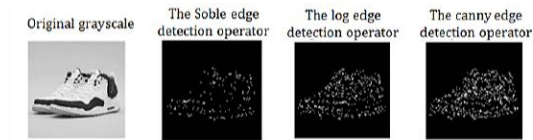


Figure 3. Edge detection results without noise

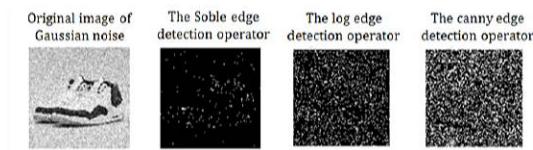


Figure 4. Edge detection results with Gaussian noise

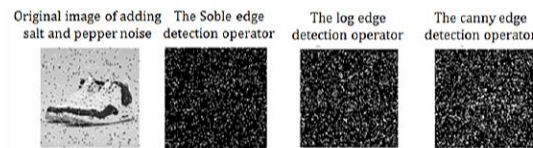


Figure 5. Edge detection results by adding salt and pepper noise

The results in figure 3 show that Sobel edge detection operator can only detect the contour of the image, with more discontinuous points, poor edge continuity and loss of some details. Log operator edge continuity is better than Sobel operator, details are highlighted. Compared with Sobel operator, Log operator has better detection effect. The edge image obtained by Canny operator is more complete. Compared with the former two, Canny operator has better detection effect.

The results in figure 4 and 5 show that Sobel operator has good noise immunity, but the detection accuracy is not high, there are many discontinuous points, and the image is blurred. Log operator has good edge integrity, but is sensitive to Gaussian noise and pepper salt noise; Canny operator has high edge positioning accuracy and good continuity, but it is easy to smooth out some edge information.

7. Conclusion. Through the above experiments, the advantages and disadvantages of the three operators are analyzed, and it is concluded that Canny operator is a better edge detection algorithm in the three operators, but it is not the best. Sobel operator is suitable for images with gray gradient and more noise; Log operator often has double edge pixel boundaries, which is sensitive to noise and suitable for edge detection of low noise or noiseless images. Canny operator is not easily disturbed by salt and pepper noise, can detect the real weak edge, high positioning accuracy, but greatly affected by Gaussian noise. Most of the time, one operator cannot meet the needs, and it needs a variety of methods to be used comprehensively. It is hoped that readers can flexibly select operators according to the advantages and disadvantages of various operators summarized in this paper.

References

1. Wu Li. Research on image edge detection algorithm in Matlab // Beijing Surveying and Mapping. 2014. No 3. P. 3-7.
2. Julong Deng. Grey Theoretical Basis // Huazhong University of Science and Technology. 2002. P. 5.
3. Shaosheng Fan., Hainan Wang. Multi-direction Fuzzy Morphology Algorithm for Image Edge Detection // Journal of Networks. 2011. No 6. P. 895-898.

UDC 004.8

Kimura K., Tu Y., Tanji R., Mozgovoy M.

Identifying winning actions in a 3D tennis game with Monte Carlo tree search

1. Abstract. AI nowadays plays an important role in many areas including video games. AI-controlled agents have become an essential part of video game worlds, inhabited by both intelligent opponents and friendly characters that make games more interesting and interactive. In certain types of games, notably sports games, it is important to provide opponents of different skill levels. One way to achieve such tunable AI is to employ an automated optimization method. Monte-Carlo tree search (MCTS) has been successfully used for this purpose in card and board games, such as chess and poker. We explore the possibility to apply MCTS in an action sports game of 3D tennis, and show how a dataset of pre-recorded tennis games can be used to train MCTS to overcome an entry-level built-in tennis AI system.

2. Introduction. Among methods of building game AI, the most common one is Rule-Base AI. It is simple and powerful. But it requires the implementer to understand the game logic well and be able to think about all game patterns, which is difficult.

This disadvantage can be overcome by using heuristic algorithms such as MCTS [1] to select sequentially plausible actions. In general, MCTS-family algorithms represent game process as a tree with nodes corresponding to game states. Then a certain procedure is used to explore the tree and find the most effective actions.

Kimura Kaito – undergraduate student, The University of Aizu; e-mail: s1250131@u-aizu.ac.jp, phone: +81-242-37-2664

Yuan Tu – undergraduate student, The University of Aizu; e-mail: s1252005@u-aizu.ac.jp, phone: +81-242-37-2664

Tanji Riku – undergraduate student, The University of Aizu; e-mail: s1270139@u-aizu.ac.jp, phone: +81-242-37-2664

Mozgovoy Maxim – associate professor, The University of Aizu; e-mail: mozgovoy@u-aizu.ac.jp, phone: +81-242-37-2664

In this study we rely on Monte-Carlo tree search to build an AI system for a 3D tennis game.

3. Monte Carlo tree search method.

Monte-Carlo tree search consists of the following four steps:

1. Selection
2. Expansion
3. Simulation
4. Backpropagation

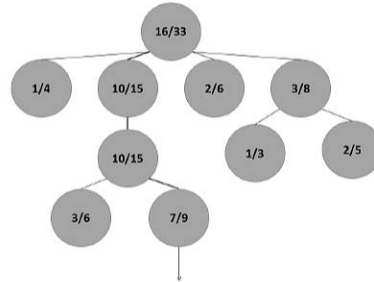


Figure 1. MCTS steps

By repeating these steps, MCTS is able to build a game tree and find branches corresponding to efficient decisions (see Fig. 2). The root node represents the current game state. From there, valid moves are explored. Each node has a record of the number of all attempts and scored attempts. The node with the highest ratio of scored attempts to all attempts is considered the most effective.

For the selection step, the UCT (Upper Confidence Trees) [4] value is commonly used. AI selects a node that has the highest UCT value from the root node to the leaf node. UCT uses the UCB1 algorithm [5] for action selection. UCB1 calculates the score of a node as

$$UCT = X_i + 2C_p \sqrt{\frac{2 \ln N_i}{n_i}}.$$

The first term x_i is a reward.

$$X_i = p_i + (1 - p_i) \frac{w_i}{n_i}.$$

In this formula, p_i stands for the potential of the move (pattern) score directly, w_i stands for the number of wins for the node considered after the i -th move, and n_i stands for the number of simulations for the node considered after the i -th move.

The value of w_i/n_i represents a possibility of winning of this action. In general, w_i/n_i is used as the reward term. In the game of tennis, however, every shot has the potential to score directly. Therefore, we adapt Equation (2) as the reward term. The scoring possibility p_i is

calculated from extracted shot pattern in the process of the simulation. The subsequent term is for biasing the search towards nodes that have not been explored enough. An abnormally high value is calculated when the number of child nodes n_i is too much smaller than the number of simulations for parent node N_i . C_p is a constant that specifies degree of bias toward unexplored nodes. If there are not enough searches for valid nodes, the searching for the node with smaller value is finished. Otherwise, the node with the highest winning rate is searched in priority.

In the expansion step, the game tree is expanded by adding child nodes if a leaf node has been explored beyond a certain threshold.

In the simulation step, the result of the game from the leaf nodes is simulated. The result of the match is reflected in all the nodes selected in the back propagation step.

By repeating these steps, the AI judges the node with the highest X_i value of reward as the valid move and acts on it.

4. Adopting MCTS

for tennis. Our work aims to build an AI system for the game “World of Tennis” (see Fig. 1). Currently this game has two built-in system: one based on



Figure 2. World of Tennis

rules, and another based on case-based reasoning [3, 2]. Three types of basic movements are provided in the game:

- Serve
- Return the shot
- Recovery movement

The player and the opponent are able to serve, make shots of different types, and move to a new position after returning a shot.

In the tennis game, the game flow is a repetition of each players action which can be represented as a game tree. In this case, each node of the game tree should contain the following information:

- Shot from: AI or opponent
- AI position

- Opponent position
- Target position

Each node contains three coordinates: the AIs, the opponent's, and the target points coordinates. All of which are indexed by dividing the court into 48 sections (see Fig. 3) to simplify game representation with a tree.

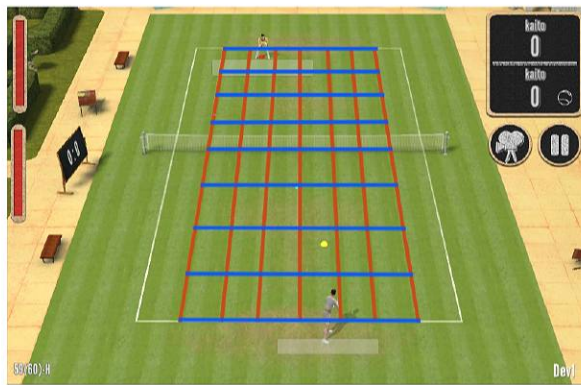


Figure 3. Tennis court divided into sections

The opponent will be in the upper court (0–23), and the AI will be in the lower court (24–47). The root node represents the current state where the opponent in position 3 is hitting the ball towards position 39. The following child nodes represent two possible effective moves for the next

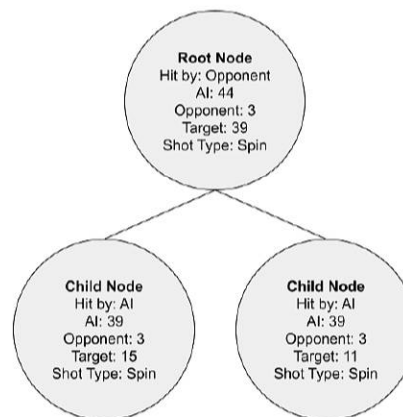


Figure 4. Node states for MCTS

move, i. e., after the AI returns the ball. Two possible next moves are both move to position 39, then return the shot to different target positions. We only showed a part of the game tree in Fig. 4, in our case, there are 24 return positions, each position has two types of shots. In total 48 nodes should be generated. In this way, the state of the tennis game is appropriately represented by the game tree.

5. Simulation logic. In MCTS, the AI simulates the game with a random selection from leaf nodes. In each simulation, the AI needs to judge whether the game will eventually result in a score or a loss of points. The accuracy of these simulations is very important in building a strong AI because it is the only standard for distinguishing strong and weak moves of AI.

In this work, more than 1400 scoring patterns from statistical data of 10,000 actual games data were extracted. The coordinates of each player and the ball, the shot types are stored as time-series data. From each match data, we extract the following information for simulation:

- Attacker (player's) position
- Defender (opponent's) position
- Target position
- Shot type
- Whether this shot is scoring or not

Player and opponent positions, shot target, intended shot type, and shot outcome directly determine the information for each game pattern.

6. Results. As mentioned above, World of Tennis has two built-in AI engines. One of them was used to create an entry-level "Coach AI" system aimed to introduce basic game patterns to beginners. We played a series of games against the Coach AI to estimate MCTS performance. As a result, MCTS-based AI won 7 games out of 10. The total score was 59:44. The scores of the matches show that the AI is able to take rational actions.

7. Conclusion. In this paper, we have created an AI for a 3D tennis game using a heuristic algorithm Monte Carlo tree search. By properly modeling complex game states, the AI can represent game process as tree and perform the search. MCTS learns whether a move at a node is valid or not by repeatedly simulating the game state of the node.

Therefore, the accuracy of the simulation is important. In this study, we conducted simulations by extracting patterns that are likely to be scored from actual game data. As a result, we were able to simulate with a certain level of accuracy and construct a working agent.

References

1. Browne C. et al. A survey of Monte Carlo tree search methods // IEEE Transactions on Computational Intelligence and AI in games 4.1. 2012. P. 1–43.
2. Mozgovoy M. Context-awareness and anticipation in a tennis video game AI system // Proceedings of the 2018 IEEE International Conference on Systems, Man, and Cybernetics, Miyazaki, Japan, 2018. P. 699–703.
3. Shinsei Kinouchi. Identifying Key Elements of Successful Behavior in a Video Game of Tennis. The University of Aizu, 2019.
4. Kocsis L., Szepesvari C. Bandit based monte-carlo planning // in European conference on machine learning. Springer, 2006. P. 282–293.
5. Auer P., Cesa-Bianchi N., Fischer P. Finite-time analysis of the multiarmed bandit problem // Mach. Learn. 2002. Vol. 47. No. 2. P. 235–256.

UDC 004.4

Li Y., Fan F.

Design and implementation of online education system based on B2C mode

Recommended for publication by associate professor Petrosian O. L.

1. Abstract. This system has set up a user-friendly online education platform with a complete set of functions [1]. The system is developed by separating front and back-ends. The back-end uses the Spring-Cloud microservice structure. The front-end uses the VUE structure. The persistence layer is designed using the MyBatis-Plus microservice sub-library, and uses Swagger to generate interface documents to access Aliyun Video on Demand and Aliyun OSS.

2. Introduction. Online education is a network-mediated teaching method. Through the Internet, students and teachers can carry out teaching activities even if they are separated by thousands of miles [2]. In addition, with the help of network courseware, students can also study anytime and anywhere, truly breaking the time and space restrictions. For these reasons, online distance education is the most convenient way for people who are busy with work and have no fixed study time. Especially under the influence of COVID-19, online education has become an indispensable educational method for enterprises, schools and other places [3]. The platform is based on the SpringCloud microservice framework and uses MySQL to store and manage the project data files. In the third part, we mainly introduce the relevant technologies used in the system.

3. Technologies used in the system. Front-end technologies include: Vue CLI4 and Node.js. Back-end technologies include: Spring-Boot, SpringCloud, Mybatis Plus, Redis, and EasyExcel. Third-party technologies include: Aliyun OSS, Video on demand, Text messaging service. The database uses MySQL 5.7 version.

Li Yike – undergraduate student, Yanan University;
e-mail: liyike1127@163.com, phone: +8617806812872

Fan Fan – undergraduate student, Yanan University;
e-mail: 1509409083@qq.com, phone: +8618581507683

3.1. General framework. We divide the system into two modules: the front-end and the back-end. The front-end adopts the Vue CLI4 framework, the back-end adopts the SpringCloud microservice architecture and the single application framework SpringBoot, while the persistence layer uses the MyBatis+Generator framework. Its front-end responsive framework uses the Bootstrap framework to support multi-terminal system compatibility. At the same time, Redis uses distributed cache to better process video files.

3.2. Function module design. The system provides users with an online education platform with a beautiful interface and simple operation methods. The following figure shows the home page of the platform.

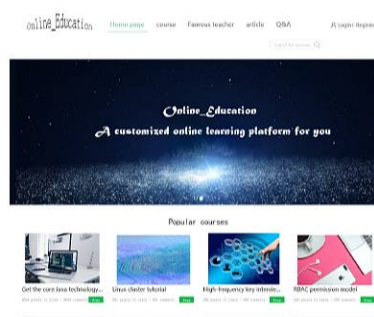


Figure 1. System home page

The application layer of this platform is a mobile phone or web browser, nginx is embedded to facilitate the packaging and release of subsequent services, the data layer uses redis cache technology, and the database uses mysql for data storage and management. The overall structure of the system is shown in Figure 2:

3.3. Data display. Data display refers to the data display on the home page, the list of teachers and their information details, and the list of courses and their details. The purpose is to give users a direct understanding of what the platform contains so that users can master the methods of using the platform more quickly.

3.4. Course playback. Course playback module includes: video playback and course details, course fast upload and download caching

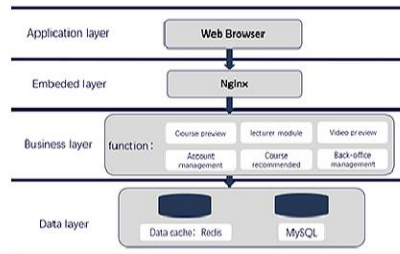


Figure 2. General diagram architecture diagram

function. The system provides high-speed download channel. The video files are processed to make downloading and uploading more easy and quick.

3.5. Information verification. The information verification consists of graphics verification code function, login and registration function, short message verification code function, and WeChat virtual scanning payment function. The system provides perfect verification functions, and stores video files and user information more safely, so that users can use them at ease.

3.6. User management. The user management is divided into two parts: lecturer management and student management. This module provides high-quality personnel information management functions.

3.7. Course management. The course management module includes: course classification management function module, course order management module. This module mainly manages and stores the course.

3.8. Statistical analysis. The statistical analysis includes: statistical analysis, authority management functions. The statistical analysis module allows analysts to visually see the popularity of courses and lecturers, and then get useful data to recommend courses and lecturers accordingly [4].

3.9. Database design. Designing a good database requires designers to be familiar with how the data is organized in order to make full use of its characteristics to design a database with high access per-

formance. Database design is an effective combination of technology and management. When designing a database, consideration should be given to how to obtain the required information and how to handle the information. Not only that, we should also consider how data are stored and how it is used. This project uses MySQL as the database. MySQL is a relational database management system, which has a wide range of applications due to its small size, fast speed, low overall cost of ownership and open source code.

Table 1. Teacher table

Field	Type	Lenth	Instruction
teacher.id	Integer	10	Number/ID (Primary key)
name	Varchar	4	Name (Cannot be empty)
tel	Integer	20	Tel number
password	Varchar	10	Password (Cannot be empty)

Table 2. Course table

Field	Type	Lenth	Instruction
course.id	Integer	10	Course no. (Primary key)
coursejs	Varchar	50	Introduction
coursename	Varchar	10	Name/Title (Cannot be empty)
coursetype	Integer	5	Course type
fbdate	Varchar	20	Public time
clicksum	Varchar	5	Click-through rate
teacher.id	Integer	10	Teacher no. (External key)

Table 3. Comment table

Field	Type	Lenth	Instruction
tiezi.id	Integer	10	Post no. (Primary Key)
subject	Varchar	10	Theme (Cannot be empty)
tiezinr	Varchar	50	Content
createtime	Varchar	20	Post time
hitcount	Integer	10	Visitor number
teacher.id	Integer	10	Teacher no. (External key)

4. Conclusion. Aiming at the current situation of online education and the main problems of online education, this product uses video file fragmentation rapid upload and a complete set of front-end separation projects to improve the quality of user experience. The system can be expanded, easy to maintain, and can build a safe and reliable online education platform for students and teachers to promote the development of online education.

References

1. Bhattacharyya S., Thander A. K. Conventional education and online education in engineering: a case study // *Journal of Physics: Conference Series*. 2021. Vol. 9. No 2. P. 15–28.
2. Morze N., Smyrnova-Trybulska E. Web-based community-supported online education during the COVID-19 pandemic // *International Journal of Web Based Communities*. 2021. Vol. 17. No 1. P. 12–30.
3. Locke B. B. Online education in the post-COVID era // *Online education in the post-COVID era*. 2021. Vol. 4. No 1. P. 3–10.
4. Liu T. Y., Zhang H., Du X. Y. Some strategies and techniques for layout in LaTeX // *Computer Knowledge and Technology*. 2019. Vol. 15. No 2. P. 15–33.

UDC 004.8

Liu Y., Chang G.

Implementation of facial mask detection in tensorflow based on improved SSD algorithm

Recommended for publication by associate professor Blekanov I. S.

1. Abstract. With the rapid spread of the new coronavirus epidemic in 2020, in order to do a good job of effective epidemic prevention, wearing masks has become the basic requirement for people. However, the traditional face recognition system will be affected by the mask and cannot normally complete the recognition work. People usually need to remove the mask for recognition. To solve this problem, this paper proposes an improved SSD algorithm and constructs a training model by tensorflow to realize the recognition of facial mask-wearing.

The model improves the extraction degree of facial features based on SSD algorithm combined with attention mechanism and introduces Dropout in the pooling layer of training stage to improve regularization, and finally completes image recognition and classification. The average experiment, the average accuracy of the model reached 95.76 %, and the average processing time of each picture reached 10.44 ms.

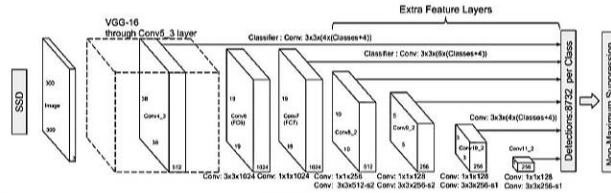


Figure 1. SSD algorithm framework

2. Overview of SSD algorithm. SSD algorithm is a model based on a convolutional neural network [1] (see fig. 1), which generates a multi-layer feature map by extracting image features. Select a six-layer feature

Liu Yuxing – undergraduate student, Yanan University;
e-mail: 1015925541@qq.com, phone: +8618992086235

Chang Geng – undergraduate student, Yanan University;
e-mail: 1173834432@qq.com, phone: +8613991051948

map to generate the default box for each pixel. Finally, the results are integrated into the Non-maximum suppression [2].

For a picture, it is input into the network to complete the training. Through the compression of the picture by each layer of neurons in the neural network, six layers (Conv4_3, Conv7, Conv8_2, Conv9_2, Conv10_2, Conv11_2) of different scales of feature maps are extracted, and then the recognition detection is carried out to obtain the results of the multilateral frame. Finally, the identification results are obtained by NMS processing the inconsistent borders.

3. Attention mechanism and dropout regularization. Under the mask, facial features will be greatly reduced, which makes the traditional recognition technology unable to correctly complete the recognition work. Therefore, an attention mechanism is introduced to solve this problem.

The attention mechanism is a deep learning technique. In the training of the neural network, this mechanism can make the neural network model pay attention to specific features, and improve the recognition degree of features by increasing the weight of noticed features. The process integrates and retains key information by extracting feature information from the original image to another image (see fig. 2).

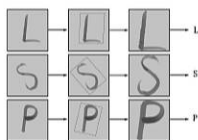


Figure 2. Attention-parametric grid sampling

Let us denote the input feature image coordinate point, let t denote the output feature image coordinate point, and Θ denotes the parameters obtained through the loss function. The following equation can be obtained:

$$\begin{pmatrix} x_i^s \\ y_i^s \end{pmatrix} = \tau_\theta = A_\theta \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix} = \begin{bmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{21} & \theta_{22} & \theta_{23} \end{bmatrix} \begin{pmatrix} x_i^t \\ y_i^t \\ 1 \end{pmatrix}.$$

The position of the input feature map coordinates corresponding to

the output feature map coordinates can be obtained by calculation. Finally, the feature results are obtained by differential image sampling.

Considering that the attention mechanism may produce over-fitting in the process of improving feature recognition, Dropout regularization technology is introduced to prevent over-fitting. In the training process, the work of some neurons is stopped randomly, so that the generalization of the model is improved, and there will be no excessively sensitive neurons to reduce the dependence on local characteristics [3].

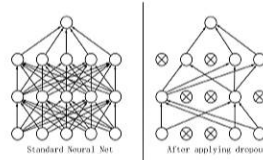


Figure 3. Dropout implementation mechanism

4. Improved SSD algorithm. According to the previous introduction, the SSD detection algorithm for wearing mask facial recognition accuracy is low, and can not extract facial features well. Therefore, this paper proposes to improve the recognition accuracy of the SSD algorithm by introducing an attention mechanism and Dropout technology.

4.1. Introducing attention mechanism. Attention mechanism can improve the neural network’s identification of features by increasing the weight. Therefore, the output feature map in the convolutional neural network is processed by the attention mechanism module. The feature weights of mask and face are improved, and then the results are output by inputting Detections [4].

4.2. Dropout optimization. In the process of training the neural network, due to the addition of attention mechanism, will produce sensi-

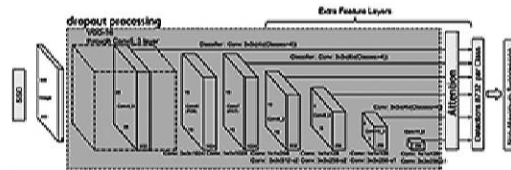


Figure 4. Improved SSD algorithm model

tive neurons, so Dropout optimization of neurons in the model. In each learning process, 50 % of the neurons are randomly selected to stop working, and the output of Dropout is extracted in the input Attention. In summary, the improved SSD algorithm model of figure 4 is obtained.

5. Implementation of improved SSD algorithm in tensorflow. Based on the above research, the improved SSD algorithm is simulated in the TensorFlow framework (see fig. 5).

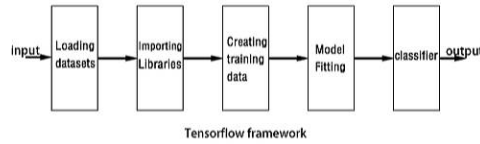


Figure 5. Tensorflow framework

In the process of model fitting, the improved SSD algorithm is used, and then 4180 face images retrieved in the network are used as learning materials. Including 2611 pictures of face wearing masks and 1569 pictures of face without masks.

According to the learning results, the model recognizes 4180 images five times, and the average running time is 31.68 s. The output classification results are obtained.

Table 1. Improved SSD Algorithm for Identifying Image Distribution Table

category	with_mask	without_mask	Sum
actual input	2611	1569	4180
identification result	2434	1689	4123
Average error	177	120	57

According to the simulation results, there are false positive errors in statistics, and some without_masks are identified as with_masks and the results are recorded or the opposite. An average of 17–43 copies per time.

Considering that the influence of false positive error is small, the false positive is considered to be correct, and the error is obtained according to the calculation.

In summary, the actual recognition accuracy of the model is about 95.76 %, which can identify the facial information of people wearing masks in images. Photos identified by NMS will directly mark the face of the person, as shown in figure 6.



Figure 6. Identification Output of Improved SSD Algorithm

6. Conclusion. With the development of neural network technology, the ability of computer to image recognition will be more accurate. Therefore, the improved SSD algorithm model has broad prospects for development. By introducing the attention mechanism and dropout technology to the traditional SSD algorithm, it not only greatly improves the recognition ability of the traditional recognition model for occlusion targets, but also has a strong expansibility. The problem of identifying occluded targets can be modified and implemented on this basis.

References

1. Zou H., Qin T. Research on the current situation of mask face detection method based on lightweight CNN // *Software*. 2020. Vol. 41. No 8. P. 186–188.
2. Ruan F. Research on pedestrian wearing mask detection based on improved SSD algorithm // *Science Technology and Economics Journal*. 2020. Vol. 28, No 35. P. 9–13.
3. Cheng H., Zeng H. Improved convolution neural network model averaging method based on Dropout // *Computer application*. 2019. Vol. 39. No 6. P. 1601–1606.
4. He H., Sun N. Image recognition algorithm based on channel domain self-attention mechanism // *In the computer age*. 2020. No 10. P. 64–67.

UDC 004.932.4

Qiao W., Zhang J.

Image fusion based on transform domain

Recommended for publication by associate professor Blekanov I. S.

1. Introduction. Image is the basis of human vision, an objective reflection of natural scenery, and an important source for human beings to understand the world and themselves. In daily life, when facing pictures and texts, people can always feel the special charm of graphics and images, but the untreated images often cannot fully reflect their expression. In recent years, image fusion has become a very important image analysis and computer vision technology. It has broad application prospects in computer vision, remote sensing, robot, medical image processing and military applications of automatic target recognition. Therefore, this study will improve the fusion effect that has the most clear image, the richest detail and reduce the amount of computation as the main research object.

2. Image fusion process based on transform domain method. Two or more images are fused by digital image fusion technology. Fourier transform is used to transform the image from the spatial domain to the frequency domain, and low-pass filtering, high-pass filtering and other processing are carried out. Then the image is converted from the frequency domain to the spatial domain by inverse Fourier transform for image reconstruction, so as to realize image enhancement and denoising, edge detection, and feature extraction.

3. Image fusion method. The overall research scheme as shown in the chart, using filter and image fusion technology, complete the software and hardware debugging, complete image input, image registration, image preprocessing, image fusion, etc. Finally, the image reconstruction is converted (as shown in figure 1).

3.1. Pretreatment of the image fusion. The obtained two kinds of image data are denoised, enhanced, corrected, segmented, and

Qiao Wenwen – underground student, Yanan University;
e-mail: 1390917752@qq.com, phone: +8618691146846

Zhang Jianpeng – underground student, Yanan University;
e-mail: 2392748644@qq.com, phone: +8617806815346

the registration feature points are selected. The data format and image size are unified, and the resolution of the original image space and spectrum are improved. The three-dimensional reconstruction and display of sequence fault images are carried out, and the mathematical model is established according to the characteristics of the target. Cross-correlation method and image registration method based on MATLAB transform domain are adopted.

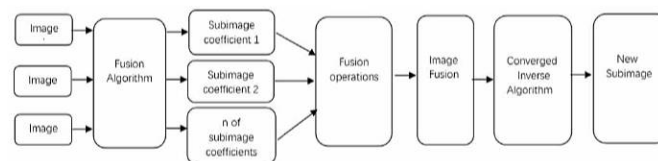


Figure 1. Image fusion method

3.2. Image fusion method realized by MATLAB. At present, the main fusion method is Fourier transform fusion method: Fourier transform is used to decompose the time domain signal into the sum of sine or cosine signals with different frequencies. In other words, Fourier transform provides another perspective to observe the image, which can transform the image from the gray distribution to the frequency distribution to observe the image. At this time, it is necessary to use the corresponding filters, such as the ideal filter and the Gaussian filter, and perform different operation in the frequency domain to achieve the expected image processing effect. Then the image is reconstructed by inverse Fourier transform to achieve image fusion.

4. Comparison of image reconstruction by different filtering methods. The frequency spectrum of the image is simply divided into the high frequency component and the low frequency component. The high frequency component represents the mutation part of the image, and the low frequency component represents the gentle region of the image. The image fusion is made with the following code:

```

subplot(1,3,1);
imshow (I1);
xlabel ('Albert Einstein');
figure(1)
subplot(1,3,2);

```

```

imshow (I2);
xlabel (' Marilyn Monroe');
I1=fftshift (fft2(double (I1)));
\% Fourier transform and move the zero point to the center
I2=fftshift (fft2(double (I2)));
[m,n,z]=size (I1);
radius =15.0;
h =fspecial ('gaussian',[ m, n], radius);
h =h./max (max (h));
\% Frequency Domain Filter
J_=I1_.*(1-h)+I2_.*h;
J =uint8(real (ifft2(ifftshift (J_))));
\% inverse transform figure (1)
subplot(1,3,3);
imshow (J);
xlabel ('Einstein +Monroe');
imwrite (J , 'out.jpg');
\% images saved to local
}

```

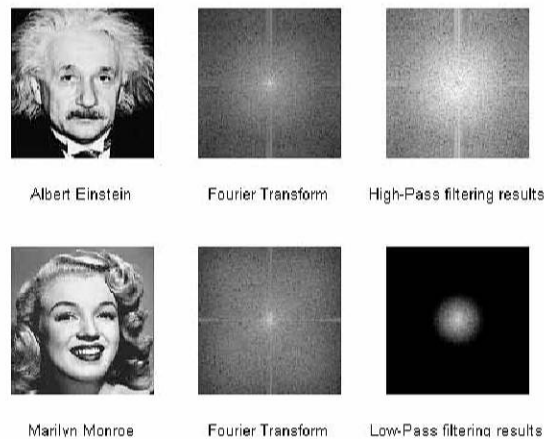


Figure 2. Two images of Einstein and Marilyn Monroe are filtered

Two images of Einstein and Marilyn Monroe are filtered (see figure 2), such as the high frequency component image of Einstein and the low

frequency component image of Marilyn Monroe (figure 3).

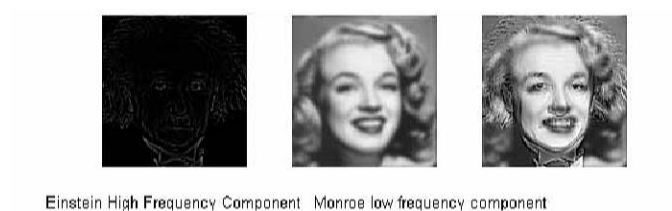


Figure 3. Einstein high-frequency component and Monroe low-frequency component

5. Conclusion. This kind of image fusion method is based on the Fourier analysis, which can make different images fused together, and it is convenient to obtain the effective information of multiple images in a limited map. As a global change, Fourier transform has certain limitations, but its inverse transform is easy to find, and the application of MATLAB makes this kind of image fusion method easier to achieve.

References

1. Fang Kai. Research on Digital Image Fusion Algorithm. Xidian University, 2006.
2. Qin Yunxia. Research on Image Fusion Algorithm. Lanzhou University, 2010.
3. Han Xiyu. Research on infrared and visible image fusion algorithm based on multi-scale and significant area analysis. University of Chinese Academy of Sciences (Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences), 2020.
4. Guo Rui. Research on the Evaluation of Image Fusion Quality // Jilin University. Document Analysis and Recognition (ICDAR). 2009. P. 1340–1344.

UDC 004.912

Su S., Blekanov I.

Neural network methods for messages analysis in user discussions on YouTube

1. Introduction. With the burgeoning development of web-space, it usually takes a few seconds for the news to be widespread. Especially, for some sensitive or widely popular events, they probably draw attention of a large amount of people who create a storm of public opinion online. Meanwhile, for journalists, organizations and relevant parties, it's necessary and obliged to flexibly analyze the user's activities and comprehend the general public opinion over specific topics. YouTube as the most famous and representative social platform, which provides us such possibility. In this paper, authors mainly propose a general solution to user discussion analysis on YouTube. Combined analytic techniques and different approaches including the state-of-the-art models are being applied. In our experiment, the case of American police misconducts in dealing with a black man is taken as our example for analysis. Temporarily our model only supports English language text.

2. Solution of user discussion analysis on YouTube. Our solution contains the following steps: data crawling, data preprocessing, language model for sentiment analysis and summarization.

2.1. Data crawling. In this work, authors constructed two different methods of implementing the data crawling process. The first solution based on Selenium, it works like the element crawler locating the position of data by walking through the hierarchy of webpage and the advantage is quite obvious that it has powerful tentacle to touch a large amount of elements on the webpage. The second approach based on YouTube Data API (Application Programming Interface) with strong robustness and low cost time, it can flexibly access the Google server to request the response which conveys the data. Then the data are saved in database.

Su Shimai – student, Saint Petersburg State University; e-mail: st073379@student.spbu.ru, phone: +7(961)610-92-88

Blekanov Ivan – associate professor, Saint Petersburg State University; e-mail: I.blekanov@spbu.ru, phone: +7(812)328-31-84

2.2. Data preprocessing. Text of user publications in a social network contain many unimportant characters such as links, emojis and in our case, words not in English. It's inevitable to filter all irrelevant letters with regular expression. Procedures of stopword filtering are not used because the sentiment analysis demands the integrity of expression of user opinion. Besides, authors set a changeable window size to split the data into multiple groups to detect the evolvement of the public opinion over time.

2.3. Language model for sentiment analysis. The purpose of the application of language model is to transform English language texts into an embedding vector, which will be utilized as the input to sentiment classification.

Universal Sentence Encoder (USE) [1] is a specific model for encoding sentences into embedding vectors, i.e., the model generates a fixed dimensional embedding representation of the English input string. The vector representation z for input text X is computed by averaging the word vectors [2] $v_{w \in X}$. For each layer,

$$z_i = g(z_{i-1}) = f(W_i z_{i-1} + b_i), \quad i = 1, \dots, n \quad (1)$$

and the softmax layer takes the final layer's representation z_n as input for prediction of one of k labels. And the model is pre-trained by minimizing cross-entropy error, which for a single training instance with ground-truth label y_p is $L(\hat{y}) = \sum_{p=1}^k y_p \log(\hat{y}_p)$, where $\hat{y} = \text{softmax}(z_n)$.

Bidirectional Encoder Representations from Transformers (BERT) [3] is a well-known and state-of-the-art model, which is used for pre-training bidirectional representations from unlabeled text. The model architecture of BERT is implemented on a competitive multi-layer bidirectional Transformer [4] encoder. The attention mechanism of Transformer is sufficient for learning the global dependencies between input and output. In this paper, authors used BERT base (number of Transformer blocks $L=12$, the hidden size $H=768$, number of self-attention heads $A=12$) as our pre-trained model.

Sentiment classification is performed after obtaining the embedding vectors of English texts with original labels. There are various approaches to sentiment classification, like the standard multi-class text classification which uses convolution model. In the paper, authors solved sentiment classification task using deep learning approach based on Tensorflow. Most importantly, it's able to generate the neutral label even

though there is no such label existing in the training dataset and it's operable to make some adjustment to the key parameters (e.g. learning rate, epoch) to obtain the optimal classification performance.

2.4. Summarization. Several techniques are being taken to summarize the main concerns in user discussion better.

Separation mechanism. The truth is the focus of the discussion is probably extended to all possible fields with time changing. In this way, authors designed a separation mechanism to split the users into multiple groups. Moreover, the window size which denotes the number of comments within a period can be modified according to the actual situation. In our case, window size = 100.

Text-to-Text Transfer Transformer (T5) [5] does not emphasize on bringing up a new method but instead explore the broad perspective on the usage. Converse to Bert, T5 is built based on standard encoder-decoder Transformer and it's open for mainly 5 tasks which include translation, question answering, classification, regression and summarization. The way to trigger the model's operation is that adding a task-specific (text) prefix before the original text. In our case, we concentrate on the summarization task (prefix: summarize) which is applied to summarize the main content of user discussions.

3. Experiment. The experiment has two parts. Firstly, labeled datasets are being used to train and evaluate the model. The second part is the application of trained model on real dataset which is crawled from the actual event: A black man was killed by American police last year.

3.1. Dataset. Our real dataset (YouTube video id=XsgryiPK2is) contains 4583 items. For each item, it includes username, level of comment, comment, publish time and etc. And our training dataset with a label (0: negative, 1: positive) is downloaded from Twitter sentiment analysis training corpus dataset (Year 2012). Authors split 80% as the training dataset, 20% as the testing dataset to evaluate the performance of different models.

3.2. Model comparison. Through the adjustment of several parameters, we obtain the following performance under the current setting. As we can notice that the accuracy of both models approaches 0.8 which demonstrates good fit to the testing dataset. Therefore the trained models are then applied to predict the our own unlabeled dataset.

Table 1. Performance of USE and BERT

Model	Epoch	Acc	Rate	Speed
USE	20	0.7547	0.0005	Around 0.7h
BERT	20	0.7676	0.0005	Around 3.5h

3.3. Real case result. The generated label of our real dataset is acquired based on the trained model. Authors analyzed the result from two perspectives, the evolution of public sentiment and the paramount concerns in user discussion.

Public sentiment evolution. Fig. 1 shows that under the two models, how the public sentiment evolves with respect to different groups sorted by time. Authors found out that basically the overall sentiment to topic “Black man was killed by American police” is negative and remains high level over time. The most distinguishable difference between these two models is the trend of positive level. Compared with figure on the right, the left positive trend is located approaching negative, since the accuracy of both models is close, it’s quite necessary to pay more attention to such occurrence which may indicate the existence of racist.

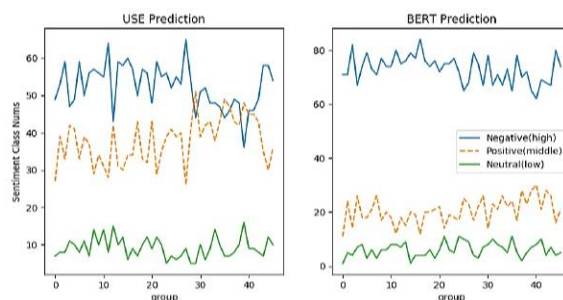


Figure 1. Public sentiment evolution

Summarization of the public opinion. Based on 46 groups, authors summarized the core content for each group and aggregate them all. As shown in Fig. 2, the size of words represent the level of importance. Larger police and cop, small black man, which distinctly tells us that the public’s attention is mainly on the police instead of black man. John Avlon is a senior political analyst and anchor. Lz Granderson is also an American journalist. Both of them are playing a certain role in the discussion. Then gun, shoot and racist turn as the next priority.



Figure 2. Public opinion

4. Conclusion. Our experiment adequately discovers the overall sentiment polarity and the main topics of public opinion by using various models and techniques. However, authors still need to improve the accuracy of models and take a deep research on how people can recognize the dangerous discussion through our model.

References

1. Daniel C. et al. Universal sentence encoder // Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2018. P. 169–174.
2. Mohit I. et al. Deep unordered composition rivals syntactic methods for text classification // Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. 2015. Vol. 1. P. 1681–1691.
3. Matthew E. P. et al. BERT: Pre-training of deep bidirectional transformers for language understanding // Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2019. Vol. 1. P. 4171–4186.
4. Ashish V. et al. Attention is all you need // Advances in Neural Information Processing Systems. 2017. P. 6000–6010.
5. Colin R. et al. Exploring the limits of transfer learning with a unified text-to-text transformer // Journal Of Machine Learning Research. 2020. Vol. 21. No 140. P. 1–67.

UDC 004

Tang Z., Pyshkin E.

Managing notifications within the context of soft planning: visual approach to requirement analysis

Abstract

In this contribution, we discuss an approach to manage the scenarios of soft planning conceptualized with the help of common-sense ontology models. Specifically, we investigate the benefits and possible drawback of managing mobile notifications processed on the base of deferred plan reminders that may be programmed by the user. We analyze the possible requirements to such a system and define major scenarios represented with the use of standard behavioural models such as statecharts. High degree of visualization is used in order to make a stress on the processes of system design modeling, as well as on representing the project conceptual core with the use of graph-based knowledge ontology, as well as through structural and behavioral models.

1. Introduction. Support for different types of planning is one of many aspects of human life digitalization. Most computer-assisted planning systems cover the scenarios, when individual or team activities may be scheduled under the variety of constraints such as time, budget, location, and resources. There are many existing solutions for time and organizational planning or project scheduling [1], event-based activity planning [2], and travel planning [3, 4], to cite a few. In a sense, all these solutions are about what we could call “determined planning” when the planning goals, time range, locations, and activities are mostly known in advance. At the same time, there is a growing interest to advancing features of personalized recommending systems using social network data and personal user profiles linked to location data [5, 6, 7]. In our earlier work we introduced a concept of *soft planning* describing personalized

Tang Ziqiang – student, Graduate School of Computer Science and Engineering, The University of Aizu; e-mail: tziqiang410@gmail.com, phone: +81-242-37-2603

Pyshkin Evgeny – senior associate professor, The University of Aizu; e-mail: pyshe@u-aizu.ac.jp., phone: +81-242-37-2603

The work was performed with financial support of The University of Aizu research funding

situational planning scenarios, which are (1) not particularly certain, (2) may be activated when some suitable but probably unplanned situation takes place, and (3) in principle, might not be implemented at all [8]. Indeed, in real life, one can decide eventually based on the current situation, and/or past positive or negative experience. For example, while traveling, even if the major trip plans are fixed, the process of selecting interesting spots accessible on the way is not completely trivial. For example, an approach suggested in [9], extends the case of self-planning scenarios by adding the points of interests, which were not planned in advance using the expansion of the areas along the trip so that to recommend interesting spots based on their popularity. Basic algorithms are based on calculating distances, though the decision on whether a particular distanced point is reachable may depend on many other factors such as available means of transportation, opening hours, weather conditions, type of surrounding landscape, etc., as it is demonstrated in [10]. Moreover, the plans to visit a particular spot might be shifted in time, and might not be implemented exactly in the same moment. Here are some example scenarios illustrating the concept of soft planning:

1. The traveler discovers an interesting location which is worth to be visited, but which is not in the best season right now (such as a waterfall, which is in a dry period, or cherry trees, which are not blossoming, etc.). Thus, the traveler does not exactly know a best time frame for going there. A reminder could describe a deferred plan, which may be activated, as soon as it happens that the traveler is nearby this location in better season.
2. The theater enthusiast would like to watch a particular streamed performance as soon as the streaming video is available on the web (but the time is not exactly known right now).
3. One may realize that there's something missing at home (such as particular food products or household items). The purchase might not be urgently required, but a notification reminding about it while we are in some relevant shop, may be helpful.

2. Related Work. Support for notifications is one of the key features of smart devices we use today [11]: smart device notifications are no longer just situational alerts, they become an integral proactive component of mobile services engaging with their users [12]. Notifications give the common approach to link the current context to a situation

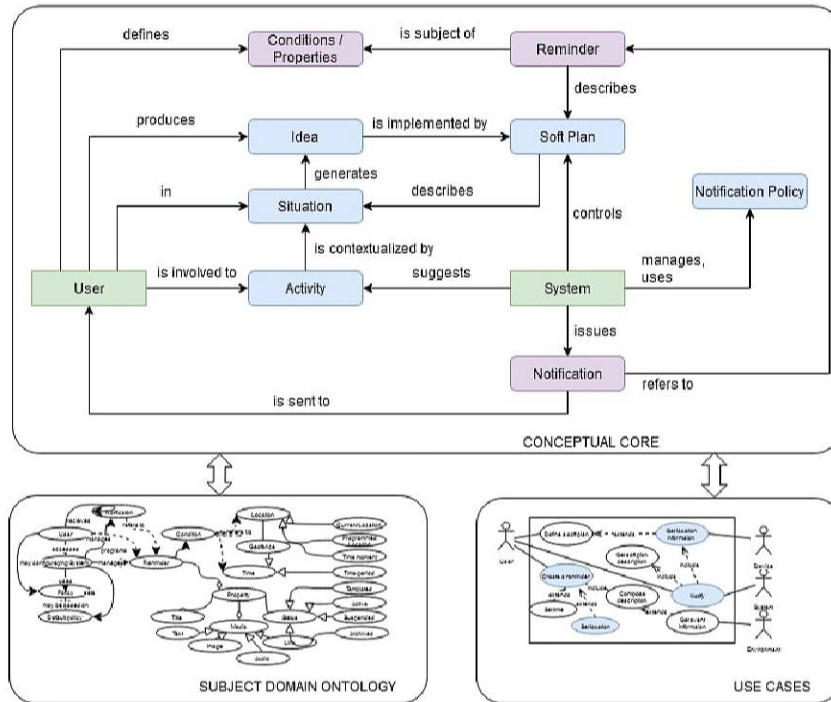


Figure 1. Soft planning system: common-sense ontology

which might be distant in time or/and location and, in a sense, to program a deferred action so that to let the user adapting her behavior according to the previewed condition (situation). With a software for notification management applied to the context of soft planning, the user may expect the features supporting “intelligent programming” of reminders, which would lead to notifying users about the possibility to implement a deferred activity as soon as a suitable situation happens. Using programmed reminders together With a reasoning engine may be helpful for getting some more information from the user notes that would be related to deferred activity.

In [8] we conceptualized the soft planning model in the form of common-sense ontology providing foundations for designing an information system architecture as it is demonstrated in [13]. Compared to

our early design, the diagram in Figure 1 shows a revised model introducing major concepts, which are necessary to understand how soft planning scenarios appear. As a conceptual core of the whole project, the common-sense ontology is linked to domain-specific ontologies and expected scenarios to be implemented in the system. Figure 2 gives some more insights to our understanding of the links between the soft and the major properties that have to be taken into the consideration while implementing support for managing user-defined reminders and related notifications.

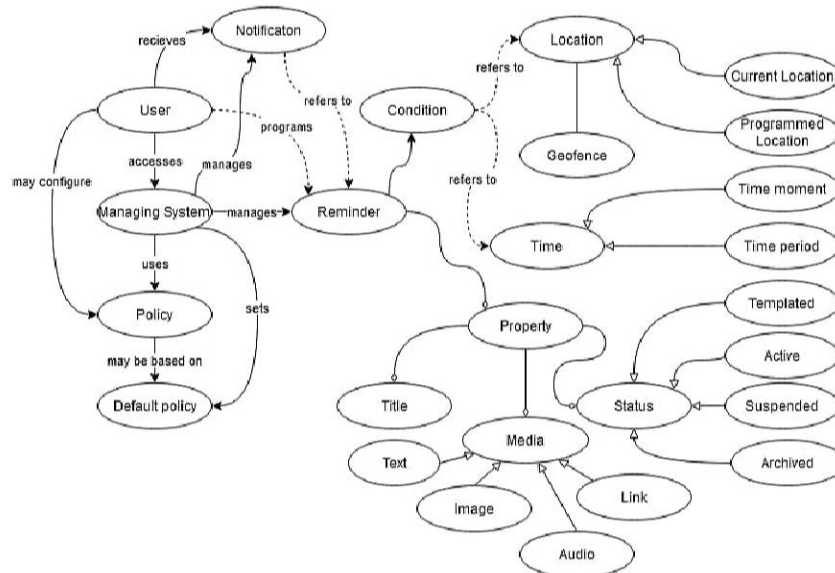


Figure 2. Soft planning reminders and notifications

Thus, the concept of soft planning presumably connects our fixed plans to situations, when we may have less strict schedule containing the activities, which are not planned in advance, but may (or may not!) be implemented as soon as suitable conditions take place. The scenarios of soft planning may be considered as an element of activity centred computing paradigm addressing the multi-faceted nature of human activity in the real world, when people tend to create *“activity representations that are simple, yet flexible enough to accommodate different levels*

of rigidity” [14] and “re-program” their future experience based on the feedback they may have from the current activity, or from the experience of other travelers.

3. Managing Reminders: Behavioral Models. According to the model shown in Figure 2, the reminders set by the user, may belong to four principal classes:

1. *Active reminders* are reminders configured by the user and being processed by the system so that when the appropriate user conditions (such as being nearby the desired location) are fulfilled, the system issues a notification.
2. *Templates* are reminders stored in the system for their future use. These reminders may be configured partially, or suggested by the system for common scenarios.
3. *Suspended reminders* are reminders that were delayed by the user. For suspended reminders, the user may need specific configuration options, which may refer to the earlier established policies (such as reactivate the reminder after certain period of time, reactivate it manually, or even remind the user about the existing suspended actions).
4. *Archived reminders* may be useful for the cases if the user wishes to save the previously configured reminders for future re-activation.

Based on these classes, we defined two major stages of reminder and notification management using statechart behavioral models:

1. System superstate (according to terminology from Harel’s statechart visual formalism [15]) “Setting and configuring a reminder” (Figure 3) describes major events and actions necessary to support the processes of creating a reminder (from scratch or based on the existing templates), working with the lists of active and archived reminders, saving and deleting reminders.
2. System superstate “Working with issued notifications” (Figure 4) describes major events and actions that may be necessary at the moment, when a notification is issued. After opening the reminder referred by the issued notification, the user should be able to “forget” this reminder, archive it, keep active, or re-configure it (thus, the system may trigger to the superstate “Setting and configuring a reminder”).

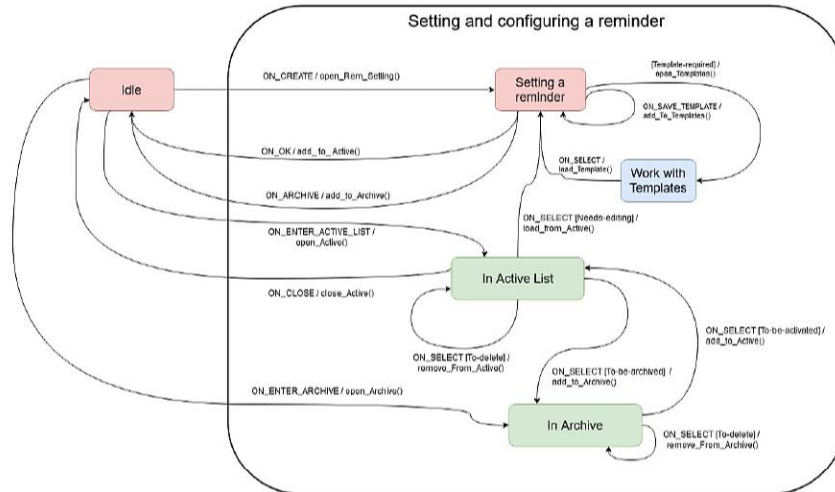


Figure 3. Setting a reminder

There are usability aspects that have to be taken into the consideration: specifically, how frequently the system issues the notifications, and what is the optimal degree of sensitivity for renewing the archived notification. There may be also recurring reminders, which may be programmed to be used several times depending on the context and current conditions. Finally, to decide how to react, the system might need to use different policies (both default and defined by the user). Such policies may be useful for implementing the system behavior in specific conditions, for example: what to do after issuing a notification, what is a time span between recurring reminders, etc.

4. Early Designs and Use Case Analysis. Based on the conceptual model that we introduced and Google Maps API, we designed a prototype Android application partially implementing the earlier defined scenarios. This software is aimed at rediscovering possible drawbacks of suggested models and at investigating directions for future work. This application partially implements major possible scenarios of soft planning, as Figure 5 illustrates (greyed use cases correspond to the features we tried to implement in this prototype). We use this research version as an experimental model aimed at refining further functional and usability

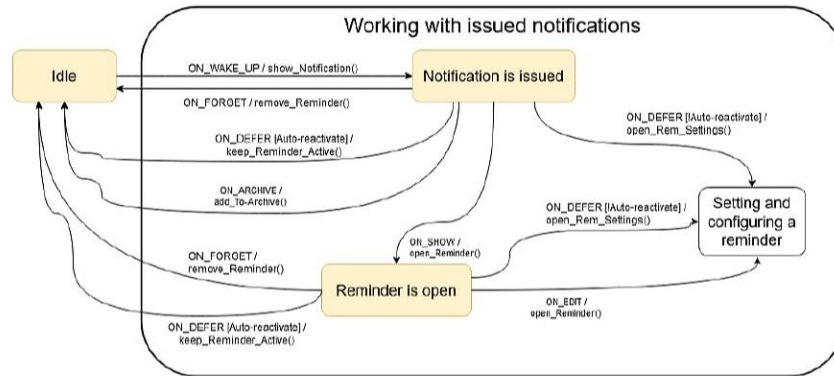


Figure 4. Work with issued notifications

requirements.

Specifically, we investigated the problem of using the geofence [16] target area, which depends on the location set by the user for a particular reminder. Theoretically, when a person enters this area, the application should display a notification suggesting the deferred activity connected to the above mentioned location, and asks the person about the desired action (accept and remove, accept and keep active, ignore (and keep active as a default policy), decline and remove, delay, or archive).

Geofence is an area around some location, when the latter is considered to be easily reachable if the user enters the geofence. However, to generate geofence-based reliable suggestions, it is not enough to using the geometric distancing only. We need to pay attention to the means of transportation (geofence for a car driver may be surely larger compared to the bike riders or walkers), landscape characteristics (to walk in the mountains is harder than in the city, unless the latter is San-Francisco), or to the conditions that may change on the way (there may be waiting time for visiting some attraction, or the working hours may be our of time necessary to go to the desired destination, for example).

5. Discussion. This research on developing mobile services based on the concept of situation-based planning is still in the stage of early designs. In addition to the concept introduced with the help of ontology models in [8], in the current work we made an effort to visually describe the requirements to constructing software tools that would pro-

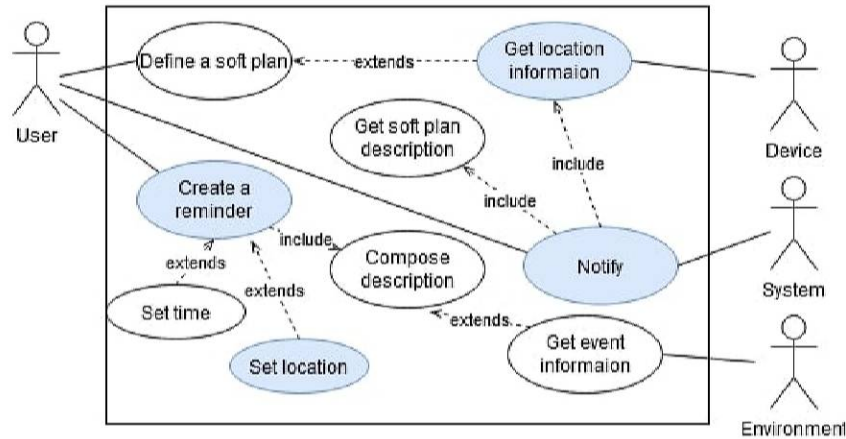


Figure 5. Major use cases in the soft planning system

vide services for the above discussed scenarios. Our experiments with the existing intelligent assistants and our prototype application show that in order to know the fact that the desired "reminded" location is in the proximity of current user location (or within the corresponding geofence) might not be enough to decide, whether the related notification should be issued. Careful analysis is required to define reasonable default notification management policies that would simplify the process of configuring a reminder and assure minimizing annoying unnecessary notifications and preventing of never-ending reminders. We could also mention the following problems among the open issues:

- Identification of possible useful scenarios of user collaboration in setting and using "soft" plans;
- Integration of soft planners with regular time schedulers and activity organizers;
- Possibilities to apply a reasoning engine, so that to get some more information related to the deferred activity from the context including the task of extracting location or time descriptions from the user's notes, with an idea that both location and time may be defined very approximately (such as "some convenience store", "in the cherry blossom period", "late in the evening" etc.).

References

1. Canberra Hyatt, Patrick Weaver. A brief history of scheduling // Melbourne, Australia: Mosaic Project Services Pty Ltd. 2006.
2. Kun-Ming Yu, Jiayi Zhou, Cheng-Yan Yu, Jian-Yuan Liu., Chi-Chung Lee, Hsin-Wen Chang, Hung-Nien Hsieh. An event-based wireless navigation and healthcare system for group recreational cycling // 2009 Fifth International Conference on Mobile Ad-hoc and Sensor Networks. IEEE, 2009. P. 294–298.
3. Igo Ramalho Brilhante, Jose Antonio Macedo, Franco Maria Nardini, Raffaele Perego, Chiara Renso. On planning sightseeing tours with tripbuilder // Information Processing & Management. 2015. Vol. 51. No 2. P. 1–15.
4. Shkolnikov F., Kuznetsov A., Pyshkin E. Merging tourist routes for collaborative trip planning // Proc. of the 3rd Int. Conf. on Applications in Information Technology (ICAIT). 2018. P. 35–40.
5. Jie Bao, Yu Zheng, David Wilkie, Mohamed Mokbel. Recommendations in location-based social networks: a survey // GeoInformatica. 2015. Vol. 19. No 3. P. 525–565.
6. Cenamor Isabel, de la Rosa Tomás, Núñez Sergio, Borrajo Daniel. Planning for tourism routes using social networks // Expert Systems with Applications. 2017. Vol. 69. P. 1–9.
7. Logesh R., Subramaniaswamy V., Vijayakumar V. A personalised travel recommender system utilising social network profile and accurate GPS data // Electronic Government, an International Journal. 2018. Vol. 14, No 1. P. 90–113.
8. Tang Z., Pyshkin E. Ontological approach to personalized situational planning: Concept and scenarios // 2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS). IEEE, 2019. P. 561–564
9. Chu Chung-Hua, Huang Chenyang. A platform for travel planning by using google maps // 2015 16th IEEE International Conference on Mobile Data Management. IEEE, 2015. Vol. 2. P. 120–125.

10. Kuznetsov A., Pyshkin E. Approach to develop an assistant application for controlling trace accuracy in travel timelines // Proc. of the 8th International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services (CENTRIC 2018). 2018. P. 14–18.
11. Dominik Weber, Alexandra Voit, Jonas Auda, Stefan Schneegass, Niels Henze. Snooze! investigating the user-defined deferral of mobile notifications // Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services. 2018. P. 1–13.
12. Martin Pielot, Amalia Vradi, Sounel Park. Dismissed! a detailed exploration of how mobile phone users handle push notifications // Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services. 2018. P. 1–11.
13. Garcia Ander, Arbelaitz Olatz, Linaza Maria Teresa, Vansteenwegen Pieter, Souffriau Wouter. Personalized tourist route generation // International Conference on Web Engineering Springer, 2010. P. 486–497.
14. Bardram J. E., Jeuris S., Tell P., Houben S., Volda S. Activity-centric computing systems // Activity-centric computing systems. Communications of the ACM. 2019. Vol. 62. No 1. P. 72–81.
15. Harel David. Statecharts: A visual approach to complex systems. MCC, 1986.
16. Create and monitor geofences [Internet resource]: URL:<https://developer.android.com/training/location/geofencing.html> (date: 24.01.21).

UDC 004.81

Yang W., Li P.

Research on sentiment analysis of network public opinion based on BI-LSTM

Recommended for publication by associate professor Blekanov I. S.

1. Abstract. Natural language processing is a great help in analyzing public opinion. BI-LSTM can better capture the bidirectional semantic dependence, so it can realize the sentiment analysis of public opinion more efficiently. Public opinion data through Scrapy crawling, can quickly obtain a large number of experimental data. Doc2num can be used for text feature vectorization. Because the training model uses a lot of real data, the robustness and accuracy of the model can be effectively improved.

2. Introduction. With the rapid development of the Internet, the influence of public opinion on society is gradually increasing. The demand of management department for sentiment analysis of public opinion is also increasing gradually. But there isn't a mature way to do it. This paper studies the sentiment analysis of network public opinion and analyzes the current situation of public opinion and its treatment. After the scheme design, and for each step of the detailed design. Finally, the experimental results are obtained.

3. Present situation. With more and more attention paid to online public opinion, more and more research has been done on sentiment analysis of online public opinion. Traditional human labor is too cumbersome and inefficient. In the sentiment analysis model, BI-LSTM can better capture the bidirectional semantic dependence. Therefore, it is very important to study a sentiment analysis model based on BI-LSTM.

4. Overall design scheme. The main requirements of this paper are public opinion information collection, data pretreatment and public opinion analysis. The collection of public opinion information is mainly realized by Scrapy crawlers. After that, the data crawled by crawlers will

Yang Wenfeng – undergraduate student, Yanan University;
e-mail: wenfeng_yang1998@163.com, phone: +8615667530218

Li Pengyi – undergraduate student, Yanan University;
e-mail: PengYi_29@163.com, phone: +8615319468365

be preprocessed to remove dirty data and prepare for word segmentation.

The clean data obtained by preprocessing is vectorized with text feature through Doc2Num. Then the sentiment classification model is trained through the training set to obtain the sentiment classification prediction model based on BI-LSTM. Then, the data obtained by the crawler is imported to process and predict, and the predicted results are finally obtained. The overall design flow chart is shown in Figure 1.

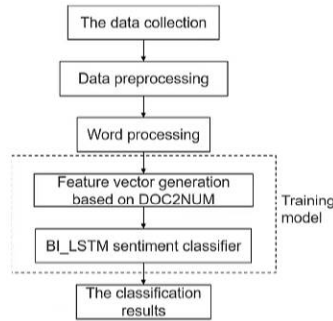


Figure 1. Overall design flow chart

5. Collection of public opinion data. Using Scrapy distributed crawler framework, information crawling is realized through Application Programming Interface (API) of Sina Weibo.

5.1. Establish database as required. According to the business needs, the main information of crawling includes: user ID, user nickname, publication time, gender, province, age, content published by the user, and the website of the published content. The database tables are as follows.

Table 1. Database tables

The field names	Paraphrase
id	User's ID
nick_name	User's nickname
upload_at	Published time
gender	User's gender
province	User's province
age	User's age
content	User's published content
content_url	The URL of the published content

5.2. Demonstration of the process of crawling data. Scrapy is one of the most popular screen Scrapy and Web collection framework, based on Python language development, with fast, concise, high integration features. The Scrapy architecture diagram is shown in Figure 2.

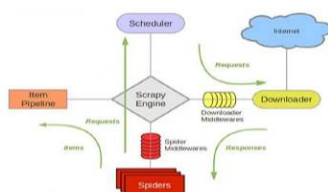


Figure 2. Scrapy architecture diagram

In this paper, a directional crawler is used to initialize the web links and start the Spider module. The links are delivered to the scheduler through the Spider middleware and engine, and the scheduler queues them up. The scheduler passes the link to the queue header through the engine to the downloader for download. Stop crawler when queue is empty. When the downloader receives the connection, it makes a request and crawls the content. When the download is complete, the data is passed to the crawler. The specified data is then extracted using XPath and BeautifulSoup. Selenium can continue extracting after a page of data has been extracted until the preset number of extracted pages has been completed. After successfully fetching the data, the Spider module passes the data through the engine to the Item Pipelines module which stores the data in the database.

6. Experimental data preprocessing.

6.1. Data cleaning. Because there are a lot of useless data such as symbols, links, numbers and letters in the crawling data, it is called noise data here. Noise data is meaningless in data analysis, so it needs to be preprocessed.

The regular expression code is simple, the processing speed is fast, and it can well complete the work of capturing the string, so the method of regular expression is adopted here to extract the data.

6.2. Processing of participles, new words and stops. This paper uses the precise mode of Jieba word segmentation tool for text seg-

mentation. Sentences can be segmented as accurately as possible. When dealing with new words and stop words, you can use the API interface provided by the Jieba word segmentation system to write the hot words and new words and their weight into the file, so that the system can accurately identify the relevant words. Or choose the appropriate stop word dictionary, and the file relative path can be written.

7. Data modeling. After data preprocessing and word segmentation, the subjective text published by users can be analyzed and its attached emotions summarized. In this paper, the bidirectional long and short term memory network (BI-LSTM) in deep machine learning is used to implement the system.

7.1. Text feature vectorization. First, the text data needs to be transformed into text feature vectors. This paper uses DOC2NUM model to vectorize text features. The BI-LSTM model itself already takes into account the semantic relationships of the context, so Doc2Num only needs to number all the words.

Doc2Num will pre-count the word frequency n_x , and then set the word frequency threshold y . For words with n_x less than y , it means that the occurrence of the word is too few and has little influence on the final result, so it can be removed. Then number the remaining words. Finally, replace the words in the text with numbers.

7.2. Sentiment Classification of public opinion. Firstly, the training data is used to train and test the model, and a better sentiment classification model based on BI-LSTM is obtained.

After that, the data crawled by the crawler will be preprocessed, and the preprocessed data will be imported into Doc2Num to generate the word vector model.

Then the word vector is input into the trained classifier to make emotion prediction and the prediction result can be obtained.

Here, two results are printed. One is the positive and negative results of the text prediction. The other is the probability of the accuracy of the prediction. The two results are then stored in the new field in the original database. The new table is shown in Table 2.

Table 2. New Database tables

The field names	Paraphrase
PAN	The positive and negative emotions of the published content
pro	Probability of prediction accuracy

8. Experimental result. After training with a large amount of data, the robustness of the model is gradually enhanced. The reliability of the final analysis results is also increasing. The PAN value can tell you the emotional direction of a public opinion. The accuracy of the judgment result can be known by the Pro value. Therefore, the results of sentiment analysis of public opinion can be obtained through the two values of PAN and pro. Then assist the government management departments to carry out the judgment and management of public opinion orientation.

9. Conclusion. Through training, the emotion classification model is established, and the Scrapy crawler framework, Jieba word segmentation system, Doc2Num model and BI-LSTM model are mainly applied. It mainly realizes the analysis of sentiment classification of public opinion information.

This paper analyzes the correlation between the above framework and model, discusses the specific implementation process and core technology, and finally expounds the results of public opinion analysis. After a large amount of data training, the model can give better analysis results with high accuracy, which provides data support for public opinion management.

References

1. Ma N. Design and Implementation of Online Public Opinion Monitoring System in Colleges and Universities. Nanjing University of Science and Technology. 2018. 60 p.
2. Guan X., Liu X., Liu Z. An algorithm based on simple CNN and BI-LSTM network for Chinese word segmentation // Journal of Physics: Conference Series. 2020. Vol. 1621. Art. No 012001.
3. Dimitris C. Master Python Scrapy Framework. Peoples Posts and Telecommunications Press. 2018. 258. p.

UDC 004.8

Yuda K., Otomo K., Ito R., Wakana I., Tanji R., Kamei S.,
Mozgovoy M.

Behavior similarity between human players and built-in AI characters in universal fighting engine

Abstract

In certain game genres, such as one-vs-one fighting games, AI-controlled opponents are expected to behave in a human-like manner to facilitate higher player engagement. To develop a human-like AI decision making system, it is essential to be able to perform proper evaluation, i.e., to be able to test any given opponent for human-likeness. In this paper, we propose a tool for a fighting game environment Universal Fighting Engine, able to analyze game recordings and assess similarity between individual characters by examining their internal states and their actions. We test our tool on a sample of human players and five skill levels of a built-in AI system. Our results show that human players possess certain “human-like” traits that make their play styles distinguishable from a conventional rule-based AI system.

1. Introduction. Computer games can serve as a good platform to try artificial intelligence (AI), because its actions are clearly visualised and easy to understand; moreover, games have well-defined rules to follow. In 2016, when AlphaGo [2] won a professional Go player Lee

Yuda Kaori – master student, The University of Aizu; e-mail: m5241108@u-aizu.ac.jp, phone: +81-242-37-2664

Otomo Kimitaka – undergraduate student, The University of Aizu; e-mail: s1250047@u-aizu.ac.jp, phone: +81-242-37-2664

Ito Ryoya – undergraduate student, The University of Aizu; e-mail: s1260233@u-aizu.ac.jp, phone: +81-242-37-2664

Wakana Ippo – undergraduate student, The University of Aizu; e-mail: s1260210@u-aizu.ac.jp, phone: +81-242-37-2664

Tanji Riku – undergraduate student, The University of Aizu; e-mail: s1270139@u-aizu.ac.jp, phone: +81-242-37-2664

Kamei Shota – undergraduate student, The University of Aizu; e-mail: s1270112@u-aizu.ac.jp, phone: +81-242-37-2664

Mozgovoy Maxim – associate professor, The University of Aizu; e-mail: mozgovoy@u-aizu.ac.jp, phone: +81-242-37-2664

Sedol, it impressed many people in different industries. AI abilities and potential impacts of this technology were widely discussed afterwards.

The focus of our research is to make a human-like AI system for the game of fighting. Fighting is one of most popular game genres, having numerous hit games in its history, such as Street Fighter, Mortal Kombat or Super Smash Bros. People often enjoy playing with other people as popularity of online games indicates, so developing a human-like AI (in terms of play style) may improve user experience. As a preliminary step of this work, it is necessary to understand what exactly constitutes “human-like behavior” and what kind of features such an AI system have to possess. In this paper we evaluate play styles of various human players and of a generic rule-based AI using an automated scoring algorithm, based on cosine similarity.

2. Universal Fighting Engine. In this study, we use Universal Fighting Engine 2 (UFE) [3], which is a development tool for one-vs-one fighting games in Unity. UFE is highly extensible: for example, we can limit the selection of actions or speed of making decisions for individual characters. It also includes a “Fuzzy AI” for non-player characters (NPC), which is a fine-tunable rule-based AI. It can be set to different difficulty levels by adjusting the parameters corresponding to time between decisions, time between actions, rule compliance, aggressiveness and combo efficiency. We will mainly discuss about differences in human player and Fuzzy AI.

3. Cosine similarity analysis. In order to compare different players’ behavior profiles, we have developed a similarity scoring procedure based on a cosine similarity metrics. It operates as follows. We analyze recordings of games where a character of our interest participates, and create its “behavior fingerprint” as an ordered list of probabilities of every possible tuple (A_1, A_2, A_3) , representing three consecutive player actions. Each action is uniquely defined with its elements `currentState`, `currentSubstate`, and `currentBasicMove` (see Table 1). Behavior fingerprints can be then compared using cosine similarity [4], yielding a similarity ratio of $[0, 1]$.

Table 1. Elements of player actions

Element	Meaning
currentState	Indicates characters state (e.g., Stand, Jump, Down)
currentSubstate	Indicates additional state modifier of the character (e.g., Resting, Blocking)
currentBasicMove	Indicates characters basic movement state (e.g., Idle, MoveForward)

$$\begin{aligned}
\text{Similarity}(A, C) &= \cos(|A|, |C|) \\
&= \frac{A \cdot C}{|A||C|} \\
&= \frac{\sum_{i=1}^n A_i C_i}{\sqrt{\sum_{i=1}^n A_i^2} \cdot \sqrt{\sum_{i=1}^n C_i^2}}
\end{aligned}$$

4. Experimental Results. We test four human players, Ryoya, Kaori, Ippo, and Riku. In addition, we test a non-player character operated by 5 different skill levels of Fuzzy AI. We test 12 different combinations of players. Each game recording consists of 10 matches. Each match takes two rounds of 100 seconds each (unless it ends prematurely with a knockout). We calculate similarity scores for the player pairs and analyze obtained results.

In Figure 1, graph edges show pairs of players participated in our game recordings. Edge labels denote similarity between the player. The minimum similarity detected in our recordings is 0.18, and the maximum similarity is 0.73.

In Figure 2, solid lines show the similarity between Fuzzy AI and a human opponent Ryoya. Dashed lines show the similarity between different levels of Fuzzy AI. While resemblance of Fuzzy AI with Ryoya varies in these recordings, it never achieves scores higher than 0.73. In contrast, the similarity between two Fuzzy AI-based agents is 0.91...0.98 regardless of tunable parameters values.

Figure 3 shows the relationships between four players: Kaori, Ippo, Riku, and Fuzzy-AI (normal difficulty level). It shows 39 possible links for each character out of every combination of game. For example, there are 7 links between Kaori and Ippo, the minimum similarity ratio is 0.57 and the maximum ratio is 0.68.

5. Discussion and Conclusion. We have shown how the behavior of individual players (including those controlled by an AI system) can

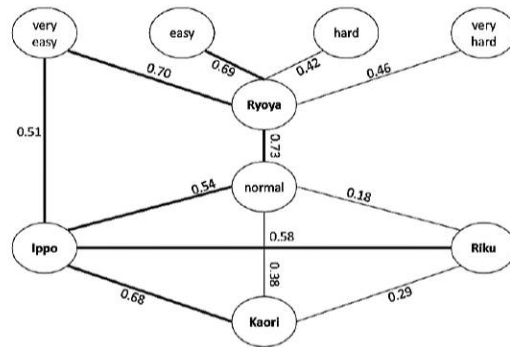


Figure 1. Similarity with match opponent (direct similarity)

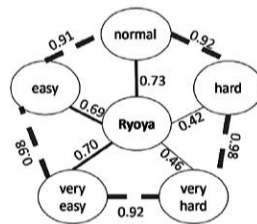


Figure 2. Similarity between Fuzzy AI agents of different skill levels

be evaluated by analyzing their action choice. However, we could not find clear differences between human players and rule-based AI. Still, our observations let us make the following conclusions:

1. A typical rule-based AI system preserves its play style regardless of its difficulty level.
2. People mostly keep their play style even when playing against different opponents.
3. People are diverse, and they are different or similar to other people and/or AI systems, and there is no easy way to tell whether the given player is human or AI.
4. It is possible to compare play styles of the players A and B indirectly by comparing the recordings of matches of A and B against other players.

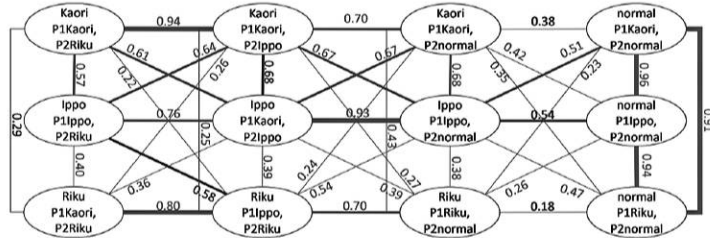


Figure 3. Direct and indirect similarities between the players

It remains unclear whether our inability to separate human players from AI-controlled characters can be explained by the weakness of our method or by objective factors. One of our future goals is to employ a Turing test-like approach to check whether people are able to distinguish human participants from AI players in fighting.

References

1. Mola Bogdan G., Mozgovoy M., Ito T., Rikimaru T. Believability assessment for fighting game AI // Proceedings of Game-On'2017 Conference, Carlow, Ireland, 2017. P. 87–89.
2. Silver D., Huang A., Maddison C. et al. Mastering the game of Go with deep neural networks and tree search // Nature. 2016. Vol. 529. P. 484–489.
3. Universal Fighting Engine [internet resource]:
URL:<http://www.ufe3d.com> (date: 24.01.21).
4. Nguyen H. V., Bai L. Cosine similarity metric learning for face verification // In Asian Conference on Computer Vision, 2010. P. 709–720.

UDC 004.8

Zhao C., Blekanov I.

Two towers collaborative filtering algorithm for movie recommendation

1. Introduction. Recommendation System is a very useful tool for both users and item-makers. For users, recommendation systems can help them to obtain interesting information efficiently. For item-makers, they can use recommendation systems to improve the user stickiness and the conversion rate of users. In this paper, authors mainly focus on representing users and movies as vectors in a better way. Through the Deep Learning technique, by combining the traditional CF(Collaborative Filtering) [1] algorithm with the Matrix Factorization [2] algorithm, authors propose a new model architecture based on the NeuralCF model [3] and Youtube Two Tower model [4], which we call the “Two Tower NeuralCF model”. The model was tested on the Movielens [5] 100k and 1M datasets using measure RMSE.

2. Recommendation system. Recommendation System is a function $f(U, I, C)$, where U represents a user and I represents a large amount of item (movies), C represent a particular context.

The output of function $f(U, I, C)$ is a recommendation list for user U . As we mentioned previously, the performance of function $f(U, I, C)$ depends on how users and items are represented. The procedure of representing an object as a vector is called Embedding. Since Google proposed Word2Vec[6] in 2013, the idea of Embedding has rapidly expanded from natural language processing to all machine learning domains, and recommendation systems are no exception. Next, we will introduce several common object representation method.

2.1. One-hot and multi-hot vector. One-hot vector means a vector only has one element equal to 1, and all others equal to 0. The definition of a multi-hot vector is similar. The disadvantage of one-hot encoding and multi-hot encoding is that the result is very sparse data.

Zhao Chi – master student, Saint Petersburg State University; e-mail: dandanv5@hotmail.com, phone: +7(812)328-31-84

Blekanov Ivan – associate professor, Saint Petersburg State University; e-mail: I.blekanov@spbu.ru, phone: +7(812)328-31-84

And deep learning technique is very unsuitable for dealing with sparse data.

2.2. Collaborative filtering. Collaborative filtering method is simple to implement, which is based on the generation of co-occurrence matrix $R = \{r_{ij}\} \in \mathbb{R}^{m \times n}$ of users and items. Where m, n are the number of users and items, respectively. Then use rows of matrix R represent users, and columns of matrix R represent items. r_{ij} means the rating of item j given by user i . We can not ensure that the matrix R is a square matrix, so we cannot directly calculate users' similarity to items. Moreover, in most cases, the matrix R is a sparse matrix. We can not use the deep learning technique.

2.3. Matrix factorization. This method considers the co-occurrence matrix R as the product of two lower-rank matrices, i. e.

$$R \approx \hat{R} = PQ^T. \quad (1)$$

Where P, Q are latent matrices of users and items (movies) respectively. The dot product of all the row vectors in these two matrices is the estimate of r_{ij} , $i = 1, \dots, m$ and $j = 1, \dots, n$. The object of matrix factorization algorithm is learning of these two matrices.

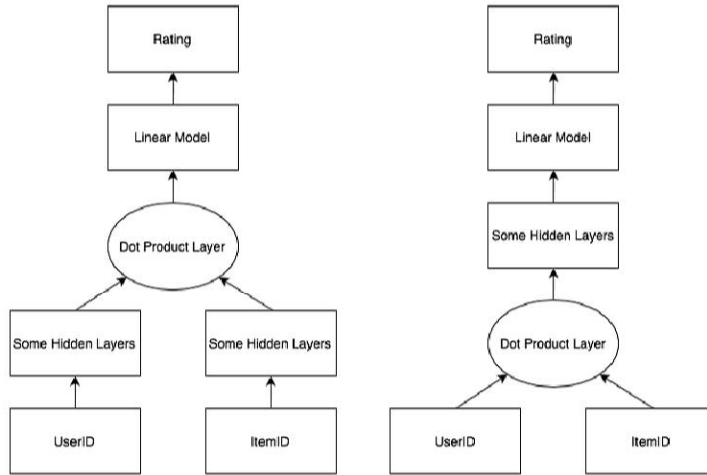


Figure 1. Logical Architecture of Two Tower NeuralCF and NeuralCF Model

The NeuralCF model adds some hidden layers after the dot product

layer based on the collaborative filtering and matrix factorization algorithm. And Youtube two-tower model gives us the inspiration to deal with users and items separately. The difference between the Two Tower NeuralCF model and NeuralCF model is the order of hidden layers and the dot product layer. Figure 1 shows the difference between Two Tower NeuralCF Model (left) and NeuralCF Model (right).

3. Proposed solution. This part introduces the Two Tower NeuralCF model details, including the objective function, back-propagation and forward-propagation.

3.1. Objective function and backward-propagation. Denote $\{x_i\}_{i=1}^M$ and $\{y_j\}_{j=1}^N$ as the set of users and items, where $x_i \in \mathcal{X}$ and $y_j \in \mathcal{Y}$ are both mixtures feature vector of user and item (i.e. sparse features and dense features, could be in a very high dimensional space). \mathcal{X} and \mathcal{Y} are vector space in their corresponding dimension.

We aim to build a model with two parameterized Embedding functions $u : \mathcal{X} \mapsto \mathbb{R}^k$ and $v : \mathcal{Y} \mapsto \mathbb{R}^k$. In this article, the default input vectors are all column vectors. Then, for users, input embedding matrix is W_u and output embedding matrix is W'_u . We denote input and out embedding matrices of items as W_i and W'_i respectively. The layer before the output is the inner product of two embeddings. Which is:

$$s(x, y) = \left\langle u(x, W_u, W'_u), v(y, W_i, W'_i) \right\rangle. \quad (2)$$

The output of the model is:

$$r(x, y) = ws(x, y) + b. \quad (3)$$

The goal of our model is to learn these four matrices W_u, W'_u, W_i and W'_i from a training dataset of T examples, denote by:

$$\mathcal{T} := \{(x_i, y_i, r_i)\}_{i=1}^T, \quad (4)$$

where (x_i, y_i) is the pair of user x_i and movie y_i , and $r_i \in \mathbb{R}$ is the corresponding interactive behaviour for each pair. In this work, we consider this problem as a regression problem, and the target is to predict the “rating”. We use MSE (mean square error) as our loss function.

$$L_T(W_u, W'_u, W_i, W'_i, w, b) = \frac{1}{T} \sum_{i \in [T]} (\hat{r}_i - r)^2. \quad (5)$$

Apply optimization algorithm on loss function (5) to update the parameters. In our model, we use Adam optimizer [7]. This step is called back-propagation. Next, we will introduce forward-propagation.

4. Forward-propagation. As figure 1(left) shown, we used the same architecture of hidden layers in each side. The procedure of forward-propagation is following:

1. Initialize parameters randomly.

2. For user tower:

$$\begin{aligned} h_{u1} &= W_u^T x + b_{u1}, \\ h_{u1} &= \text{relu}(h_{u1}), \\ h_{u2} &= W'_u{}^T h_{u1} + b_{u2}, \\ h_{u2} &= \text{relu}(h_{u2}), \\ u &= \text{dropout}(p, h_{u2}). \end{aligned} \quad (6)$$

3. For item tower:

$$\begin{aligned} h_{i1} &= W_i^T x + b_{i1}, \\ h_{i1} &= \text{relu}(h_{i1}), \\ h_{i2} &= W'_i{}^T h_{i1} + b_{i2}, \\ h_{i2} &= \text{relu}(h_{i2}), \\ i &= \text{dropout}(p, h_{i2}). \end{aligned} \quad (7)$$

4. Now, we have user embedding vector u and item embedding vector i , in interactive layer do a dot product operation on them:

$$\begin{aligned} s &= u \cdot i, \\ r &= ws + b, \\ r &= \text{relu}(r). \end{aligned} \quad (8)$$

5. Evaluation of Two Tower NeuralCF model. We used RMSE (root mean square error) to evaluate our model. Table 1 shows that the Two Tower NeuralCF model's convergence speed is faster and has a lower

RMSE than the NeuralCF model.

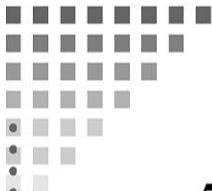
Table 1. RMSE of Two Models

Dataset\Model	NeuralCF	Two Tower NeuralCF
Movielens 100k(Epoch=10)	1.0866	0.9360
Movielens 100k(Epoch=100)	0.9649	0.9399
Movielens 1M(Epoch=10)	1.0602	0.8841
Movielens 1M(Epoch=100)	1.0414	0.8660

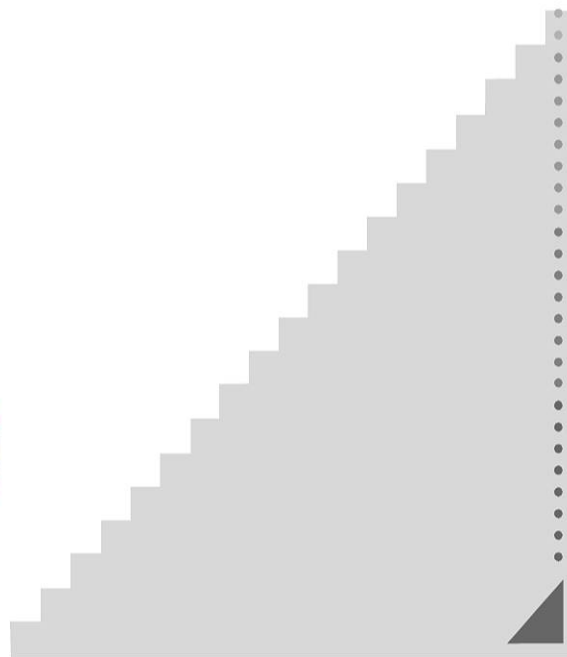
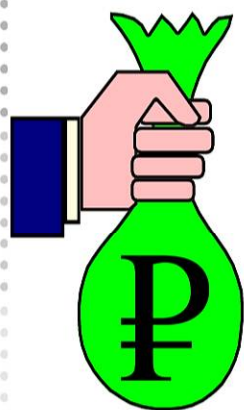
6. Conclusion and Future work. We have proposed a new architecture to find the user’s and item’s Embedding matrix. The final step in the recommendation system is to generate a recommendation list. Therefore, in the next step, we will complete our movie recommendation system based on this approach.

References

1. Goldberg D., Nichols D., Oki B. M. et al. Using collaborative filtering to weave an information tapestry // Communications of the ACM. 1992. Vol. 35. No 12. P. 61–70.
2. Koren Y., Bell R., Volinsky C. Matrix factorization techniques for recommender systems // Computer. 2009. Vol. 42. No 8. P. 30–37.
3. He Xiangnan, et al. Neural collaborative filtering // Proceedings of the 26th international conference on world wide web. 2017. P. 173–182.
4. Yi Xinyang, et al. Sampling-bias-corrected neural modeling for large corpus item recommendations // Proceedings of the 13th ACM Conference on Recommender Systems. 2019. P. 269–277.
5. Harper F. M., Konstan J. A. The movielens datasets: History and context // Acm transactions on interactive intelligent systems (TIIS). 2015. Vol. 5. No 4. P. 1–19.
6. Mikolov T., Chen K., Corrado G. et al. Efficient estimation of word representations in vector space // Proceedings of the International Conference on Learning Representations (ICLR). 2013, arXiv:1301.3781.
7. Kingma D., Ba J. Adam: A method for stochastic optimization // Proceedings of the International Conference on Learning Representations (ICLR). 2015, arXiv:1412.6980.



*5. Управление
социально-
экономическими
системами*



УДК 519.83

Берестюкова В. М.

Байесовское равновесие в дуополии Бертрана

Рекомендовано к публикации доцентом Седаковым А.А.

1. Введение. Рассмотрим класс игр с неполной информацией – байесовские игры. Изучаем случай, когда игроки делают ходы одновременно и независимо друг от друга. К фактическим участникам игры добавляется виртуальный игрок «природа». Природа наделяет каждого из игроков случайной переменной, значения которой называются «типами». Вероятностное распределение типов для каждого из игроков известно. В начале игры природа выбирает типы игроков. Тип, в частности, определяет функцию выигрыша участника. Таким образом, неполнота информации в байесовской игре – незнание игроком типа конкурента.

Рассмотрим модель дуополии Бертрана как байесовскую игру, в которой обе фирмы устанавливают цены на свою продукцию, а объемы продаж определяются с использованием функции рыночного спроса [1]. Неполнота информации в данной модели заключается в том, что игроки не знают точные удельные затраты конкурента, а владеют информацией только об их распределении. Задача заключается в нахождении такого поведения игроков, которое позволит игроку ориентироваться в такой неполноте информации.

2. Постановка задачи. Рассмотрим дуополию Бертрана – ситуацию, при которой имеются только два игрока, не связанных между собой монополистическим соглашением о ценах, рынках сбыта, квотах [2].

Для каждого игрока i задается конечное множество его типов T_i , которое представляет собой удельные затраты. Изучим модель, в которой первый и второй игроки имеют m -мерный и n -мерный наборы типов удельных затрат соответственно:

- $t_1 \in T_1 = \{c_1^1, \dots, c_1^m\}$ – набор удельных затрат первого игрока;

Берестюкова Валерия Михайловна – студент, Санкт-Петербургский государственный университет; e-mail: st062284@student.spbu.ru, тел.: +7(988)897-17-07

- $t_2 \in T_2 = \{c_2^1, \dots, c_2^n\}$ – набор удельных затрат второго игрока.

Кроме этого, задается одно вероятностное распределение P на множестве типов $T = \Pi_i T_i$ и дополнительно $P(t_i) = \sum_{t_{-i}} P(t_i; t_{-i}) > 0$ для каждого t_i . Зная распределение P , можно найти условные вероятности $P(t_{-i}|t_i) = P(t_i; t_{-i})/P(t_i)$. Также задаются состояния природы, каждое из которых определяет игру в нормальной форме. В этой игре множество действий игрока зависит только от его типа и не зависит от типов других игроков.

Игра происходит следующим образом:

1. Природа сообщает игроку только его тип. Игроки не владеют информацией о типе своего соперника.
2. Игроки одновременно и независимо друг от друга выбирают действия, допустимые для своего типа, $p_i(t_i)$, $i = 1, 2$.
3. В конечном итоге игроки получают выигрыши, которые формируются следующим образом:

$$q_i(t_i, t_j) = D_i(p_i(t_i), p_j(t_j)) = 1 - bp_i(t_i) + dp_j(t_j),$$

$$\Pi_i(p_i(t_i), p_j(t_j)) = (p_i(t_i) - c_i)(1 - bp_i(t_i) + dp_j(t_j)).$$

где Π_i – прибыль i -го игрока, q_i – спрос на товар i -го игрока, p_i – цена i -го игрока, p_j – цена j -го игрока, D_i – функция рыночного спроса i -го игрока, c_i – удельные затраты i -го игрока, b и d – зависимости изменения спроса от изменения цены ($0 < d \leq b$): если i -й игрок повысит цену p_i , то его спрос q_i понизится; если j -й игрок повысит свою цену, то спрос i -го игрока q_i увеличится.

Представление i -го игрока статистической байесовской игры в нормальной форме определяется множеством стратегий \mathbb{R}_+ , множеством типов T_1, T_2 , вероятностными распределениями P_1, P_2 и функциями выигрыша Π_1, Π_2 . Тип $t_i \in T_i$ i -го игрока, известный только игроку i , определяет функцию выигрыша $\Pi_i(p_1, p_2; t_1, t_2)$. Обозначим игру как $G = \{\mathbb{R}_+, \mathbb{R}_+; T_1, T_2; P_1, P_2; \Pi_1, \Pi_2\}$.

В статистической байесовской игре G стратегией i -го игрока является функция $p_i(t_i) : T_i \rightarrow \mathbb{R}_+$ такая, что для каждого типа $t_i \in T_i$ $p_i(t_i)$ определяет действие из \mathbb{R}_+ , выбранное заданным природой ти-

пом t_i . Стратегиями игроков в изучаемой модели являются цены p_1, p_2 . Стратегии для первого игрока $p_1 = \{p_1(t)\}, t \in T_1$, а для второго $p_2 = \{p_2(t)\}, t \in T_2$. Набором стратегий игроков будет $p = (p_1, p_2)$.

В качестве выигрыша будем понимать математическое ожидание прибыли Π_i , которое определяется следующим образом [2]:

$$u_i(p_i, p_j; t_i) = \sum_{t_j} P_i(t_j|t_i)\Pi_i(p_i(t_i), p_j^*(t_j)).$$

В качестве решения будем использовать концепцию байесовского равновесия по Нэшу [3]:

3. Байесовское равновесие. В статистической байесовской игре G стратегии $p^* = (p_1^*, p_2^*)$ являются байесовским равновесием по Нэшу, если для каждого игрока i и каждого его типа $t_i \in T_i$ действие

$$p_i^*(t_i) = \operatorname{argmax}_{p_i(t_i) \in \mathbb{R}_+} \sum_{t_j} P_i(t_j|t_i)\Pi_i(p_i(t_i), p_j^*(t_j)),$$

т. е. ни один игрок не хочет менять свою стратегию p_i^* на другую, предполагая, что его противник придерживается выбранной стратегии p_j^* .

В рассматриваемой модели Бертрана байесовское равновесие по Нэшу будет формировать такой набор стратегий:

$$p_1^*(t_1) \in \operatorname{argmax}_{p_1} \left[\sum_{t_2} (p_1 - t_1)(1 - bp_1 + dp_2(t_2))P(t_2|t_1) \right] = \tag{1}$$

$$= \operatorname{argmax}_{p_1} \left[\sum_{i=1}^n P(c_2^i|t_1)(p_1 - t_1)(1 - bp_1 + dp_2^i) \right],$$

$$p_2^*(t_2) \in \operatorname{argmax}_{p_2} \left[\sum_{t_1} (p_2 - t_2)(1 - bp_2 + dp_1(t_1))P(t_1|t_2) \right] = \tag{2}$$

$$= \operatorname{argmax}_{p_2} \left[\sum_{j=1}^m P(c_1^j|t_2)(p_2 - t_2)(1 - bp_2 + dp_1^j) \right],$$

$$p_1^j = p_1(c_1^j), j \in \{1, \dots, m\} \text{ и } p_2^i = p_2(c_2^i), i \in \{1, \dots, n\}.$$

Из (1) и (2) можно получить наилучшие ответы игроков. Разре-

шая (1) и (2) по p_i и p_j соответственно, находим стратегии игроков:

$$p_1^j = \frac{1}{2b} \left(1 + bc_1^j + d \sum_{i=1}^n p_2^i P(c_2^i | c_1^j) \right), \quad (3)$$

$$p_2^i = \frac{1}{2b} \left(1 + bc_2^i + d \sum_{j=1}^m p_1^j P(c_1^j | c_2^i) \right). \quad (4)$$

Таким образом, представления (3) и (4) – это наборы цен, образующие байесовское равновесие по Нэшу, которые могут установить игроки, имея данные наборы типов издержек.

4. Анализ на чувствительность. Рассмотрим пример с заранее известными условными вероятностями и удельными затратами:

$$t_1 \in T_1 = \{c_1^L, c_1^H\} = \left\{ \frac{4}{5}, \frac{6}{5} \right\}, \quad t_2 \in T_2 = \{c_2^L, c_2^H\} = \left\{ \frac{3}{4}, \frac{5}{4} \right\},$$

$$P(c_1^L | c_2^L) = \frac{5}{6}, \quad P(c_1^H | c_2^L) = \frac{1}{6}, \quad P(c_1^L | c_2^H) = \frac{2}{3}, \quad P(c_1^H | c_2^H) = \frac{1}{3},$$

$$P(c_2^L | c_1^L) = \frac{1}{2}, \quad P(c_2^H | c_1^L) = \frac{1}{2}, \quad P(c_2^L | c_1^H) = \frac{3}{4}, \quad P(c_2^H | c_1^H) = \frac{1}{4}.$$

Применяя формулы (3) и (4), получаем равновесные цены:

$$x = \begin{pmatrix} p_1^L \\ p_1^H \\ p_2^L \\ p_2^H \end{pmatrix}.$$

Таким образом, функции цен монотонно убывают при увеличении параметра b . Рассмотрим поведение функции спроса $q_1(p_1^L, p_2^L) = 0,0051b + 1$ для ситуации (p_1^L, p_2^L) . Очевидно, что функция спроса растет на всем промежутке $(0, +\infty)$ при любом значении $b > 0$. Проверив остальные пары стратегий, получаем, что при (p_1^L, p_2^H) , (p_1^H, p_2^H) функция спроса растет с увеличением b , а при (p_1^H, p_2^L) убывает. Изучив функцию прибыли, получаем во всех случаях отрицательные значения производной. Это говорит о том, что функции при-

были при каждой паре стратегий убывает при увеличении b .

Проанализировав поведение функций цен, спроса и прибыли, получаем, что для достижения наибольшей прибыли i -му игроку выгодно, чтобы значение b было как можно ближе к нулю. При увеличении параметра b показатели прибыли значительно ухудшаются.

5. Заключение. В работе рассмотрена дуополия Бертрана как байесовская игра, т. е. игра, в которой игроки имеют неполную информацию о других игроках. Данная модель является игрой между двумя конкурирующими фирмами с различными наборами типов удельных затрат. Используя определение байесовского равновесия, получена система наилучших ответов, т. е. равновесные цены каждой фирмы при заданных стратегиях конкурента. Также получена зависимость функций спроса, цен и прибыли от параметра b , влияющего на чувствительность и поведение рынка.

Литература

1. Maschler M., Solan E., Zamir S. Game Theory. New York: Cambridge University Press, 2013. 979 p.
2. Tirole J. The theory of Industrial Organization. London: The Mit Press, 2000. 450 p.
3. Gibbons R. Game Theory for Applied Economists. New Jersey: Princeton University Press, 1992. 267 p.

УДК 519.71

Булгакова М. А.

Вектор τ для сетевых игр с попарным взаимодействием

1. Модель игры. Пусть N – конечное множество игроков, которые принимают решения в двух состояниях z , $|N| = n \geq 2$. Игра начинается в состоянии z_1 , где каждый игрок $i \in N$ выбирает поведение $b_i^1 = (b_{i1}^1, \dots, b_{in}^1)$ – n -мерный вектор предложений связи другим игрокам. Примем следующие обозначения: $M_i \subseteq N \setminus \{i\}$ – игроки, которым $i \in N$ может предложить связь, при этом значение $a_i \in \{0, \dots, n-1\}$ – максимальное число связей игрока i .

Для каждого поведения b_i^1 существует такое подмножество реализованных предложений связи $Q_i \subset M_i$, для которого выполняется

$$b_{ij}^1 = \begin{cases} 1, & \text{если } j \in Q_i, \\ 0, & \text{в противном случае,} \end{cases}$$

при дополнительном ограничении

$$\sum_{j \in N} b_{ij}^1 \leq a_i. \quad (1)$$

Условие (1) означает, что число возможных связей ограничено для каждого игрока. Также очевидно, что $|Q_i| \leq a_i$.

Связь ij реализуется тогда и только тогда, когда $b_{ij}^1 = b_{ji}^1 = 1$. Сформированные связи ij образуют ребра сети g , вершинами которой являются игроки.

Обозначим через $N_i(g)$ соседей игрока i в сети g , т. е. $N_i(g) = \{j \in N \setminus \{i\} : ij \in g\}$. Результатом первого состояния является сеть $g(b_1^1, \dots, b_n^1)$. После ее формирования игроки переходят в состояние $z_2(g)$, которое обуславливается сетью. В состоянии $z_2(g)$ соседи по сети играют попарно в одновременные биматричные игры, после чего получают выигрыши и игра заканчивается. Имеем двухшаговую игру $\Gamma_{z_1}(g)$, где стратегия – это правило, которое для каждого

Булгакова Мария Александровна – ассистент, Санкт-Петербургский государственный университет; e-mail: m.a.bulgakova@spbu.ru, тел.: +7(953)174-73-16

Работа выполнена при финансовой поддержке РФФ, грант № 17-11-01079

игрока определяет множество его возможных соседей в первом состоянии, и поведение в каждой биматричной игре во втором состоянии в соответствии со сформированной ранее сетью. Обозначим через $u_i = (b_i^1, b_i^2)$, $i \in N$, стратегию игрока i в двухшаговой игре $\Gamma_{z_1}(g)$.

2. Характеристическая функция. Во втором состоянии игра представляет собой семейство попарных одновременных биматричных игр $\{\gamma_{ij}\}$, $i \in N, j \in N_i$, между соседями на сети, с матрицами выигрышей A_{ij} и C_{ij} игроков i и j соответственно.

$$A_{ij} = \begin{pmatrix} a_{11}^{ij} & a_{12}^{ij} & \cdots & a_{1k}^{ij} \\ a_{21}^{ij} & a_{22}^{ij} & \cdots & a_{2k}^{ij} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}^{ij} & a_{m2}^{ij} & \cdots & a_{mk}^{ij} \end{pmatrix}, \quad C_{ij} = \begin{pmatrix} c_{11}^{ij} & c_{12}^{ij} & \cdots & c_{1k}^{ij} \\ c_{21}^{ij} & c_{22}^{ij} & \cdots & c_{2k}^{ij} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1}^{ij} & c_{m2}^{ij} & \cdots & c_{mk}^{ij} \end{pmatrix},$$

$$a_{pl} \geq 0, \quad c_{pl} \geq 0, \quad p = 1, \dots, m, \quad l = 1, \dots, k.$$

Константы m и k одинаковы для всех i и j . Обозначим через $\Gamma_{z_2}^S(g)$ подыгру игры Γ , которая происходит в состоянии z_2 . Рассмотрим ее в кооперативной форме. Найдем характеристическую функцию для каждой коалиции $S \subset N$ как нижнее значение антагонистической игры двух лиц: коалиции S и $N \setminus S$, построенной на основе игры $\Gamma_{z_2}^S(g)$, при этом выигрыш коалиции S рассматривается как сумма выигрышей игроков, входящих в S . Обозначим

$$w_{ij}^i = \max_p \min_\ell a_{p\ell}^{ij}, \quad w_{ij}^j = \max_\ell \min_p c_{p\ell}^{ij}, \quad p = \overline{1, m}, \quad \ell = \overline{1, k},$$

и $v(z_2; S)$, $S \subset N$, – нижнее значение антагонистической игры $\Gamma_{z_2}^S(g)$.

Лемма 1 [1]. *Функция $v(z_2; S)$ определяется следующим образом:*

$$v(z_2; \{\emptyset\}) = 0, \quad v(z_2; \{i\}) = \sum_{j \in N_i} w_{ij}^i,$$

$$v(z_2; S) = \frac{1}{2} \sum_{i \in S} \sum_{j \in N_i \cap S} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ij}) + \sum_{i \in S} \sum_{k \in N_i \setminus S} w_{ik}^i, \quad S \subset N,$$

$$v(z_2; N) = \frac{1}{2} \sum_{i \in N} \sum_{j \in N_i} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ij}).$$

Рассмотрим кооперативную форму двухшаговой игры $\Gamma_{z_1}(g)$. Для коалиции $S \subseteq N$ определим характеристическую функцию

$v(\bar{z}_1; S)$ как нижнее значение в антагонистической игре двух лиц: коалиции S , играющей как максимизирующий игрок, и коалиции $N \setminus S$ (минимизирующий игрок). В этом случае наилучшим поведением на первом шаге для минимизирующего игрока будет не создавать связи с игроками из коалиции S , тем самым уменьшая выигрыш коалиции S на величину $\sum_{i \in S} \sum_{k \in N_i \setminus S} w_{ik}^i$. Обозначим через $v(z_1; S)$, $S \subset N$, нижнее значение антагонистической игры $\Gamma_{z_1}(g)$.

Теорема 1 [1]. *Функция $v(z_1; S)$ определяется по следующим формулам:*

$$v(z_1; \{i\}) = 0, \quad v(z_1; \emptyset) = 0, \quad (2)$$

$$v(z_1; S) = \max_g \frac{1}{2} \sum_{i \in S} \sum_{j \in N_i(g) \cap S} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}), \quad S \subset N, \quad (3)$$

$$v(z_1; N) = v(z_2; N) = \max_g \frac{1}{2} \sum_{i \in N} \sum_{j \in N_i} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}). \quad (4)$$

В [1] показано, что данная игра является выпуклой.

3. Вектор τ . Рассмотрим одно из решений кооперативной двухшаговой игры $\Gamma_{z_1}(g)$ – вектор τ [2]. В силу конструкции характеристической функции можно предположить, что формулу вычисления компонент вектора τ можно упростить.

Теорема 2. *Для игры $\Gamma_{z_1}(g)$ коэффициент λ для вычисления компонент вектора τ равен $\frac{1}{2}$:*

$$\tau_i(N, v(z_1; N)) = \frac{1}{2}(v(z_1; N) - v(z_1; N \setminus \{i\})) + \frac{1}{2}v(z_1; \{i\}).$$

Доказательство. Значение характеристической функции для произвольной коалиции S в игре $\Gamma_{z_1}(g)$ определяются в соответствии с формулами (2), (3). Компоненты вектора τ для выпуклой игры вычисляются по формуле

$$\tau_i(N, v(z_1; N)) = \lambda(v(z_1; N) - v(z_1; N \setminus \{i\})) + (1 - \lambda)v(z_1; \{i\}),$$

где коэффициенты λ определяются из уравнения

$$\sum_{j \in N} (\lambda(v(z_1; N) - v(z_1; N \setminus \{j\})) + (1 - \lambda)v(z_1; \{j\})) = v(z_1; N).$$

Согласно значениям характеристической функции (2), (3), слагаемое $(1 - \lambda)v(z_1; \{i\}) = 0$. Таким образом, для $\Gamma_{z_1}(g)$ имеем следующую формулу вычисления компонент вектора τ :

$$\tau_i(N, v(z_1; N)) = \lambda(v(z_1; N) - v(z_1; N \setminus \{i\})), \quad (5)$$

где коэффициенты λ определяются из уравнения

$$\sum_{j \in N} \lambda(v(z_1; N) - v(z_1; N \setminus \{j\})) = v(z_1; N). \quad (6)$$

Вычислим значение разности $v(z_1; N) - v(z_1; N \setminus \{j\})$. Как уже упоминалось ранее, в случае двухшаговой сетевой игры с попарным взаимодействием в силу способа определения характеристической функции игрокам из коалиции S выгодно не создавать связи с коалицией $N \setminus S$. Поэтому разница между коалицией N и $N \setminus \{j\}$ будет лишь в связях, образуемых игроком j и его соседями из N . Имеем

$$v(z_1; N) - v(z_1; N \setminus \{j\}) = \sum_{i \in N_j} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}). \quad (7)$$

Подставим в уравнение (6) значения характеристической функции для разности (7) и для максимальной коалиции (4):

$$\sum_{j \in N} \lambda \left(\sum_{i \in N_j} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}) \right) = \frac{1}{2} \sum_{i \in N} \sum_{j \in N_i} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}).$$

Так как коэффициент λ не зависит от игроков и их выигрышей, то

$$\lambda \sum_{j \in N} \left(\sum_{i \in N_j} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}) \right) = \frac{1}{2} \sum_{i \in N} \sum_{j \in N_i} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}). \quad (8)$$

Преобразуя (8), получаем $\lambda = \frac{1}{2}$. Таким образом, в игре $\Gamma_{z_1}(g)$ в векторе τ коэффициент $\lambda = \frac{1}{2}$ и не зависит ни от числа игроков, ни от количества связей между ними, ни от структуры сети.

Теорема 2 доказана.

Теорема 3. В двухшаговой сетевой игре с попарным взаимодействием значение τ -value совпадает с вектором Шепли.

Доказательство. В [3] получена упрощенная формула компонент вектора Шепли для двухшаговой кооперативной игры с попарным взаимодействием на полной сети:

$$\varphi_i[v(z_1; N)] = \frac{1}{2} \sum_{j \in N} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}).$$

Найдем выражение для вектора τ на полной сети. В теореме 2 доказано, что для такого вида сети коэффициент $\lambda = \frac{1}{2}$. Учитывая (2), подставим найденный коэффициент в формулу (5):

$$\tau_i(N, v(z_1; N)) = \frac{1}{2}(v(z_1; N) - v(z_1; N \setminus \{i\})).$$

Так как рассматривается полная сеть, то множество соседей j -го игрока $N_j(g) = N \setminus \{j\}$. Поскольку $\max_{p, \ell} (a_{p\ell}^{ii} + c_{p\ell}^{ii}) = 0$, то без потери точности имеем

$$\tau_i(N, v) = \frac{1}{2} \sum_{i, j \in N} \max_{p, \ell} (a_{p\ell}^{ij} + c_{p\ell}^{ji}),$$

что совпадает с вектором Шепли.

Теорема 3 доказана.

4. Заключение. Для двухшаговой сетевой игры с попарным взаимодействием получена упрощенная формула вычисления компонент вектора τ , что значительно уменьшает вычислительную сложность данного решения. Также показано, что в аналогичной игре на полной сети вектор τ и вектор Шепли совпадают.

Литература

1. Булгакова М. А. Решения сетевых игр с попарным взаимодействием // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2019. Т. 15. № 1. С. 147–156.
2. Tijs S. H. An axiomatization of the \hat{o} -value // Mathematical Social Sciences. 1987. Vol. 13. P. 177–181.
3. Petrosyan L. A., Bulgakova M. A., Sedakov A. A. Time-consistent solutions for two-stage network games with pairwise interactions // Mobile Networks and Applications. 2018. P. 15–23.

УДК 519.72

Бусел В. Д., Лазарева А. В.

Использование различных мер центральности в задаче оптимизации передачи информации в самоорганизующихся сетях

Рекомендовано к публикации профессором Громовой Е. В.

1. Введение. В данной статье рассматривается задача по оптимизации передачи информации в самоорганизующихся сетях MANET [1] с использованием теоретико-игрового подхода [2]. В качестве агентов сети в исходной постановке задачи выступают группы спасательных служб, расположенные в местах с недоступной сотовой связью.

В основе модели лежит сетевая структура из работ [3, 4], рассматриваемая в виде графа, в вершинах которого находятся агенты, принадлежащие различным игрокам. Оптимизация сети производится с помощью дополнительных агентов, добавляемых в граф. На практике эту функцию выполняют дроны. Аналогичная задача рассматривалась в работе [5], но в ней был выбран другой критерий отбора. В данной статье в качестве критерия отбора используются метрики Betweenness Centrality [6], а также метрика, за основу которой была взята и адаптирована функция из [7].

В ходе работы написан алгоритм на языке Python, находящий наилучший вариант положения дронов по выбранному критерию. Проанализированы отдельные характеристики сети, включая данные, полученные с помощью моделирования. В результате среди рассмотренных метрик была выявлена оптимальная.

2. Постановка задачи. Пусть задано множество игроков $P = \{1, \dots, n\}$. Каждый игрок $i \in P$ имеет непустое множество $M_i \neq \emptyset$ агентов, расположенных в подмножестве $X \subset \mathbb{N}^2$, т. е. в

Бусел Владислав Дмитриевич – студент, Санкт-Петербургский государственный университет; e-mail: vladbusel@gmail.com, тел.: +7(919)637-55-12

Лазарева Алина Вячеславовна – студент, Санкт-Петербургский государственный университет; e-mail: lazer737@yandex.ru, тел.: +7(953)373-78-28

узлах целочисленной решетки. Дронами назовем дополнительных агентов, которые могут быть расположены в узлах сетки. Рассмотрим сеть, которая не меняется со временем.

Представим сеть в виде графа на сетке [5]. Определим его вершины как $v = (a, x)$, где $a \in M$, $x \in X$. Между вершинами игрока i v_i^p и v_i^s , где $s \neq p$, установлена устойчивая связь в виде ребра (v_i^p, v_i^s) , если они находятся в соседних узлах решетки. Множество вершин обозначим V , а множество ребер – E . Таким образом, получим граф $R = (V, E)$. Подграфом игрока i назовем граф, содержащий в себе всех агентов этого игрока $R_i = (V_i, E_i) \subset R$.

Задача состоит в поиске оптимального местоположения дронов, при котором суммарный выигрыш игроков максимизируется. Обоснование выбора вида функции выигрыша в данной задаче будет приведено ниже.

3. Алгоритм решения. Введем функцию выигрыша игрока i следующим образом [3]. Обозначим как d диаметр графа [6] R – это максимальное из всех кратчайших расстояний между парами вершин в графе. Соответственно, обозначим как d_i диаметр подграфа R_i игрока i . Установив подвижные агенты (дроны) в позициях, способствующих уменьшению диаметров подграфов, вычислим диаметры расширенных подграфов d_i^* игроков. В кооперативном варианте игры игроки преследуют цель максимизации функционала

$$H = \sum_{i=1, \dots, n} (d_i - d_i^*). \quad (1)$$

Оптимальное решение задачи (1) может быть не единственным, поэтому для выбора наилучшего в некотором смысле решения будут использованы две метрики: мера центральности «степень посредничества» (betweenness centrality) [6] и мера влияния узлов [7].

Мера центральности «степень посредничества» (betweenness centrality) может быть вычислена по формуле

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}, \quad (2)$$

где σ_{st} – общее число кратчайших путей из узла s в узел t , а $\sigma_{st}(v)$ равно числу этих путей, проходящих через v . Значения $g(v)$, полученные для каждого из дронов, нужно просуммировать, чтобы

получить единственное число, характеризующее качество полученного кооперативного решения.

При использовании второго подхода вычисляется метрика влияния узлов

$$\phi(x) = \sum_{k=1}^L a_k^p x^{k-1}, \quad (3)$$

где a_k^p – число всех кратчайших путей длины k , содержащих p в пути, x – параметр, отвечающий за значимость путей большей длины в сравнении с более короткими путями, $0 \leq x \leq 1$. В данной задаче при использовании второй метрики считаем ее значения для дронов в подграфах игроков.

При выборе метрики *betweenness centrality* (2) максимизируется посредничество дронов в путях между агентами игроков. В случае второй метрики (3) дроны ставятся в места с большой плотностью узлов.

4. Сравнительная характеристика. В примере игры (рис. 1) дрон первого игрока помещается в позицию (2, 4), а дрон второго – в (5, 4). Выигрыш в диаметре для первого игрока составил 2 (с 7 сократился до 5 с помощью дрона), а у второго игрока равен 3 (сократился с 9 до 6).

При использовании второй метрики, дроны расположены в позициях (6, 3) и (6, 4). Диаметр у первого игрока остался тот же, у второго стал равен 5, поэтому выигрыш составил 0 и 4 соответственно.

Таким образом, для меры *betweenness centrality* средняя длина кратчайших путей в подграфе первого игрока получилась равна 4,22; а для второго – 2,9. Для второй метрики получились следующие значения: 2,89 и 2,58 для первого и второго игроков соответственно.

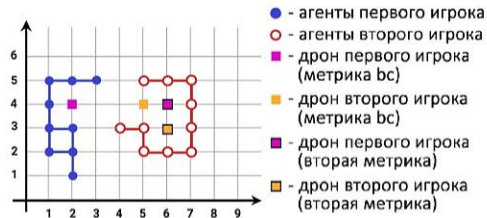


Рис. 1. Пример игры

Для сравнения показателей полученных сетей было проведено моделирование работы сети в NS-3 симуляторе [8]. При моделировании агенты отправляли данные агентам того же игрока. Полученные результаты представлены на рис. 2.

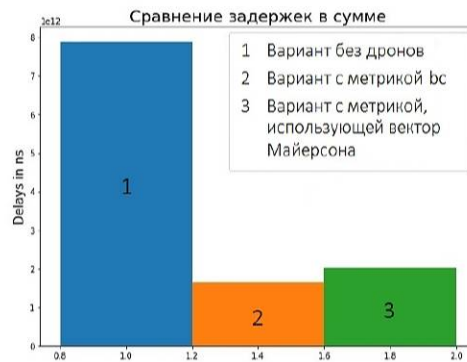


Рис. 2. Моделирование в NS-3. Задержки

Из полученных результатов можно сделать вывод, что расположение дронов, полученное с использованием первой метрики, приводит к небольшому улучшению в работе сети по сравнению с использованием второй метрики. Кроме того, следует отметить, что добавление подвижных агентов существенно улучшает качество работы сети.

5. Заключение. В работе предложено два подхода для определения местоположения подвижных агентов в случае, когда оптимальное кооперативное решение не является единственным. Были использованы две меры центральности и проведено численное моделирование, в том числе с использованием симулятора работы сети NS-3. Показано, что использование метрики «степень посредничества» приводит к незначительному улучшению качества работы сети, однако детальное изучение этого вопроса является предметом дальнейших исследований.

Литература

1. Han Z., Niyato D., Saad D. et al. Game Theory in Wireless and Communication Networks. Theory, Models and Applications. New York: Cambridge University Press, 2012. 554 p.
2. Novikov D. A. Games and networks // Automation and Remote Control. 2014. Vol. 75. No 6. P. 1145–1154.
3. Gromova E., Gromov D., Timonin N. et al. A dynamic game of mobile agents placement on MANET // Proc. of the IEEE conference SIMS 2016. P. 153–158.
4. Plekhanova T., Gromova E., Gromov D. et al. The strategic placement of mobile agents on a hexagonal graph using game theory // Proc. of the IEEE conference ICAT 2017. P 1–6.
5. Киреев С. А. Оптимизация передачи информации в самоорганизующихся сетях // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 381–386.
6. Newman M. Networks: An Introduction. 1st Edition. Oxford: Oxford University Press, 2010. 772 p.
7. Мазалов В. В., Чиркова Ю. В. Сетевые игры. 1-е изд. СПб.: Лань, 2018. 320 с.
8. NS-3 Documentation [Электронный ресурс]: URL:<https://www.nsnam.org/doxygen/index.html> (дата обращения: 21.03.21).

УДК 311.2

Кочеров И. П.

Анализ данных по ежедневному заражению КОВИД-19

Рекомендовано к публикации профессором Прасоловым А. В.

1. Введение. Пандемия, настигшая человечество более года назад, захватила большое число стран. Всемирная организация здравоохранения (ВОЗ) объявила эпидемию заболевания, вызванного SARS-CoV-2 (COVID-19). Многие организации мира собирают информацию о распространении болезни в виде ежедневных заражений в стране, количеств смертей и выздоровлений, объемов тестирования и других данных для оценки происходящего, прогнозирования динамики и, в конечном итоге, составления моделей развития болезни. Этому в известной степени препятствует отсутствие унифицированных инструкций сбора информации, а зачастую и цензурирование данных.

На первом этапе пандемии основная задача прогноза ставилась как определение даты и высоты пика ежедневных заражений, чтобы оценить мощности медицинских учреждений, способных обеспечить население соответствующим обслуживанием, а также оценить степень изоляции людей от быстрого распространения вируса. По прошествии 14 месяцев, когда в некоторых странах пандемия развивается уже третьей волной, использование классической модели SIR [1] не дает нужного результата. Основная надежда возлагается на вакцинацию и пересмотр математических моделей.

Однако большой объем данных, собранных по всему миру [2], содержит особенность, которая выглядит странной при внимательном рассмотрении. Приведем как пример Германию и США (рис. 1, 2).

Кроме существенных колебаний измеряемая величина (в данном случае – ежедневные заражения) имеет высокочастотные колебания с амплитудами случайного характера. Даже визуально легко вычислить, что эти колебания имеют период около недели. Более точный анализ подтверждает этот вывод по большинству стран.

Кочеров Иван Павлович – студент, Санкт-Петербургский государственный университет; e-mail: st061387@student.spbu.ru; тел.: +7(905)814-41-42

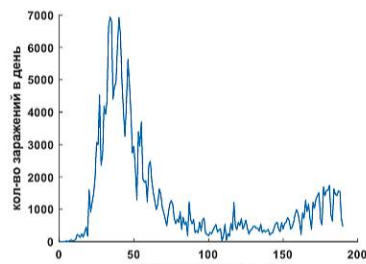


Рис. 1. Германия

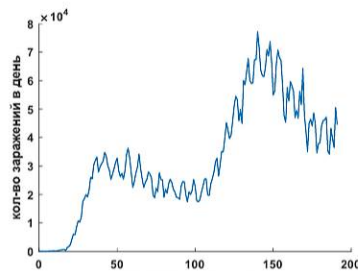


Рис. 2. США

2. Природа колебаний. Цель данной работы – на примере нескольких стран статистически доказать, что причина этих недельных колебаний лежит не в биологических смыслах вируса, а в человеческой природе правил измерения. Такие же колебания наблюдаются и, следовательно, такие же вопросы возникают при анализе данных по смертности, социальной активности и выздоровлениям.

Первый вариант объяснения заключается в следующем: данные о количестве заражений могли быть скорректированы, например, с целью предотвращения паники в стране. Допустим, раз в неделю данные изменяли, из-за чего и возникли колебания. Однако, колебания наблюдались уже в самом начале эпидемии в каждой стране, в то время, когда государство еще не регулировало распространение вируса.

Следующая идея была основана на том, что колебания имеют биологический характер: из-за биологических факторов вируса данные могут проявляться колебаниями в зависимости от времени. Например, на это могли повлиять инкубационный период или свойства размножения вируса.

3. Выявление недельного цикла. Было сделано предположение, что природа колебаний не имеет отношения к биологическим процессам. Рассматривая эту идею подробнее, вычислим период колебаний, описав их тригонометрической функцией. Наилучшее приближение дают функции с периодом именно семь дней (рис. 3, 4).

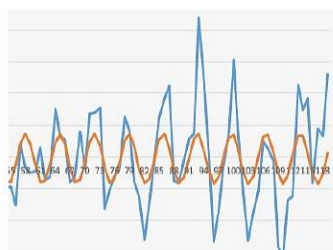


Рис. 3. Германия

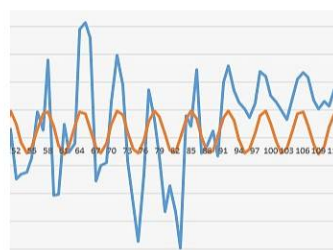


Рис. 4. США

Но неделя – мера времени, придуманная человеком.

Определим, на какие дни недели выпадают локальные минимумы. Оказалось, что минимумы во всех странах выпадают на воскресенье – понедельник.

Стало понятно, что колебания экзогенные, не биологического происхождения. Окончательно, объяснение заключается в следующем: воскресенье – выходной день. В этот день меньше врачей выходит на работу, меньше тестов на коронавирус делается, меньше лабораторий функционирует и т. п. Понедельник – рабочий день, и в этот день тесты делаются, но результаты могут приходить не в тот же день, а через какое-то время, отсюда и получается минимум именно в воскресенье – понедельник, а максимум в среду – четверг.

Утверждение. *Причиной возникновения семидневных колебаний является человеческий фактор.*

Рассмотрим уровень ежедневного прироста заражений как случайную величину. Предположим, что на сбор данных по заражениям не влияет человеческий фактор. Для вируса не существует дней недели, значит расположение локальных максимумов относительно них будет подчиняться равномерному закону распределения. Чтобы найти частоту попадания локальных максимумов на определенный день недели, проведем скользящим семидневным интервалом по данным о заражениях, выявим максимумы на этих интервалах и установим соответствие между днем недели и количеством максимумов, в него попадающих. Полученные значения – эмпирические частоты n_i . Далее используем критерий согласия Пирсона [3]:

1. Выдвигаем гипотезу H_0 : случайная величина расположения локальных максимумов подчиняется равномерному закону распределения на отрезке $[1, 7]$. Уровень доверия примем $\alpha = 0,05$.

2. Вычислим значение статистики

$$\hat{\gamma}_{\chi^2}(X_n) = \sum_{i=1}^k \frac{(n_i - np_i^{(0)})^2}{np_i^{(0)}}. \quad (1)$$

Здесь X_n – выборка, n_i – эмпирические частоты, $np_i^{(0)}$ – теоретические частоты. Для равномерного распределения значения p_i вычисляются следующим образом. Обозначим за $\hat{a} = x_{min}$, $\hat{b} = x_{max}$ оценки неизвестных параметров a и b по выборке. Тогда

$$\begin{cases} p_1^{(0)}(\hat{a}, \hat{\sigma}) = P\{\xi \in (-\infty, x_2]\} = \frac{x_2 - \hat{a}}{\hat{b} - \hat{a}}, \\ p_i^{(0)}(\hat{a}, \hat{\sigma}) = P\{\xi \in (x_i, x_{i+1}]\} = \frac{x_{i+1} - x_i}{\hat{b} - \hat{a}}, \\ p_k^{(0)}(\hat{a}, \hat{\sigma}) = P\{\xi \in [x_k, \infty)\} = \frac{\hat{b} - x_k}{\hat{b} - \hat{a}}. \end{cases} \quad (2)$$

В данной задаче из (2) находим $p_i = 0,17$. Итак, значение статистики для США

$$\hat{\gamma}_{\chi^2}(X_n) = \sum_{i=1}^7 \frac{(n_i - 217p_i^{(0)})^2}{217p_i^{(0)}} = 168,91. \quad (3)$$

3. Определим критическую область $(\chi_{1-\alpha, k-l-1}^2, \infty)$, где $\chi_{1-\alpha, k-l-1}^2$ – квантиль уровня $1 - \alpha$ распределения χ^2 с $k - l - 1$ степенями свободы, l – количество оцениваемых по выборке параметров. Если значение статистики $\hat{\gamma}_{\chi^2}(X_n)$ попадает в эту область, то гипотеза H_0 отвергается, в противном случае нет оснований отвергнуть гипотезу.

Для данной задачи $k = 7$ (кол-во элементов в рассматриваемом отрезке $[1, 7]$), $l = 2$ (параметры α, σ), $\chi_{1-\alpha, k-l-1}^2 = 9,49$. Критическая область $(9,49; \infty)$. Так как значение статистики попадает в критическую область, гипотеза H_0 отвергается. Значит распределение не равномерное, откуда и следует, что расположения максимумов относительно дней недели зависят от графика работы человека.

В качестве дополнительной проверки полученных выводов проделаем тот же анализ с локальными минимумами колебаний.

1. Выдвигаем гипотезу H_0 : случайная величина расположения локальных минимумов подчиняется равномерному закону распределения на отрезке $[1, 7]$. Уровень доверия примем $\alpha = 0,05$.
2. Вычислим значение статистики согласно (1)

$$\hat{\gamma}_{\chi^2}(X_n) = \sum_{i=1}^7 \frac{(n_i - 217p_i^{(0)})^2}{217p_i^{(0)}} = 185,67. \quad (4)$$

3. Критическая область $(9,49; \infty)$. Гипотеза H_0 отвергается.

Утверждение доказано.

Для того чтобы убрать выявленные колебания из исходных данных, существует два пути:

1. Сглаживание данных.
2. Переход к недельной системе фиксации замеров (рассмотрение не ежедневных данных, а их суммы за неделю).

Сглаживание данных методом скользящего среднего по семь дней часто встречается в аналогичных исследованиях [4]. Но при сглаживании таким способом в получившемся временном ряде не учитываются три первых и три последних элемента, что влечет за собой потерю точности при дальнейшем исследовании. Также в процессе сглаживания может возникнуть эффект Слущкого–Юла [5–7], т. е. возможно проявление систематических колебаний там, где их не было изначально.

Переход к недельной системе счисления позволяет устранить колебания без изменения исходных данных (рис. 5). Такой подход часто используют для решения подобных задач [8].

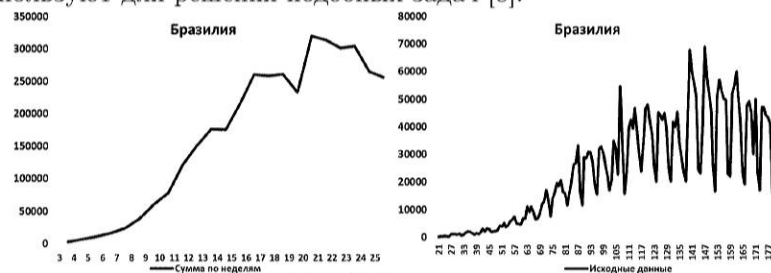


Рис. 5. Бразилия

4. Заключение. В работе проведен анализ данных, которые в настоящее время собираются по разным странам для создания математических моделей прогнозирования. На ранней стадии пандемии целью модели были определение даты и высоты пика ежедневных заражений. Высокочастотные колебания данных, а также их смещение по времени отражались только на точности оценок. На более поздних периодах, когда начинают исследовать волны распространения, по-видимому, перспективнее перейти к замерам заражений (выздоровлений и смертей) на недельной основе.

Литература

1. Cooper I., Mondal A., Antonopoulos C. A SIR model assumption for the spread of COVID-19 in different communities [Электронный ресурс] // Chaos, Solitons and Fractals. 2020. Vol. 139. URL:<https://www.sciencedirect.com/science/article/pii/S0960077920304549> (дата обращения: 10.12.2020).
2. Johns Hopkins Coronavirus Resource Center [Электронный ресурс]: URL:<https://coronavirus.jhu.edu/> (дата обращения: 20.10.2020).
3. Буре В. М., Парилина Е. М., Седаков А. А. Методы прикладной статистики. СПб.: Лань, 2019. 148 с.
4. Грешилов А. А., Стакун В. А., Стакун А. А. Математические методы построения прогнозов. М.: Радио и связь, 1997. 112 с.
5. Домбровский В. В. Эконометрика: учебник. М.: Новый учебник, 2004. 342 с.
6. Юл Д. Э., Кендэл М. Дж. Теория статистики / пер. с англ. Ф. Лившиц. М.: Госстатиздат, 1960. 779 с.
7. Slutsky E. The summation of random causes as the source of cyclic processes // Econometrica. Vol. 5. No 2. P. 105–146.
8. Cacciapaglia G., Cot C., Francesco S. Second wave COVID-19 pandemics in Europe: a temporal playbook [Электронный ресурс] // Scientific Reports. 2020. URL:<https://www.nature.com/articles/s41598-020-72611-5> (дата обращения: 12.11.2020).

УДК 519.87

Магнитская Н. Г.

Супераддитивное расширение характеристической функции в игре управления инвестициями

Рекомендовано к публикации профессором Громовой Е. В.

1. Введение. В настоящее время кооперативные игры занимают важное место в научных исследованиях. Построение характеристических функций является одной из основных задач теории кооперативных игр. Одним из важных свойств характеристической функции является супераддитивность. Использование супераддитивной функции при решении задач дает ряд преимуществ [1]. В работе [2] для однократных кооперативных игр предложен способ, позволяющий для произвольной характеристической функции $V(S)$ построить ее супераддитивное расширение. Этот способ затем развит в работе [3] для класса дифференциальных игр. В данной работе рассмотрена дифференциальная игра управлениями инвестициями, для которой построена δ -характеристическая функция [4] и ее супераддитивное расширение.

2. Постановка задачи. Рассмотрим дифференциальную игру n лиц, где динамика игры задается системой обыкновенных дифференциальных уравнений

$$\dot{x}(t) = g(t, x, u), \quad x(t_0) = x_0,$$

$$x \in \mathbb{R}^l, \quad u = (u_1, \dots, u_n), \quad u_i = u_i(t) \in U_i \subset \text{comp} \mathbb{R}^k.$$

Игра начинается из состояния x_0 в момент времени t_0 . Предполагаем, что продолжительность игры T представляет собой случайную величину с известной функцией распределения $F(t)$. Предположим, что функция распределения имеет вид

$$F(\tau) = \begin{cases} 0, & \tau < \bar{T} - \delta, \\ \varphi(\tau), & \bar{T} - \delta \leq \tau < \bar{T} + \delta, \\ 1, & \tau \geq \bar{T} + \delta, \end{cases} \quad (1)$$

Магнитская Наталья Григорьевна – магистрант, Санкт-Петербургский государственный университет; e-mail: st048764@student.spbu.ru, тел.: +7(981)692-80-07

где $\varphi(\tau)$ – абсолютная непрерывная неубывающая функция, удовлетворяющая следующим условиям: $\varphi(\bar{T}-\delta) = 0$, $\varphi(\bar{T}+\delta) = 1$. Из этого следует, что игра может закончиться в течение следующего периода времени $[\bar{T}-\delta, \bar{T}+\delta]$, где $\delta \in [t_0, \bar{T}]$. В работе [5] выведена формула для математического ожидания выигрыша:

$$K_i(x_0, \delta, \bar{T}; u) = \int_{t_0}^{\bar{T}-\delta} h^i[s, x(s), u] ds + \int_{\bar{T}-\delta}^{\bar{T}+\delta} h^i[s, x(s), u](1 - F(s)) ds. \quad (2)$$

3. Дифференциальная игра управления инвестициями.

Рассмотрим дифференциальную игру управления объемами инвестиций. Предположим, что существуют n лиц, которые инвестируют средства в акции, связанные с научной сферой [6]. Пусть $x(t)$ – запас акций в момент времени t , связанных с научной сферой, под управлением игрока i будем понимать темпы инвестирования в единицу времени $u_i(t, x)$. Динамика изменения фазовой переменной $x(t)$ подчиняется уравнению

$$\dot{x}(t) = u_1(t) + \dots + u_n(t), \quad x(0) = x_0.$$

Пусть функция полезности $h^i[s, x(s), u]$ для каждого игрока i имеет линейный по x и квадратичный по u вид. Тогда ожидаемое вознаграждение игрока $i \in N$ представимо как математическое ожидание

$$E \left(\int_0^T (q_i x(t) - r_i u_i^2(t)) dt \right),$$

и далее может быть представлено в виде суммы интегралов (2).

Пусть случайная величина T равномерно распределена на интервале $[\bar{T}-\delta, \bar{T}+\delta]$. Тогда функция распределения (1) имеет вид

$$F(\tau) = \begin{cases} 0, & \tau < \bar{T} - \delta, \\ \frac{t - \bar{T} + \delta}{2\delta}, & \bar{T} - \delta \leq \tau < \bar{T} + \delta, \\ 1, & \tau \geq \bar{T} + \delta. \end{cases}$$

Метод решения дифференциальных игр со случайным моментом окончания с разрывной функцией распределения в программных стратегиях рассмотрен в [7].

4. Управления для кооперативной и некооперативной формы игры. Рассмотрим равновесие по Нэшу в программных стратегиях, используя принцип максимума Понтрягина [8]. Получаем следующие управления:

$$u_i^{NE}(t)_{I_1} = \frac{q_i(\bar{T} - t)}{2r_i}, \quad t \in [0, \bar{T} - \delta],$$

$$u_i^{NE}(t)_{I_2} = \frac{q_i(\bar{T} + \delta - t)}{4r_i}, \quad t \in [\bar{T} - \delta, \bar{T} + \delta].$$

Оптимальные управления для кооперативной формы игры:

$$u_i^*(t)_{I_1} = \frac{\hat{q}(\bar{T} - t)}{2r_i}, \quad t \in [0, \bar{T} - \delta],$$

$$u_i^*(t)_{I_2} = \frac{\hat{q}(\bar{T} + \delta - t)}{4r_i}, \quad t \in [\bar{T} - \delta, \bar{T} + \delta],$$

$$\hat{q} = q_1 + \dots + q_n.$$

5. Характеристическая функция. Для кооперативной игры найдем значение характеристической функции $V(S, \cdot)$ для каждой коалиции $S \subseteq N$ [9]. Характеристическая функция в кооперативной игре – отображение из множества всех возможных коалиций:

$$V(\cdot) : 2^N \rightarrow R, \quad V(\emptyset) = 0.$$

Игра обладает свойством супераддитивности, если

$$V(S_1 \cup S_2) \geq V(S_1) + V(S_2), \quad \forall S_1, S_2 \subseteq N, \quad S_1 \cap S_2 = \emptyset.$$

Будем использовать следующее определение характеристической функции:

$$V^\delta(S, \cdot) = \begin{cases} 0, & S = \emptyset, \\ \max_{\substack{u_i, i \in S \\ u_j = u_j^{NE}, j \in N \setminus S}} \sum_{i \in S} K_i(u_S, u_{N \setminus S}^{NE}), & S \subset N, \\ \max_{u_1, \dots, u_n} \sum_{i=1}^n K_i(u_1, \dots, u_n), & S = N. \end{cases} \quad (3)$$

В общем случае функция (3) не является супераддитивной.

Используя принцип максимума Понтрягина, получаем соответствующие управления для решения вспомогательных задач оптимизации и вычисляем значение характеристической функции для коалиции S :

$$V^\delta(S, x(t)) = -\frac{(\bar{T} - \delta) (\bar{T}^2 \dot{q}_S^2 + \bar{T} \delta \dot{q}_S^2 + \delta^2 \dot{q}_S^2)}{12} \sum_{i \in S} \frac{1}{r_i} + \\ + \dot{q}_S x \bar{T} - \frac{\delta^3 \dot{q}_S^2}{8} \sum_{i \in S} \frac{1}{r_i}, \quad \dot{q}_S = \sum_{i \in S} q_i.$$

6. Супераддитивность характеристической функции.

Проверим супераддитивность функции на примере трех игроков, $|N| = 3$.

$$V^\delta(N, x(t)) - V^\delta(\{j, k\}, x(t)) - V^\delta(\{i\}, x(t)) = \\ = \left(-(q_i + q_j + q_k)^2 \left(\frac{1}{r_i} + \frac{1}{r_j} + \frac{1}{r_k} \right) + \right. \\ \left. + q_i^2 \frac{1}{r_i} + (q_j + q_k)^2 \left(\frac{1}{r_j} + \frac{1}{r_k} \right) \right) C < 0,$$

$$V^\delta(\{j, k\}, x(t)) - V^\delta(\{j\}, x(t)) - V^\delta(\{k\}, x(t)) = \\ = C \left(-2q_j q_k \left(\frac{1}{r_j} + \frac{1}{r_k} \right) - \frac{q_j^2}{r_k} - \frac{q_k^2}{r_j} \right) < 0,$$

где $C = \frac{(\bar{T} - \delta)(\bar{T}^2 + \bar{T}\delta + \delta^2)}{12} + \frac{\delta^3}{8} > 0$.

Таким образом, построенная характеристическая функция не является супераддитивной.

7. Супераддитивное расширение характеристической функции. В работе [2] предложен следующий метод построения супераддитивного расширения характеристической функции:

$$\bar{V}(S, \cdot) = \max_{Q \in \mathcal{Q}(S)} \left\{ \sum_{k=1}^l V(Q_k, \cdot) \mid Q_k \in \mathcal{Q}, l = |Q| \right\}.$$

Согласно данному подходу, построим новую характеристическую функцию

$$\begin{aligned}\bar{V}^\delta(N) &= V^\delta(\{i\}, x(t)) + V^\delta(\{j\}, x(t)) + V^\delta(\{k\}, x(t)) = \\ &= (q_i + q_j + q_k)x\bar{T} - \left(\frac{(\bar{T} - \delta)(\bar{T}^2 + \bar{T}\delta + \delta^2)}{12} + \frac{\delta^3}{8} \right) \left(\frac{q_i^2}{r_i} + \frac{q_j^2}{r_j} + \frac{q_k^2}{r_k} \right),\end{aligned}$$

$$\begin{aligned}\bar{V}^\delta(\{j, k\}) &= V^\delta(\{j\}, x(t)) + V^\delta(\{k\}, x(t)) = \\ &= (q_j + q_k)x\bar{T} - \left(\frac{(\bar{T} - \delta)(\bar{T}^2 + \bar{T}\delta + \delta^2)}{12} + \frac{\delta^3}{8} \right) \left(\frac{q_j^2}{r_j} + \frac{q_k^2}{r_k} \right), \\ &\quad \forall j, k \in N, j \neq k,\end{aligned}$$

$$\bar{V}^\delta(\{j\}) = q_j x \bar{T} - \left(\frac{(\bar{T} - \delta)(\bar{T}^2 + \bar{T}\delta + \delta^2)}{12} + \frac{\delta^3}{8} \right) \frac{q_j^2}{r_j}, \quad \forall j \in N.$$

Можно убедиться, что построенная функция является супераддитивной. На основе новой характеристической функции могут быть построены различные принципы оптимальности в кооперативной игре: вектор Шепли, С-ядро и др.

8. Заключение. В работе исследована одна дифференциальная игра управления объемами инвестиций на случайном временном промежутке, имеющем равномерное распределение. При помощи принципа максимума Понтрягина получены выражения для оптимальных управлений и равновесия по Нэшу. Для кооперативной формы игры построена δ -характеристическая функция. Доказано, что она не является супераддитивной функцией, а затем получено ее супераддитивное расширение.

Литература

1. Gromova E. V., Petrosyan L. A. On an approach to constructing a characteristic function in cooperative differential games // Automation and Remote Control. 2017. Vol. 78. P. 1680–1692.
2. Aumann R. J., Dreze J. H. Cooperative games with coalition structures // International Journal of game theory. 1974. Vol. 3. P. 217–237.

3. Громова Е. В., Марова Е. В. Супераддитивное расширение характеристических функций в кооперативных дифференциальных играх // Математическая теория игр и ее приложения. 2020. Т. 2. Вып. 3. С. 19–39.
4. Petrosjan L., Zaccour G. Time-consistent Shapley value allocation of pollution cost reduction // Journal of Economic Dynamics and Control. 2003. Vol. 27. Iss. 3. P. 381–398.
5. Костюнин С. Ю., Шевкопляс Е. В. Об упрощении интегрального выигрыша в дифференциальных играх со случайной продолжительностью // Вестник Санкт-Петербургского университета. Серия 10: Прикладная математика. Информатика. Процессы управления. 2011. Вып. 4. С. 47–56.
6. Dockner E. J., Jorgensen S., Long N. V., Sorger G. Differential Games in Economics and Management Science. Cambridge Books, 2000. 396 p.
7. Магнитская Н. Г. Одна дифференциальная игра со случайным дискретным моментом окончания // Процессы управления и устойчивость. 2019. Т. 6. № 1. С. 445–450.
8. Понтрягин Л. С. Математическая теория оптимальных процессов. М.: Наука, 1961. 392 с.
9. Gromova E. V., Marova E. V. On the characteristic function construction technique in differential games with prescribed and random duration // Contributions to Game Theory and Management. 2018. Vol. 11. P. 53–65.

УДК 519.85

Маркелова А. Ю., Лаисцева М. Ю., Петросян О. Л.

Управление микрогрид с системой хранения энергии и генерацией возобновляемой энергии

1. Введение. Системы энергоменеджмента включают в себя планирование и эксплуатацию энергопроизводящих и энергопотребляющих установок, а также распределение и хранение энергии [1]. В данной работе рассмотрена одна из задач энергоменеджмента – планирование графика хранения энергии с целью минимизации финансовых затрат пользователей.

2. Постановка задачи. Предложенная оптимизационная модель минимизирует расходы на покупку электроэнергии за счет составления расписания для заряда и разряда энергии накопительной батареи, а также обмена с энергетическим рынком при условии выполнения ограничений системы и достижением энергетического баланса. На рис. 1 приведена модель системы энергоменеджмента, управляющей накопительной батареей, где прямая линия – поток электроэнергии, а пунктирная линия – поток данных.

2.1. Целевая функция. Минимальные затраты на использование электроэнергии сети могут быть достигнуты путем покупки энергии в периоды низких цен и продажи ее в периоды высоких цен, а так же накопление и последующее использование энергии в часы пика [2]. Целевую функцию оптимизационной модели можно сформулировать следующим образом:

$$F = \sum_{t=0}^{95} P_{buy}(t)C_{buy}(t) - \sum_{i=0}^{95} P_{sell}(t)C_{sell}(t), \quad (1)$$

Маркелова Анастасия Юрьевна – студент, Санкт-Петербургский государственный университет; e-mail: st061336@student.spbu.ru, тел.: +7(937)168-34-33

Лаисцева Милена Юрьевна – студент, Санкт-Петербургский государственный университет; e-mail: st063214@student.spbu.ru, тел.: +7(967)976-03-87

Петросян Ованес Леонович – доцент, Санкт-Петербургский государственный университет; e-mail: petrosian.ovanes@yandex.ru, тел.: +7(911)740-80-19

где $C_{buy}(t)$ и $C_{sell}(t)$ – тарифы в момент времени t на покупку и продажу электроэнергии соответственно, $P_{buy}(t)$ и $P_{sell}(t)$ – мощности закупленной и проданной электроэнергии в момент времени t соответственно. Первая часть формулы (1) – это общая стоимость электроэнергии, импортируемой из электрической сети, вторая часть относится к продаже электроэнергии сетевой компании в течение заданных периодов времени.

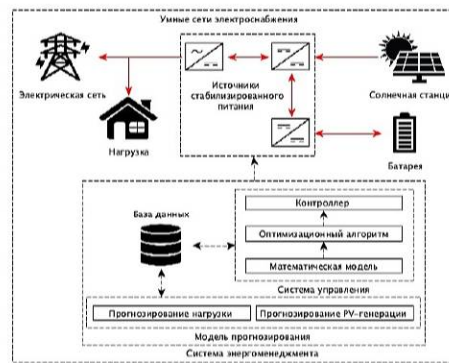


Рис. 1. Модель энергоменеджмента

2.2. Ограничения. В оптимизационной модели для расчета допустимого решения функции затрат рассматриваются следующие ограничения. Ограничение энергетического баланса указывает на то, что возобновляемые источники энергии, батареи и электросети должны удовлетворять потребность сети в электроэнергии на каждом шаге, а именно:

$$P_{buy}(t) + P_{pv}(t) - P_{dch}(t) = L(t) + P_{ch}(t) + P_{sell}(t), \quad \forall t,$$

где $L(t)$ – прогнозируемая нагрузка в момент времени t , $P_{pv}(t)$ – прогнозируемая мощность фотоэлектрической станции в момент времени t , $P_{ch}(t)$, $P_{dch}(t)$ – соответственно мощность заряда и мощность разряда батареи в момент времени t .

Соотношение мощности батареи между двумя последовательными шагами определяется по формуле:

$$SOC(t) = SOC(t - 1) + P_{ch}(t)\eta^{ch} + P_{dch}(t)/\eta^{dch}, \quad \forall t, t \neq 1,$$

где $SOC(t)$ – остаточная мощность в батарее в момент времени t , η_{ch} – эффективность зарядки аккумулятора, η_{dch} – эффективность разрядки аккумулятора.

При моделировании работы аккумуляторной системы должны быть удовлетворены ограничения на скорость зарядки, скорость разрядки и емкость батареи:

$$0 \leq P_{ch}(t) \leq P_{ch}^{max}, \quad P_{dch}^{max} \leq P_{dch}(t) \leq 0,$$

$$SOC_{min} \leq SOC(t) \leq SOC_{max},$$

где P_{ch}^{max} – максимальная зарядная мощность батареи, P_{dch}^{max} – максимальная разрядная мощность батареи, SOC_{min} – минимальное состояние заряда батареи, SOC_{max} – максимальное состояние заряда батареи.

3. Моделирование. Механизм моделирования передает данные контроллеру батареи на каждом временном шаге и запрашивает SOC для следующего временного шага. После установления SOC батареи рассчитывается энергия, необходимая для удовлетворения энергетического баланса. Если полученное значение положительно, энергия будет закуплена из сети в необходимом объеме, если отрицательно, то это количество энергии продается в сеть.

4. Метрика. Для представления результатов оптимизации применяется метрика, равная среднему значению относительных затрат на покупку электроэнергии с использованием батареи – C_{BESS} , к затратам на покупку электроэнергии без батареи – C после моделирования для всех тестовых случаев:

$$Score = \frac{1}{N} \sum_{i=0}^N \frac{\sum_{j=1}^{D_i} C_{BESS}^{ij} - \sum_{j=1}^{D_i} C^{ij}}{\sum_{j=1}^{D_i} C^{ij}},$$

где N – количество тестовых случаев, $N = 11$, D_i – количество дней для построения управлений в тестовом случае i .

5. Описание данных. Schneider Electric – энергомашиностроительная компания, специализирующаяся на управлении энергией и автоматизации, опубликовала в открытом доступе данные, которые были предложены для решения задач энергоменеджмента в соревновании “Power Laws: Optimizing Demand-side Strategies” [3]. Наборы

данных содержат 11 тестовых случаев различного периода моделирования (до 10 дней), каждый из которых содержит информацию о характеристиках батареи. Для каждого момента времени предлагается многомерный временной ряд, состоящий из прогнозных значений энергопотребления, выходной мощности фотоэлектрической станции, тарифа на электроэнергию на следующие сутки с шагом в 15 минут.

6. Результаты моделирования. Для решения задачи были использованы различные пакеты программного обеспечения с открытым кодом, предназначенные для решения задач линейного программирования: CBC [4], CLP [5], GLOP [6], SCIP [7]. Эксперименты проводились на машине, имеющей следующие характеристики: 2-ядерный процессор Intel Core i5, 1,8 GHz, 8 ГБ. Репозиторий Github с реализацией протестированных моделей и визуализацией результатов можно найти в [8].

Таблица. Сравнение пакетов

№	CBC	CLP	GLOP	SCIP
Время(с)	1785	1397	1286	5419
Score	-0,1858	-0,1852	-0,1863	-0,1890

Наиболее эффективным по времени выполнения и результатам оптимизации является пакет GLOP, что видно из результатов, приведенных в таблице.

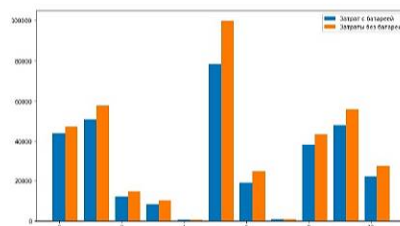


Рис. 2. Результаты моделирования GLOP

На рис. 2 представлено сравнение затрат на покупку электроэнергии с использованием батареи и без батареи для всех тестовых случаев.

7. Выводы. Разработана модель для решения задачи энергоменеджмента – определения оптимального графика хранения электроэнергии с целью минимизации финансовых затрат пользователя. Приведено сравнение пакетов программного обеспечения с открытым кодом, предназначенных для решения задач линейного программирования. Лучшие результаты оптимизации получены с помощью пакета GLOP, и обеспечивают снижение затрат на покупку электроэнергии на 18,63%. В качестве дальнейших исследований, для увеличения энергоэффективности моделируемой системы, будут рассмотрены дополнительно технологии распределенной энергетики, такие как системы когенерации.

Литература

1. Lupangu C., Bansal R. C. A review of technical issues on the development of solar photovoltaic systems // Renewable and Sustainable Energy Reviews. 2017. P. 950–965.
2. Li X., Wang S. A review on energy management, operation control and application methods for grid battery energy storage system // CSEE Journal of Power and Energy Systems. 2019. P. 1–15.
3. Power Laws: Optimizing Demand-side Strategies [Электронный ресурс]: URL:<https://www.drivendata.org/competitions/53/optimize-photovoltaic-battery/page/105/> (дата обращения: 12.03.21).
4. CBC Package [Электронный ресурс]: URL:<https://projects.coin-or.org/Cbc> (дата обращения: 12.03.21).
5. CLP Package [Электронный ресурс]: URL:<https://projects.coin-or.org/Clp> (дата обращения: 12.03.21).
6. GLOP Package [Электронный ресурс]: URL:<https://developers.google.com/optimization/lp/glop> (дата обращения: 12.03.21).
7. SPIC Package [Электронный ресурс]: URL:<https://www.scipopt.org> (дата обращения: 12.03.21).
8. Github [Электронный ресурс]: URL:<https://github.com/Nastiyam/Smart-energy-CPS-> (дата обращения: 12.03.21).

УДК 517.977.5

Нечипорук А. А.

Эквивалентность моделей динамики мнений в социальных сетях

Рекомендовано к публикации профессором Парилиной Е. М.

1. Введение. Исследование моделей динамики мнений в социальных группах начиналось с работ о достижимости консенсуса. Основоположной в этой области является модель Де Гроота [1]. В своей работе он привел простую модель динамики мнений, в которой каждый участник изменяет свое мнение, взвешивая собственное мнение и мнения связанных с ним участников. Эта модель была модернизирована добавлением управления. Например, в [2, 3] рассмотрена проблема контроля мнений агентов через влияние на некоторых агентов социальной сети. В этом случае игроки контролируют мнения агентов. Цель контроля – сделать среднее мнение об агентах как можно ближе к желаемому.

Цель данной работы – модернизация модели, представленной в [2, 3], для возможности преобразования сетевой структуры без влияния на оптимальное управление, что может стать основой для выводов о взаимосвязи различных видов модели.

2. Модель. За основу взята модель, описанная в [3]. Рассматривается динамика мнений в обществе с бесконечным временем. Общество представлено парой (N, g) , где N – конечный набор агентов общества, а g – граф, который отражает коммуникационную структуру. Граф $g = (N, E)$ определяется как N – множество вершин, E – множество ребер. Помимо агентов общества существует независимый член общества, называемый игроком, который контролирует мнение агентов. В качестве примера можно рассматривать медицентр как игрока, влияющего на мнение агентов. Динамика мнений в модели определяется следующими уравнениями:

$$x_i(t+1) = x_i(t) + a_i \left(\frac{x_i(t) + \sum_{j \in S_i} x_j(t)}{|S_i| + 1} - x_i(t) \right) + u_i(t), \quad i \in N, \quad (1)$$

Нечипорук Александра Алексеевна – студент, Санкт-Петербургский государственный университет; e-mail: st044395@student.spbu.ru, тел.: +7(962)694-65-24

где $x_i(t) \in \mathbb{R}^1$ – мнение агента i в момент времени t , $S_i = \{j : (i, j) \in E\}$ – множество соседей агента i в графе g , $a_i \in \mathbb{R}_+$ – заданный коэффициент для любого агента общества, $u_i \in U \subset [0, \infty)$ является управлением игрока на агента i .

Цель игрока состоит в том, чтобы поддерживать мнение агентов общества ближе к определенному уровню \bar{x} , сводя к минимуму его затраты на контроль мнения агентов. Следующий функционал представляет расходы игрока:

$$J(u) = \sum_{t=0}^{\infty} \delta^t \sum_{i=1}^n [(x_i(t) - \bar{x})^2 + \gamma u_i^2(t)],$$

где $\delta \in (0, 1)$ – коэффициент дисконтирования, $\gamma > 0$ – стоимость единицы контроля.

3. Модернизация модели. Рассмотрим модернизированную версию представленной выше модели. В ней рассмотрим динамику мнений, определяемую уравнениями (1), но при этом в функционал добавим «цену» каждого агента следующим образом:

$$J(u) = \sum_{t=0}^{\infty} \delta^t \sum_{i=1}^n [q_i(x_i(t) - \bar{x})^2 + \gamma u_i^2(t)],$$

где $q_i \in \mathbb{N}$ – цена агента i .

4. Эквивалентность моделей. Рассмотрим несколько моделей с различными графами и покажем соотношение параметров, при котором они будут эквивалентны и будут иметь решения, равные по величине.

4.1. Определение моделей. Выделим два графа, порождающих модели.

4.1.1. Граф-звезда. Первый граф характеризуется тем, что один агент связан со всеми остальными (центр), все остальные агенты связаны только с этим агентом, и больше ни с кем. Воздействие со стороны игрока происходит только на центр. Такой граф представлен на рис. 1. Динамика мнений на этом графе будет определяться уравнениями:

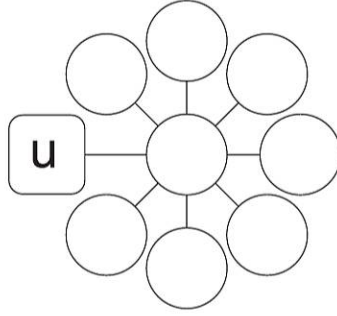


Рис. 1. Изображение графа-звезды при $n = 8$

$$\tilde{x}_i(t+1) = \begin{cases} \tilde{x}_i(t) + \tilde{a}_i \left(\frac{\sum_{j=1}^n \tilde{x}_j(t)}{n} - \tilde{x}_i(t) \right) + u(t), & i = 1, \\ \tilde{x}_i(t) + \tilde{a}_i \left(\frac{\tilde{x}_i(t) + \tilde{x}_1(t)}{2} - \tilde{x}_i(t) \right), & i \neq 1. \end{cases} \quad (2)$$

4.1.2. Граф «2 агента».

Второй граф представляет собой две вершины и связь между ними. Игрок воздействует на одного из агентов, как изображено на рис. 2. Динамика мнений для этого случая определяется уравнениями:

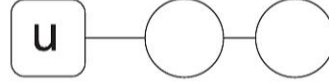


Рис. 2. Изображение графа «2 агента»

$$\hat{x}_i(t+1) = \begin{cases} \hat{x}_i(t) + \hat{a}_1 \left(\frac{\hat{x}_1(t) + \hat{x}_2(t)}{2} - \hat{x}_1(t) \right) + u(t), & i = 1, \\ \hat{x}_i(t) + \hat{a}_2 \left(\frac{\hat{x}_1(t) + \hat{x}_2(t)}{2} - \hat{x}_2(t) \right), & i = 2. \end{cases} \quad (3)$$

4.2. Эквивалентность моделей. Показано соотношение параметров, при котором модели и задачи управления, соответствующие им, эквивалентны.

Утверждение 1. Для модели динамики, определяемой уравнениями (2), при $\tilde{a}_2 = \tilde{a}_i$ и $\tilde{x}_2(0) = \tilde{x}_i(0)$ для $\forall i \in \{3, 4, \dots, n\}$ и модели, задаваемой уравнениями (3), при одинаковом управлении и при соотношении констант $\hat{a}_2 = \tilde{a}_2$, $\hat{a}_1 = \tilde{a}_1 \frac{2(n-1)}{n}$ и $\hat{x}_1(0) = \tilde{x}_1(0)$,

$\widehat{x}_2(0) = \widetilde{x}_2(0)$ будут выполняться равенства:

$$\widehat{x}_1(t) = \widetilde{x}_1(t), \quad \widehat{x}_2(t) = \widetilde{x}_2(t) = \widetilde{x}_3(t) = \dots = \widetilde{x}_n(t) \quad \forall t. \quad (4)$$

Доказательство. Используем метод математической индукции. По условию равенства (4) верны для $t = 0$. Предполагаем, что выполняются равенства $\widehat{x}_1(t) = \widetilde{x}_1(t)$, $\widehat{x}_2(t) = \widetilde{x}_2(t) = \widetilde{x}_3(t) = \dots = \widetilde{x}_n(t)$. Покажем, что выполняются равенства $\widehat{x}_1(t+1) = \widetilde{x}_1(t+1)$, $\widehat{x}_2(t+1) = \widetilde{x}_2(t+1) = \widetilde{x}_3(t+1) = \dots = \widetilde{x}_n(t+1)$. Используя (2), (3), а также условия утверждения о равенстве изначальных состояний системы и коэффициентов a , получим

$$\begin{aligned} \widetilde{x}_1(t+1) &= \widetilde{x}_1(t) + \widetilde{a}_1 \left(\frac{\sum_{j=1}^n \widetilde{x}_j(t)}{n} - \widetilde{x}_1(t) \right) + u(t) = \\ &= \widehat{x}_1(t) + \widehat{a}_1 \frac{n}{2(n-1)} \left(\frac{\widehat{x}_1(t) + \sum_{j=2}^n \widehat{x}_2(t)}{n} - \widehat{x}_1(t) \right) + u(t) = \\ &= \widehat{x}_1(t) + \widehat{a}_1 \frac{n}{2(n-1)} \left(\frac{(n-1)\widehat{x}_2(t) - (n-1)\widehat{x}_1(t)}{n} \right) + u(t) = \\ &= \widehat{x}_1(t) + \widehat{a}_1 \left(\frac{\widehat{x}_2(t) - \widehat{x}_1(t)}{2} \right) + u(t) = \widehat{x}_1(t+1). \end{aligned}$$

Тогда $\forall i \in \{2, \dots, n\}$ имеем

$$\begin{aligned} \widetilde{x}_i(t+1) &= \widetilde{x}_i(t) + \widetilde{a}_i \left(\frac{\widetilde{x}_i(t) + \widetilde{x}_1(t)}{2} - \widetilde{x}_i(t) \right) = \\ &= \widehat{x}_i(t) + \widehat{a}_i \left(\frac{\widehat{x}_2(t) + \widehat{x}_1(t)}{2} - \widehat{x}_i(t) \right) = \widehat{x}_i(t+1). \end{aligned}$$

Утверждение 1 доказано.

Утверждение 2. При условиях, описанных в утверждении 1, задачи, определяемые функционалами

$$J(u) = \sum_{t=0}^{\infty} \delta^t \sum_{i=1}^n [\widetilde{q}_i(\widetilde{x}_i(t) - \bar{x})^2 + \gamma u_i^2(t)], \quad (5)$$

$$J(u) = \sum_{t=0}^{\infty} \delta^t \sum_{i=1}^2 [\widehat{q}_i(\widehat{x}_i(t) - \bar{x})^2 + \gamma u_i^2(t)], \quad (6)$$

для моделей, определяемых графом-звездой и графом «2 агента» соответственно, являются эквивалентными при $\hat{q}_1 = \tilde{q}_1$, $\hat{q}_2 = \sum_{i=2}^n \tilde{q}_i$.

Доказательство. Отметим, что $u_i \neq 0$ только при $i = 1$. Значит функционалы (5), (6) могут быть записаны как

$$J(u) = \sum_{t=0}^{\infty} \delta^t \left[\tilde{q}_1 (\tilde{x}_1(t) - \bar{x})^2 + \gamma u^2(t) + \sum_{i=2}^n \tilde{q}_i (\tilde{x}_i(t) - \bar{x})^2 \right], \quad (7)$$

$$J(u) = \sum_{t=0}^{\infty} \delta^t \left[\hat{q}_1 (\hat{x}_1(t) - \bar{x})^2 + \gamma u^2(t) + \hat{q}_2 (\hat{x}_2(t) - \bar{x})^2 \right], \quad (8)$$

соответственно. При подстановке равенств (4) в (7) получаем

$$J(u) = \sum_{t=0}^{\infty} \delta^t \left[\tilde{q}_1 (\hat{x}_1(t) - \bar{x})^2 + \gamma u^2(t) + (\hat{x}_2(t) - \bar{x})^2 \sum_{i=2}^n \tilde{q}_i \right], \quad (9)$$

При \hat{q}_i , заданными в условии, равенство (9) будет совпадать с функционалом (8).

Утверждение 2 доказано.

5. Заключение. Рассмотрена и модернизирована модель динамики мнений в обществе с бесконечным временем. Определены отдельные графы и показано соотношение параметров, при котором определяется взаимно-однозначное соответствие между моделями, определяемыми этими графами. Получен вид функционалов, соответствующий этим моделям, и соотношение параметров, при которых задачи управления эквивалентны.

Литература

1. De Groot M. H. Reaching a consensus // Journal of the American Statistical Association. 1974. Vol. 69. No 345. P. 118–121.
2. Mazalov V. V., Parilina E. M. The Euler-equation approach in average-oriented opinion dynamics // Mathematics. 2020. Vol. 8. No 3. P. 1–6.
3. Mazalov V. V., Dorofeeva Y. A., Parilina E. M. Opinion control in a team with complete and incomplete communication // Contributions to Game Theory and Management. 2020. Vol. 13. P. 324–334.

УДК 51-76

Никитина А. А.

Оценка параметров модели SQAIRD с учетом временных интервалов

Рекомендовано к публикации доцентом Губар Е. А.

1. Введение. С текущей пандемической ситуацией исследования в области математического моделирования эпидемических процессов не теряют свою актуальность и продолжают развиваться, зачастую на основе уже выведенных моделей, доказавших свою эффективность. Моделирование данных процессов помогает отвечать на множество различных вопросов, касающихся наглядного представления процесса распространения вируса в зависимости от заданных условий и рассматриваемого окружения, и существует множество реализаций возможных подходов к моделированию и подбору параметров. Задача заключается в том, чтобы на основании итальянских статистических данных по COVID-19 определить параметры модифицированной SIR-модели [1] (S – susceptible, I – infected, R – recovered), представляющей собой систему дифференциальных уравнений. В моделях данного класса вся популяция разбивается на подгруппы в зависимости от статуса агента популяции, где статусы могут быть следующими: восприимчивые (S – susceptible), инфицированные (I – infected), выздоровевшие (R – recovered), умершие (D – dead) и т. п., а сама модель описывает динамику перехода между этими состояниями.

Изучение базовой SIR-модели привело к появлению различных модификаций эпидемических моделей: в частности SEIR-модели (S – susceptible, E – exposed, I – infected, R – recovered) [2], на которую опираются многие последующие модели, например, касаясь актуальной темы распространения COVID-19; модели SAPHIRE (S – susceptible, A – unascertained cases, P – presymptomatic infectious, H – isolated in hospital, I – ascertained infectious, R – removed, E – exposed) [3], которая представляет процесс распространения вируса в городе Ухань;

Никитина Анастасия Александровна – студент, Санкт-Петербургский государственный университет; e-mail: st056063@student.spbu.ru, тел.: +7(906) 255-88-03

или модели в статье [4], рассматривающую похожую на SAPHIRE-модель модификацию SEIR-модели с целью оценки характеристик COVID-19 на основе европейских данных в начальный момент развития пандемии.

В данной работе параметры, которые бы давали приближение к реальным итальянским статистическим данным в период с 24 февраля 2020 года по 18 декабря 2020 года, определялись на временных интервалах в зависимости от существовавших эпидемических мер.

2. Формализация модели. Рассмотрим модель, которая основана на модели SQAIRD (S – susceptible, Q – quarantined, A – asymptomatic, I – infected, R – recovered, D – dead) из [5]. В ней популяция разделяется на шесть подгрупп, каждая из которых представляет собой долю по отношению к общей численности населения N , и где каждый человек относится лишь к одному подмножеству в один момент времени в зависимости от его состояния из возможных шести: S – восприимчивый к болезни, Q – на самоизоляции (в карантине), A – бессимптомно зараженный, I – зараженный и проявляющий симптомы, R – выздоровевший с приобретением иммунитета, D – умерший.

Основные положения. Принимаются два основных, характерных для моделей класса SEIR, предположения:

1. Предположение о неизменяемости численности населения, т. е. не рассматриваются естественный прирост населения и убыль по любым иным причинам, помимо смерти от болезни. Также популяция считается закрытой. Следовательно,

$$S(t) + Q(t) + A(t) + I(t) + R(t) + D(t) = 1 \quad \forall t, \quad (1)$$

где $S(t)$, $Q(t)$, $A(t)$, $I(t)$, $R(t)$, $D(t)$ – доли агентов популяции, находящихся в соответствующих подгруппах.

2. Предположение о равновероятном социальном взаимодействии среди населения, т. е. не рассматриваются отдельные социальные группы, где были бы различные вероятности контактов.

Была произведена модификация изначальной модели SQAIRD, а именно, введен параметр ω , позволивший переменной Q влиять на переменную S . Рассматриваемая модель описывается системой дифференциальных уравнений:

$$\begin{cases} \dot{S}(t) = -\beta S(A + I) - \gamma S + \omega Q, \\ \dot{Q}(t) = \gamma S - \omega Q, \\ \dot{A}(t) = \beta S(A + I) - \alpha k A - (1 - \alpha)\sigma A, \\ \dot{I}(t) = \alpha k A - (\sigma + \mu)I, \\ \dot{R}(t) = \sigma I + (1 - \alpha)\sigma A, \\ \dot{D}(t) = \mu I. \end{cases} \quad (2)$$

Для системы (2) были определены ее начальные данные и параметры. Также, основываясь на работе [4], был введен параметр удельной доли выявленных случаев, который дает некоторую оценку в сравнении выявленных случаев с результатами модели.

3. Оценка параметров. Параметры системы (2): β – вероятность заражения, γ – вероятность ухода на самоизоляцию, ω – вероятность выхода с самоизоляции, α – вероятность развития симптомов, k – характеристика вируса, показывающая вероятность развития симптомов при наличии вируса в организме, σ – вероятность выздоровления, μ – вероятность летального исхода. Значения всех параметров принадлежат промежутку $[0, 1]$.

Временные интервалы. Поскольку направленные на ограничение распространения вируса действия правительства не учитываются в системе дифференциальных уравнений (2), то разбиваем рассматриваемый период времени на шесть временных интервалов в зависимости от вводимых противокоронавирусных мер [6, 7] и адаптируем значения параметров под каждый промежуток.

Параметр β . В работе [8] рассмотрено множество модификаций SIR-моделей, а также представлена величина R_0 , которая в настоящий момент считается одной из основных характеристик эпидемического процесса – заразности заболевания. Она определяет среднее количество людей, которые будут заражены болеющим человеком, попавшим в окружение восприимчивых к данной болезни людей. Поскольку параметр β отвечает за вероятность заражения, то существует взаимосвязь между β и R_0 . В работе [9] показано, на основании чего была выведена зависимость β от R_0 для модели (2):

$$\begin{cases} \dot{A}(t) = \beta S(0)(A + I) - \alpha k A - (1 - \alpha)\sigma A = 0, \\ \dot{I}(t) = \alpha k A - (\sigma + \mu)I = 0, \end{cases}$$

где $S(0)$ – начальное состояние $S(t)$. Выразив $I = \frac{\alpha k A}{(\sigma + \mu)}$, найдем

$$\dot{A}(t) = \beta S(0) \left(A + \frac{\alpha k A}{(\sigma + \mu)} \right) - \alpha k A - (1 - \alpha)\sigma A = 0.$$

Преобразуя, имеем

$$\frac{\beta S(0) \left(1 + \frac{\alpha k}{(\sigma + \mu)} \right)}{\alpha k + (1 - \alpha)\sigma} = 1,$$

где $1 = R_0$. Таким образом, получаем

$$\beta = \frac{R_0(\alpha k + (1 - \alpha)\sigma)}{S(0) \left(1 + \frac{\alpha k}{\sigma + \mu} \right)}. \quad (3)$$

Так для параметра β установление значения свелось к определению величины R_0 , которая взята из работы [4]. В соответствии с принимаемыми противокоронавирусными мерами значение параметра β изменяется в случаях введения ограничения на передвижение, комендантского часа и закрытия заведений. Некоторые меры могут уменьшить значение параметра в 4 раза, как приведено в примере в [4].

Параметр γ . Величина, характеризующая долю соблюдающих карантин, была вычислена на основании исследования индекса самоизоляции, проведенной компанией Яндекс [11]. В соответствии с временными интервалами была получена усредненная оценка индекса самоизоляции на каждый интервал, рассматриваемый в модели.

Параметр ω . Значение параметра рассчитывалось на основании значения γ с учетом влияния изменения социального поведения в связи с эпидемиологической обстановкой – соблюдение социальной дистанции и гигиенических норм [12].

Параметр α . Выбор значения параметра α делался на основе оценки вероятности бессимптомного течения болезни [13].

Параметр σ . Значение данного параметра было выбрано как обратная величина к среднему времени выздоровления [9]. Информация по скорости выздоровления была взята из доклада [14].

Параметр k . Выбор значения параметра k , по аналогии со значением параметра σ , был связан с обратной величиной к среднему времени развития симптомов [14].

Параметр μ . Значение этого параметра основывается на вероятности, что болезнь перейдет в тяжелую стадию, и коэффициенте CFR (case fatality rate), значения которых взяты из доклада [14] с учетом удельной доли выявленных случаев [4]. Также было принято решение менять значение μ в зависимости от времени в связи с тем, что в начальный момент распространения эпидемии в Италии не было возможности в полной мере оказать медицинскую помощь всем нуждающимся. Однако спустя некоторое время ситуация с медицинским обеспечением улучшилась, что повлияло на показатель смертности от COVID-19.

Начальные условия. Статистические данные по Италии для сравнения с результатами моделирования были взяты из [10], где отсчет ведется с 24 февраля 2020 года, что повлияло на выбор начального момента времени и на значение начальных условий. По условию (1), начальные условия также подчиняются равенству $S(0) + Q(0) + A(0) + I(0) + R(0) + D(0) = 1$. Начальные условия были выбраны в соответствии со статистическими данными, пояснения требует лишь выбор начальных состояний для групп Q и A . Для группы самоизолированных было установлено $Q(0) = 0$ в связи с предположением, что на момент начала эпидемии доля самоизолированных была пренебрежительно мала в связи с недостаточной информированностью о вирусе. Для группы же бессимптомных выбор основывался на факте, что в целом бессимптомных в разы больше, чем людей с симптомами [14]. Помимо того, в силу ограниченных возможностей тестирования, в начале эпидемии тесты на коронавирус проводились в основном для попавших в больницу пациентов. Так как в модели уже учтена вероятность развития симптомов в качестве параметра, то используем эту вероятность вместе с оценкой средней длительности развития симптомов, что тоже является одним из параметров системы. По статистическим данным Италии на 24 февраля 2020 года было 229 подтвержденных случаев, а на 29 февраля 2020 года, т. е. через пять дней – оценка средней длительности проявления симптомов [14], – 1694 подтвержденных случая, откуда $(1694 - 229)/\alpha$

человек можно считать зараженными, но без выраженных симптомов.

4. Численное моделирование. В данном параграфе представлены значения параметров и результаты численного моделирования. Размер популяции $N = 60461828$ [10], удельная доля выявленных случаев равна 59% [4], коэффициент $R_0 = 3,79$ [4]. В начале эксперимента все начальные данные системы (2) на основании статистических данных [10] равны следующим значениям: $S(0) = 1 - (229 + 7)/N - 1465/(N \cdot 0,261)$; $A(0) = 1465/(N \cdot 0,261)$; $Q(0) = 0,0$; $I(0) = 229/N$; $R(0) = 0,0$; $D(0) = 7/N$. Значения параметров в начале эксперимента, когда не были приняты противокоронавирусные меры, установлены следующим образом: $\gamma = 0,0$; $\omega = 0,0$; $\alpha = 0,261$; $\sigma = 1/20$; $k = 1/6$; $\mu = 0,2 \cdot 0,038 \cdot 0,59 = 0,004484$; а β зависит от всех перечисленных параметров по формуле (3) – $\beta \approx 0,169559$. На рис. 1 виден результат численного моделирования при использовании только начальных значений параметров, что при сравнении со статистическими данными по заболеваемости и смертности показывает необходимость в разбиении рассматриваемого периода на временные интервалы и адаптации параметров к ним. В таблицах 1, 2 приведены разбиения с последующей настройкой значений параметров, описанной в пункте 3. На рис. 2 можно видеть как результат численного моделирования с учетом значений параметров из таблиц 1, 2 относительно совпадает со статистическими данными в первые две волны, а в дальнейший период времени расходится со статистическими данными, потому что параметры не были под него адаптированы. На рис. 3 дано сравнение представлений количества инфицированных людей с учетом удельной доли выявленных случаев и общим предполагаемым количеством заболевших.

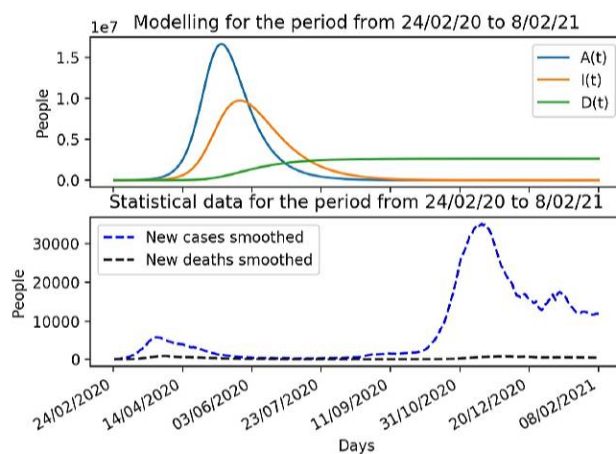


Рис. 1. Сравнение численного моделирования с заданными начальными параметрами со статистическими данными

Таблица 1. Заданные изменения значения параметров на трех временных интервалах в период с 10 марта 2020 года по 26 августа 2020 года

t	10 марта – 4 мая	5 мая – 18 июля	19 июля – 26 августа
β	$0,25\beta$	$0,5\beta$	$1,3\beta$
γ	$1 - 0,19$	$1 - 0,46$	$1 - 0,71$
ω	$0,7(1 - \gamma)$	$0,7(1 - \gamma)$	$0,7(1 - \gamma)$
μ	$0,004484$	$0,1\mu$	$0,1\mu$

Таблица 2. Заданные изменения значения параметров на двух временных интервалах в период с 27 августа 2020 гоа по 18 декабря 2020 года

t	27 августа – 19 ноября	20 ноября – 18 декабря
β	$0,5\beta$	$0,25\beta$
γ	$1 - 0,89$	$1 - 0,58$
ω	$0,7(1 - \gamma)$	$0,7(1 - \gamma)$
μ	$0,1\mu$	$0,1\mu$

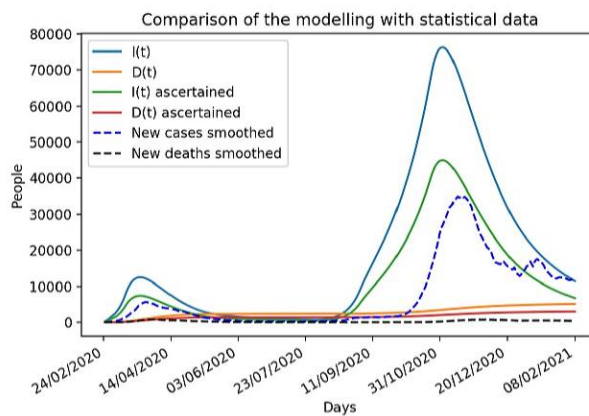


Рис. 2. Результат моделирования с описанными параметрами в сравнении со статистическими данными Италии по COVID-19 в период с 24 февраля 2020 года по 8 февраля 2021 года

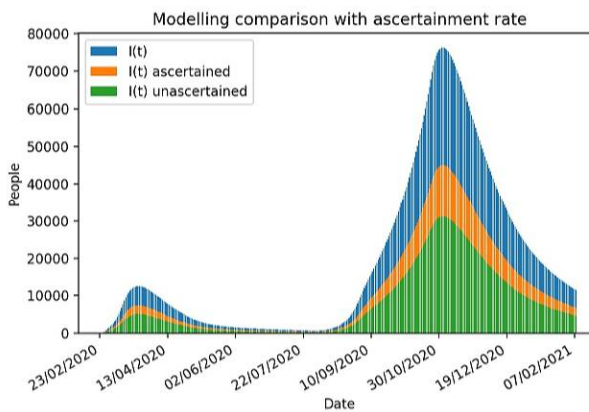


Рис. 3. Численное моделирование с учетом удельной доли выявленных случаев в период с 24 февраля 2020 года по 8 февраля 2021 года

5. Заключение. Изучены работы по исследованию компартментальных моделей, рассмотрены различные способы оценок параметров моделей распространения COVID-19 на примере нескольких стран. Проведено исследование SQAIRD-модели, в процессе которого выведена зависимость вероятности заражения от заразности заболевания для данной модели. Также осуществлена модификация рассматриваемой модели. Приведены результаты численного моделирования на примере статистических данных по Италии.

Литература

1. Kermack W.O., McKendrick A.G. A contribution to the mathematical theory of epidemics // Proceedings of the Royal Society of London A. 1927. Vol 115. P. 700–721.
2. Allen L.J.S. Stochastic population and epidemic models // Mathematical Biosciences Institute Lecture Series. Vol. 1.3: Stochastics in Biological Systems. 2015. P. 22–27.
3. Hao X., Cheng S., Wu D. et al. Reconstruction of the full transmission dynamics of COVID-19 in Wuhan // Nature. 2020. Vol. 584. P. 420–424.
4. Wang K., Ding L., Yan Y. et al. Modelling the initial epidemic trends of COVID-19 in Italy, Spain, Germany and France [Электронный ресурс]: URL:<https://doi.org/10.1371/journal.pone.0241743> (дата обращения: 29.01.21).
5. Gubar E., Policardo L. et al. Optimal lockdown policies driven by socioeconomic costs: targeted vs. uniform approaches [Электронный ресурс] // Journal of Political Economy. 2021. URL:<https://www.journals.uchicago.edu/toc/jpe/current> (дата обращения: 30.03.21).
6. Governo Italiano Presidenza del Consiglio dei Ministri [Электронный ресурс]: URL:<http://www.governo.it/node/14343> (дата обращения: 10.03.21).
7. Quotidiano Politico Economico Finanziario Normativo Il Sole 24 Ore [Электронный ресурс]: URL:<https://lab24.ilsole24ore.com/storia-coronavirus/> (дата обращения: 10.03.21).

8. Pastor-Satorras R., Castellano C. et al. Epidemic processes in complex networks [Электронный ресурс]: URL:<https://arxiv.org/abs/1408.2701> (дата обращения: 21.02.21).
9. Van den Driessche P. Reproduction numbers of infectious disease models // PubMed. 2017. Vol. 2. P. 288–303.
10. Our world in data [Электронный ресурс]: URL:<https://ourworldindata.org/> (дата обращения: 23.01.21).
11. Индекс самоизоляции [Электронный ресурс]: URL:<https://yandex.ru/company/researches/2020/back-to-life> (дата обращения: 15.03.21).
12. Gatto M., Bertuzzo E., Mari L. et al. Spread and dynamics of the COVID-19 epidemic in Italy: effects of emergency containment measures // Proceedings of the National Academy of Sciences. 2020. Vol. 117. P. 10484–10491.
13. Poletti P., Tirani M. et al. Probability of symptoms and critical disease after SARS-CoV-2 infection [Электронный ресурс]: URL:<https://arxiv.org/abs/2006.08471> (дата обращения: 14.02.21).
14. World Health Organization [Электронный ресурс]: URL:<https://www.who.int/docs/default-source/coronaviruse/who-china-joint-mission-on-covid-19-final-report.pdf> (дата обращения: 29.01.21).

УДК 004.855.5

Папернюк А. О.

Применение ALS в задаче коллаборативной фильтрации

*Рекомендовано к публикации старшим преподавателем
Кумачевой С. Ш.*

1. Введение. В наше время современному пользователю нелегко среди большого количества товаров и объектов найти вещь, необходимую ему. И тут на помощь приходят рекомендательные системы, которые на основе предпочтений пользователя предлагают наиболее подходящие товары.

2. Рекомендательные системы. В машинном обучении условно разделяют три принципа построения рекомендательных систем: коллаборативная фильтрация; методы, основанные на контенте; гибридные модели. Коллаборативная фильтрация представляет собой совокупность алгоритмов, учитывающих взаимодействия пользователей с объектами. Модели, основанные на контенте, используют дополнительное описание предпочтений, которое помогает находить объекты, схожие с понравившимися, и рекомендовать их. Гибридные модели объединяют два вышеупомянутых подхода.

3. Постановка задачи. Пусть имеется n пользователей и m музыкальных треков. Каждый пользователь прослушивает произвольное количество музыкальных композиций. Следовательно, формируется $(n \times m)$ -матрица R . Элемент $r_{u,i} \in [0, +\infty)$ этой матрицы соответствует количеству прослушиваний пользователя u трека i .

Данная работа посвящена практической реализации алгоритмов коллаборативной фильтрации и их сравнению с целью выявления модели, наилучшим образом соответствующей предъявляемым требованиям. В основу исследования легло задание, полученное при прохождении практики в отделе музыки компании ООО «Мейл.Ру». Цель исследования – реализация модели, которая для каждого пользователя рекомендует наиболее подходящие треки таким образом,

Папернюк Александр Олегович – студент, Санкт-Петербургский государственный университет; e-mail: st062261@student.spbu.ru, тел.: +7(981)108-53-07

чтобы они были персонализированными, т. е. подобранными исключительно по предпочтениям клиента.

В качестве выборочных данных были предоставлены две разреженные матрицы, каждая размерности: 6631477×119990 , где первое значение представляет количество пользователей, а второе – количество музыкальных треков. Первая матрица сформирована на конец сентября, и в ней содержится 370 357 786 ненулевых значений, а вторая – на конец октября, и в ней содержится 376 479 076 ненулевых значений. Посредством обнуления во второй матрице элементов, которые уже содержатся в первой, была сформирована третья матрица. Примем первую матрицу за тренировочные данные, а третью – за тестовые.

В данной статье рассматриваются две модели рекомендательной системы и приводится их сравнение.

4. Метод Alternative Least Square (ALS). Пусть $r_{u,i}$ – значения матрицы взаимодействий R . Введем индикаторную переменную $p_{u,i}$, которая указывает на предпочтение пользователя u объекта i . При $r_{u,i} > 0$ значение $p_{u,i} = 1$, если $r_{u,i} = 0$, то $p_{u,i} = 0$. Также введем переменную $c_{u,i} = r_{u,i} + 1$.

Задача алгоритма [1] – найти вектор $x_u \in \mathbb{R}^p$ для каждого пользователя и вектор $y_i \in \mathbb{R}^p$ для каждого объекта, который будет учитывать предпочтения пользователя. Иными словами, переменная предпочтений получается посредством векторного произведения: $p_{u,i} = x_u^T y_i$. Векторы пользователей и объектов подбираются таким образом, чтобы значение функции потерь было минимальным:

$$\min_{x_*, y_*} \sum_{u,i} c_{u,i} (p_{u,i} - x_u^T y_i)^2 + \lambda \left(\sum_u \|x_u\| + \sum_i \|y_i\| \right).$$

Член $\lambda(\sum_u \|x_u\| + \sum_i \|y_i\|)$ необходим для регуляризации модели, т. е. чтобы модель не переобучалась на тренировочных данных.

Пусть все векторы объектов можно брать из $(m \times f)$ -матрицы Y . Для каждого агента определим диагональную $(m \times m)$ -матрицу C^u , где $C_{ii}^u = c_{ui}$ и вектор $p(u) \in \mathbb{R}^m$, который содержит все предпочтения пользователя u .

Зафиксировав векторы объектов, дифференцируем функцию потерь по векторам пользователей x_u и приравниваем к нулю. Затем выразим эти векторы и получим уравнение

$$x_u = (Y^T C^u Y + \lambda I)^{-1} Y^T C^u p(u). \quad (1)$$

Для экономии большого количества времени приближенное решение уравнения (1) считается с помощью метода сопряженных градиентов [2].

Аналогично векторам объектов, определяем $(n \times f)$ -матрицу X , для каждого объекта задаем диагональную $(n \times n)$ -матрицу C_{uu}^i , где $C_{uu}^i = c_{ui}$ и вектор $p(i) \in \mathbb{R}^n$. Зафиксировав векторы пользователей, дифференцируем функцию потерь по векторам объектов y_i и полученное выражение приравняем к нулю. Выразив векторы объектов, получаем уравнение, аналогичное (1). Далее векторы объектов считаются с помощью метода сопряженных градиентов [2].

Поочередно фиксируя векторы и обновляя их, добиваемся сходимости метода. Главный плюс ALS – скорость и возможность легкого распараллеливания.

5. Рекомендации модели. После обучения ALS может составить список рекомендованных треков для каждого пользователя. Для этого берется вектор пользователя x_u^T и векторно домножается на векторы треков y_i . Далее формируется список из k треков, у которых результаты произведения оказались наибольшими. Этот список сортируется по убыванию результатов, и в итоге получаем треки, подобранные по предпочтениям клиента u . В случае, если мы считаем метрики на тестовых данных, то в рассмотрении не учитываются треки, которые уже были прослушаны пользователем, так как наша задача – предлагать пользователям треки, которые они не слушали.

6. Применение функции BM25. Для каждого элемента тренировочной матрицы применяется функция ранжирования BM25:

$$r_{u,i}^{new} = IDF(i) \frac{r_{u,i}(K_1 + 1)}{r_{u,i} + K_1 \left[(1 - b) + b \frac{|I_u|}{|I|} \right]},$$

где $|I_u|$ – общее количество взаимодействий клиента с разными объектами, $|I|$ – математическое ожидание величины $|I_u|$, K_1 и b – настраиваемые гиперпараметры, контролирующие количество прослушиваний трека пользователем и активность пользователя соответственно,

$$IDF(i) = \ln \frac{|U|}{|U_i|}.$$

Здесь $|U|$ – общее количество агентов, $|U_i|$ – количество агентов, которые взаимодействовали с объектом i .

После применения данной функции модель будет рекомендовать менее популярные, но более персонализированные треки пользователям. Итоговая весовая матрица подается на вход ALS-модели.

7. Используемые метрики. В данной работе использовались три метрики: HR, NDCG, APR. Пусть переменная rec_u обозначает множество объектов, рекомендуемых моделью, для пользователя u , и $real_u$ – это множество объектов, с которыми юзер u взаимодействует в тестовых данных.

Метрика HR (hit ratio) [3] вычисляется по формуле

$$HR@k = \sum_{u=1}^n \frac{rec_u \cap real_u}{\min(k, |real_u|)},$$

где $k = |rec_u| = \text{const}$ и $HR@k \in [0, 1]$, т. е. чем больше треков находятся на пересечении множеств rec_u и $real_u$ у пользователя, тем выше значение метрики.

Метрика NDCG [3] отвечает за то, насколько релевантные треки находятся наверху списка рекомендаций, т. е. предпочтительней, чтобы пользователи слушали треки, находящиеся наверху рекомендаций, нежели внизу. Обозначим за $rec_{u,i}$ трек, который рекомендуется пользователю u и находится на i -м месте в списке рекомендаций. Дополнительно обозначим величину

$$DCG@k = \sum_{i=1, j=1}^k \frac{I(rec_{u,i} \in real_u)}{\log_2(j+1)},$$

где I – это индикаторная функция. Также вводится переменная $IDCG$, значение которой соответствует максимально возможному значению DCG :

$$IDCG@k = \sum_{j=1}^{\min(k, |real_u|)} \frac{1}{\log_2(j+1)}.$$

И наконец, зададим метрику

$$NDCG@k = \frac{DCG@k}{IDCG@k},$$

при этом $NDCG \in [0, 1]$.

Метрика APR (average popularity rank) показывает, насколько наша модель рекомендует персонализированные (непопулярные) треки. Для данной метрики необходимо сформировать список треков, сортированный по убыванию количества прослушиваний, который назовем топ-чартом треков. Обозначим за top_i позицию i -го рекомендованного трека в топ-чарте. Тогда

$$APR@k = 1 - \frac{k(k+1)}{2 \sum_{i=1}^k top_i},$$

где $APR@k \in [0, 1)$. Чем ближе значение метрики к нулю, тем более популярные треки модель рекомендует, и наоборот.

8. Результаты. Для анализа результатов на языке Python в открытом приложении Jupyter Notebook реализована программа. В ней использовались такие библиотеки, как `scipy`, `implicit` и другие.

В итоге после подсчета метрик для двух моделей на тестовых данных, получаем результаты, представленные в таблице.

Таблица. Сравнение алгоритмов

№ модель	HR@10	NDCG@10	APR@10
ALS	0,036	0,0257	0,972
BM25 с ALS	0,0382	0,0273	0,9854
№ модель	HR@100	NDCG@100	APR@100
ALS	0,1545	0,0575	0,9413
BM25 с ALS	0,1723	0,0634	0,9677

Таким образом, алгоритм BM25 с ALS рекомендует более подходящие и персонализированные треки для пользователей.

Литература

1. Hu Y., Koren Y., Volinsky C. Collaborative filtering for implicit feedback datasets // Eighth IEEE International Conference on Data Mining. Pisa: IEEE. 2008. P. 263–272.
2. Takacs G., Pilaszy I., Tikk D. Applications of the conjugate gradient method for implicit feedback collaborative filtering // RecSys '11: Proceedings of the fifth ACM Conference on Recommender Systems. New York: Association for Computing Machinery. 2011. P. 297–300.

УДК 519.834

Савин К. А., Громова Е. В.

О свойствах характеристических функций в игре с многосторонними внешними влияниями

1. Введение. В статье изучается достаточно широкий класс однократных кооперативных игр с многосторонними внешними влияниями (multilateral externalities) [1]. Для данного класса игр построены α -, δ -, ζ - и η -характеристические функции [2, 3] и доказана теорема об их взаимосвязи. Кроме того, при некоторых дополнительных ограничениях исследуются значения векторов Шепли, построенных на основе различных характеристических функций (далее – х. ф.). Отметим, что ранее модель игры с отрицательными связями игроков в кооперативной постановке изучалась в работе [4], в которой исследовалось выполнение свойства супераддитивности для δ -х. ф. Тем не менее, в указанной работе не учитывалось ограничение на управление, что является стандартным требованием для данного класса задач. В настоящей статье управление предполагается принадлежащим компакту. Выполнение свойств характеристических функций, доказанных в настоящей статье, ранее было отмечено без доказательства в работе [3] для класса линейно-квадратичных кооперативных дифференциальных игр.

2. Постановка задачи. В этом разделе определим игру с многосторонними внешними влияниями и характеристическую функцию кооперативной игры.

2.1. Игра с многосторонними внешними влияниями. Рассмотрим однократную кооперативную игру в следующей формулировке [4]. Пусть $N = \{1, \dots, n\}$ – множество игроков. Под стратегией игрока i , $i = \overline{1, n}$, будем понимать выбор управляющего параметра e_i , $e_i \in [0, \bar{e}_i]$. Предполагается, что функция выигрыша

Савин Кирилл Алексеевич – студент, Санкт-Петербургский государственный университет; e-mail: st068146@student.spbu.ru, тел.: +7(930)271-72-58

Громова Екатерина Викторовна – профессор, Санкт-Петербургский государственный университет; e-mail: e.v.gromova@spbu.ru, тел.: +7(812)428-46-69

Работа выполнена при финансовой поддержке РФФ, грант 17-11-01079

$u_i(e_1, \dots, e_n)$ игрока i , $i = \overline{1, n}$, имеет следующий вид:

$$u_i(e_1, \dots, e_n) = f_i(e_i) - v_i\left(\sum_{i \in N} e_i\right), \quad (1)$$

где $f_i(e_i)$ – неотрицательная, строго возрастающая, дифференцируемая и строго вогнутая (выпуклая вверх) на $[0, \bar{e}_i]$ функция; $v_i(z)$ – неотрицательная, возрастающая, выпуклая (выпуклая вниз) функция на $[0, \sum_{i \in N} \bar{e}_i]$.

Достаточно часто введенный класс игр используется для описания процесса производства продукции, загрязняющей окружающую среду. Под стратегией игрока понимается выбор объема производства данной продукции, который прямо пропорционален объему вредных выбросов, поэтому под управлением e_i в данных задачах часто подразумевается объем вредных выбросов, хотя изначально речь идет об объемах производства, приносящих прибыль предприятию. Тогда $f_i(e_i)$ – функция прибыли от производства товаров, а $v_i(\sum_{i \in N} e_i)$ – функция затрат на устранение ущерба окружающей среде, аргументом которой является суммарный объем загрязнений.

В работе [1] для представленной модели некооперативной игры доказаны существование и единственность равновесия по Нэшу. Обозначим его как $(e_1^{NE}, \dots, e_n^{NE})$. Рассмотрим кооперативный вариант игры, в котором игроки будут объединяться в гранд-коалицию N для максимизации суммарного выигрыша. Очевидно, что в силу непрерывности и вогнутости функций $u_i(e_1, \dots, e_n)$, $i = \overline{1, n}$, на компакте $[0, \bar{e}_1] \times \dots \times [0, \bar{e}_n]$ имеем, что

$$\max_{e_i \in [0, \bar{e}_i]: i \in N} \sum_{i=1}^n u_i(e_1, \dots, e_n) = V(N)$$

будет достигаться в некоторой точке (e_1^*, \dots, e_n^*) .

2.2. Кооперативная игра в форме характеристической функции. Распределение заработанного игроками в кооперативной постановке игры суммарного максимального выигрыша $V(N)$ должно соответствовать тому или иному кооперативному решению (принципу оптимальности). В частности, может быть выбран множественный принцип оптимальности C -ядро [1], одноточечный вектор Шепли и др. [3]. В основном принципы оптимальности в кооперативных

играх основаны на вычислении значений так называемой характеристической функции для всевозможных допустимых коалиций игроков [4].

Приведем определения для нескольких вариантов построения характеристической функции, экономическая интерпретация которых подробнее описана в работе [3].

α -характеристическая функция определяется как нижнее значение игры между коалицией S и ее дополнением $N \setminus S$:

$$V^\alpha(S) = \begin{cases} 0, & S = \emptyset, \\ \max_{\substack{e_i \in [0, \bar{e}_i]: \\ i \in S}} \min_{\substack{e_j \in [0, \bar{e}_j]: \\ j \in N \setminus S}} \sum_{k \in S} u_k(e_1, \dots, e_n), & S \neq \emptyset, S \subset N, \\ V(N), & S = N. \end{cases}$$

Поскольку в рассматриваемой игре существует и единственно равновесие по Нэшу, то можно использовать также δ -х. ф.:

$$V^\delta(S) = \begin{cases} 0, & S = \emptyset, \\ \max_{\substack{e_i \in [0, \bar{e}_i]: \\ i \in S}} \sum_{k \in S} u_k(e_1, \dots, e_n) \Big|_{\substack{e_j = e_j^{NE}, \\ j \in N \setminus S}}, & S \neq \emptyset, S \subset N, \\ V(N), & S = N. \end{cases}$$

Выбрав точку максимума (e_1^*, \dots, e_n^*) , можно определить ζ -х. ф.:

$$V^\zeta(S) = \begin{cases} 0, & S = \emptyset, \\ \min_{\substack{e_j \in [0, \bar{e}_j]: \\ j \in N \setminus S}} \sum_{k \in S} u_k(e_1, \dots, e_n) \Big|_{e_i = e_i^*, i \in S}, & S \neq \emptyset, S \subset N, \\ V(N), & S = N. \end{cases}$$

Кроме того, в [3] предложен следующий способ построения η -х. ф.:

$$V^\eta(S) = \begin{cases} 0, & S = \emptyset, \\ \sum_{k \in S} u_k(e_1, \dots, e_n) \Big|_{\substack{e_i = e_i^*, i \in S \\ e_j = e_j^{NE}, j \in N \setminus S}}, & S \neq \emptyset, S \subset N, \\ V(N), & S = N. \end{cases}$$

3. Свойства характеристических функций. Пусть $S \subseteq N$. Для удобства введем обозначение:

$$e_S = \begin{cases} 0, & S = \emptyset, \\ \sum_{i \in S} e_i, & S \neq \emptyset, S \subset N. \end{cases}$$

Далее будем считать, что $S \neq \emptyset$. Запишем выражения для α -, δ -, ζ -, η - характеристических функций, принимая во внимание вид

функции полезности (1). Тогда имеем

$$\begin{aligned}
V^\alpha(S) &= \max_{e_i: i \in S} \min_{e_j: j \in N \setminus S} \sum_{k \in S} (f_k(e_k) - v_k(e_N)) = \\
&= \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) - \sum_{k \in S} v_k(e_S + \bar{e}_{N \setminus S}) \right), \\
V^\delta(S) &= \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) - \sum_{k \in S} v_k(e_S + e_{N \setminus S}^{NE}) \right), \\
V^\zeta(S) &= \min_{e_j: j \in N \setminus S} \left(\sum_{k \in S} f_k(e_k^*) - \sum_{k \in S} v_k(e_S^* + e_{N \setminus S}) \right) = \\
&= \sum_{k \in S} f_k(e_k^*) - \sum_{k \in S} v_k(e_S^* + \bar{e}_{N \setminus S}), \\
V^\eta(S) &= \sum_{k \in S} f_k(e_k^*) - \sum_{k \in S} v_k(e_S^* + e_{N \setminus S}^{NE}).
\end{aligned} \tag{2}$$

Теорема. Если $\forall k \in N$ функции v_k имеют вид $v_k(z) = a_k z + b_k$, то справедливо следующее равенство:

$$V^\alpha(S) - V^\delta(S) = V^\zeta(S) - V^\eta(S), \quad \forall S \subseteq N. \tag{3}$$

Доказательство. Пусть далее $S \neq \emptyset$. Доказательство для $S = \emptyset$ является тривиальным ($V(\emptyset) = 0$). В силу равенств (2) имеем

$$\begin{aligned}
&V^\alpha(S) - V^\delta(S) - (V^\zeta(S) - V^\eta(S)) = \\
&= \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) + \sum_{k \in S} (v_k(e_S^* + \bar{e}_{N \setminus S}) - v_k(e_S + \bar{e}_{N \setminus S})) \right) - \\
&- \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) + \sum_{k \in S} (v_k(e_S^* + e_{N \setminus S}^{NE}) - v_k(e_S + e_{N \setminus S}^{NE})) \right) = \\
&= \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) + \sum_{k \in S} (a_k(e_S^* - e_S)) \right) - \\
&- \max_{e_i: i \in S} \left(\sum_{k \in S} f_k(e_k) + \sum_{k \in S} (a_k(e_S^* - e_S)) \right) = 0.
\end{aligned}$$

Теорема доказана.

Пусть $Sh = \{Sh_i\}_{i=1}^n$ – вектор Шепли, построенный на основе одной из приведенных характеристических функций:

$$Sh_i = \sum_{S: i \in S} \xi(|S|) (V(S) - V(S \setminus \{i\})), \quad \xi(|S|) = \frac{(|S| - 1)!(n - |S|)!}{n!}.$$

Следствие. Пусть функции v_k имеют вид $v_k(z) = a_k z + b_k$, $\forall k \in N$. Тогда

$$Sh_i^\alpha - Sh_i^\delta = Sh_i^\zeta - Sh_i^\eta, \quad \forall i \in N,$$

где верхним индексом обозначен класс используемой характеристической функции.

Действительно, используя равенство (3), получаем

$$\begin{aligned} Sh_i^\alpha - Sh_i^\delta &= \\ &= \sum_{S:i \in S} \xi(|S|) ((V^\alpha(S) - V^\delta(S)) - (V^\alpha(S \setminus \{i\}) - V^\delta(S \setminus \{i\}))) = \\ &= \sum_{S:i \in S} \xi(|S|) ((V^\zeta(S) - V^\eta(S)) - (V^\zeta(S \setminus \{i\}) - V^\eta(S \setminus \{i\}))) = \\ &= Sh_i^\zeta - Sh_i^\eta, \end{aligned}$$

4. Заключение. В работе получены фундаментальные равенства, связывающие α -, δ -, ζ -, η - характеристические функции для одного достаточно широкого класса кооперативных игр.

Литература

1. Chander P., Tulkens H. The core of an economy with multilateral environmental externalities // International Journal of Game Theory. 1997. Vol. 26. P. 379–401.
2. Громова Е. В., Петросян Л. А. Об одном способе построения характеристической функции в кооперативных дифференциальных играх // Математическая теория игр и ее приложения. 2015. Т. 7. № 4. С. 19–39.
3. Gromova E., Marova E., Gromov D. A substitute for the classical Neumann–Morgenstern characteristic function in cooperative differential games // Journal of Dynamics and Games. 2020. Vol. 7. No 2. P. 105–122.
4. Reddy P. V., Zaccour G. A friendly computable characteristic function // Mathematical Social Sciences. 2016. Vol. 82. P. 18–25.

УДК 330.42

Сачков А. В., Щербаков Г. А.

Обсуждение потенциального применения аксиоматической теории выбора в бизнесе

Рекомендовано к публикации профессором Смирновым Н. В.

1. Введение. В данной статье рассмотрены возможности применения аксиоматической теории выбора в различных областях человеческой деятельности, в первую очередь связанных с бизнесом.

Задачи выбора являются одними из старейших в истории человечества. Этот факт вполне естественен, ведь необходимость выбрать то или иное направление действия постоянно встаёт перед каждым из нас.

Если же говорить о задаче выбора в чисто математическом смысле, то возникает некая проблема – недостаточная формализованность. Часто потенциальным решениям задачи ставят в соответствие некоторые числовые функции, называемые критериями, которые количественно оценивают решения на основе их качеств. Но как можно понять, какой из критериев важнее?

Обычно используется взвешенная сумма критериев – каждому из них приписывается вес (число от 0 до 1), символизирующий его важность. Затем каждый критерий домножается на свой вес и суммируется, «оптимальным» признаётся решение с наибольшей суммой. Такой подход, несмотря на свою простоту и интуитивную понятность, так и не был строго формализован.

Существует однако и чисто математическое решение – аксиоматическая теория выбора, которая была разработана на факультете ПМ–ПУ СПбГУ Ногиным В. Д. и Басковым О. В. Эта теория строится на небольшом количестве простых и в то же время естественных аксиом, учитывающих то, как люди принимают решения в реальной жизни, и позволяет учитывать предпочтения конкретного лица, принимающего решение (ЛПР).

Сачков Александр Валерьевич – студент, Санкт-Петербургский государственный университет; e-mail: st031354@student.spbu.ru, тел.: +7(812)428-41-54

Щербаков Глеб Александрович – студент, Санкт-Петербургский государственный университет; e-mail: st050604@student.spbu.ru, тел.: +7(812)428-41-54

В статье рассматривается, как данный подход можно использовать в страховании и актуарных расчётах, а также некоторые технические аспекты, возникающие в связи с применением аксиоматической теории выбора. В частности, «схема» программы сбора квантов, вычислительные алгоритмы и возможности их оптимизации.

2. Аксиоматическая теория выбора. В данном разделе рассмотрен аксиоматический подход в теории выбора [1].

Введём множество решений (альтернатив, вариантов) X произвольной природы (машины, недвижимост, смартфоны и пр.). Задача состоит в выборе наилучшего в некотором смысле элемента X .

Для сравнения решений введём критерии, оценивающие различные их качества. Строго говоря, критерий есть векторная функция

$$f : X \rightarrow \mathbb{R}^n,$$

где n – число параметров решения, каждый из которых представляется вещественным числом.

Также введём отношение предпочтения \succ_X на X , которое описывает предпочтения ЛПР, т. е. если ЛПР предпочитает решение $x_1 \in X$ решению $x_2 \in X$, то верно $x_1 \succ_X x_2$.

Теперь приведём постановку задачи многокритериального выбора: пусть дана тройка (X, f, \succ_X) , требуется найти $C(X)$ – множество выбираемых решений, которое представляет собой решение задачи выбора и содержит окончательно выбранные варианты.

Следует заметить, что произвольная природа элементов множества X затрудняет непосредственную работу с ними. Поэтому часто вводят множество возможных векторов $Y = f(X)$ – значения критериев всех решений, и \succ_Y – отношение предпочтения на Y , определяемое следующим образом:

$$f(x_1) = y_1 \succ_Y y_2 = f(x_2) \Leftrightarrow x_1 \succ_X x_2.$$

Можно переформулировать задачу многокритериального выбора соответственно: даны Y и \succ_Y , найти $C(Y)$.

Переходим к аксиоматичности, которая подкрепляет адекватность модели реальности, накладывая ограничения на отношение \succ_Y :

1. $y_1 \succ_Y y_2 \Leftrightarrow y_2 \notin C(Y)$;
2. отношение \succ_Y транзитивно;
3. отношение \succ_Y согласовано с каждым из критериев f_1, \dots, f_n .
Это означает, что $\forall i$ если y_1 и y_2 отличаются только в i -й компоненте и $y_1^i > y_2^i$, тогда $y_1 \succ_Y y_2$;
4. отношение \succ_Y линейно и однородно.

В случае выполнения вышеописанных аксиом можно ввести важное понятие – общий квант информации об отношении предпочтения ЛПР (для краткости просто квант). Будем использовать его упрощённую форму [1].

Пусть n обозначает число критериев, $I = \{1, \dots, n\}$ – множество номеров критериев.

Определение. Пусть $A, B \in I$, $A \neq \emptyset, B \neq \emptyset$. Говорят, что задан (общий) квант информации об отношении предпочтения ЛПР (квант) для групп критериев A, B вместе с наборами положительных параметров w_i^* для всех $i \in A$ и w_j^* для всех $j \in B$, если для вектора $y' \in \mathbb{R}^n$ вида

$$\begin{aligned} y'_i &= w_i^*, \quad i \in A, \\ y'_j &= -w_j^*, \quad j \in B, \\ y'_s &= 0, \quad s \in I \setminus \{A \cup B\}, \end{aligned}$$

выполняется $y' \succ_Y (0, \dots, 0) \in \mathbb{R}^n$.

Здесь используется пара подмножеств множества индексов, представляющих собой группы критериев, сравнительную важность которых показывает квант. Если $n = 2$, то получаем понятие элементарного, или простейшего, кванта.

Смысл введённого понятия нетрудно увидеть: ЛПР готово пожертвовать w_j^* количеством критериев f_j для всех $j \in B$, при этом получив взамен w_i^* единиц критерия f_i для всех $i \in A$. Такое определение предоставляет большую гибкость при выборе спектра решаемых задач.

Использование кванта связано с пересчётом критериев всех решений, при этом также происходит повышение размерности самого

критерия до $p = n - |B| + |A| \times |B|$. Нетрудно догадаться, что данное обстоятельство может привести к определённым проблемам в вычислительном плане при большом количестве решений.

Стоит также привести пример использования кванта для принятия решения в области страхования. Пусть имеется случайный риск X , который оценён экспертно с матожиданием $\mathbf{E}(X)$, страхователь (тот, кого страхуют), обладает капиталом ω , p есть премия – гарантированный убыток. Тогда страхователю предлагается для сравнения решения вида $(0, \mathbf{E}(X))$ и $(p, \mathbf{E}(X) + L)$, где L – общая сумма страховых выплат, которая может быть единовременной или растянутой во времени, условной и безусловной в зависимости от типа договора (полиса). Исходя из решения, принятого страхователем, можно делать рекомендации в подобных ситуациях в будущем.

3. Вычислительные аспекты. В статье [2] рассмотрены формулы для пересчёта решений при работе с общим квантом, которые предполагают умножение матриц с постоянно растущей размерностью. Как известно, операция умножения матриц является весьма затратной в вычислительном плане. В настоящее время достигнута сложность порядка $O(n^{2,37})$ [3], при этом известно, что она не может быть ниже $O(n^2)$, и требует некоторой степени оптимизации. В нашем случае специфические алгоритмы, применяемые, например, для разреженных матриц, неприменимы. Кроме того, особенности программы также требуют максимального понижения нагрузки на вычислительные устройства.

Рассмотрим два подхода. Первый заключается в дроблении множества решений на части и выбора наилучшего решения для каждой из них в отдельности. Нетрудно понять, что в данном случае возможны проблемы с корректностью (другими словами – соответствию аксиомам). Кроме того, при небольшом количестве решений могут возникать вопросы целесообразности подхода в целом.

Второй подход состоит в оптимизации процесса получения у ЛПР критериев, так как степень сужения в каждом случае может кардинально различаться – отсечение только одного решения или же сразу половины всех. Очевидно, что решение такой задачи довольно сложно с чисто теоретической точки зрения, поэтому предлагается рассмотреть эвристические идеи: можно предлагать для сравнения

решения y', y'' с максимальной абсолютной суммарной разницей по всем критериям:

$$\max_{y', y''} \rightarrow \sum_{i=1}^n |y'_i - y''_i|,$$

с максимальной усреднённой абсолютной суммарной разницей по всем критериям:

$$\max_{y', y''} \rightarrow \frac{1}{n} \sum_{i=1}^n |y'_i - y''_i|,$$

или же с максимальной абсолютной разницей по какому-либо одному критерию:

$$\max_{y', y''} \rightarrow |y'_i - y''_i| \text{ для какого-либо } i \in I.$$

4. Схема программы. Представим схему программы, предполагающуюся для автоматизации как сбора квантов у ЛПР, так и для предоставления рекомендаций на основе предпочтений самому ЛПР. Программа предполагается мобильной. Она состоит из:

1. Удалённого сервера, занимающегося основными вычислениями, агрегацией входных данных (путём парсинга сайтов-каталогов), их накоплением, хранением и обработкой.
2. Графической оболочки (GUI – Graphical user interface, графический пользовательский интерфейс), позволяющей пользователю легко получать нужные рекомендации.
3. Вычислительно-технического блока на устройстве пользователя, позволяющего получать рекомендации даже при отсутствии доступа к сети Интернет.
4. Прочих служб.

Именно тот факт, что вычисления предполагаются к проведению как на мощном удалённом сервере, так и на слабом устройстве пользователя, разница в производительности между которыми может составлять десятки тысяч раз, указывает на необходимость оптимизации соответствующих вычислений.

5. Заключение. В данной работе показано, что аксиоматический подход сочетает в себе полную математическую строгость и удобство в применении к реальным расчётам, при условии должной оптимизации. В дальнейшем предполагается рассмотреть альтернативные формулировки исходной задачи и точки соприкосновения с другими областями науки.

Литература

1. Ногин В. Д. Сужение множества Парето: аксиоматический подход. М.: Физматлит, 2018. 272 с.
2. Басков О. В. Алгоритм последовательного учета информации об относительной важности критериев в задаче многокритериального выбора // Процессы управления и устойчивость: Труды 41-й международной научной конференции аспирантов и студентов / под ред. Н. В. Смирнова, Г. Ш. Тамасяна. СПб.: Издат. Дом С.-Петербур. гос. ун-та, 2010. С. 553–558.
3. Википедия [Электронный ресурс]: URL:https://en.wikipedia.org/wiki/Matrix_multiplication_algorithm (дата обращения: 15.04.2021).

УДК 51-7

Соколова И. С.

Реализация алгоритмов решения задач многономенклатурной оптимизации управления запасами с дефицитом и скидками

Рекомендовано к публикации профессором Захаровым В. В.

1. Введение. Задача управления запасами состоит в нахождении оптимального уровня запасов, которые организация должна сохранять, чтобы получить максимальную прибыль. Запас был определен Дж. Монксом [1] как неиспользуемые ресурсы, которые имеют экономическую ценность. Целью управления запасами является минимизация затрат, связанных с хранением запасов, и максимизация прибыли в результате продажи. В соответствии с этим ставятся вопросы: когда следует размещать заказ и насколько большим он должен быть. В большинстве случаев обычно используются традиционные модели управления запасами, такие как (q, r) и (s, T) . Обе модели опираются на оценки издержек от различных видов взаимодействия с товаром и концентрируются на их минимизации. Базовая модель управления запасами (q, r) была разработана независимо Ф. Харрисом [2] и Р. Уилсоном [3], ими же была выведена формула для расчета показателей ЕОQ (Economic Order Quality), поэтому она известна как модель Харриса–Уилсона. В работе [4] проанализирована экономическая модель количества заказов и изучено ее применение на практике. В [5] показано, как оптимизировать затраты с помощью ЕОQ, а также разрешить конфликт между JIT (Just In Time) и ЕОQ. Существует множество модификаций формулы ЕОQ: формула Рыжикова [6] с учетом инфляции, Долгова [7] с учетом ограничения на вместимость склада и транспортных средств.

В данной статье представлено решение некоторых базовых моделей управления запасами. Рассматривается ЕОQ и ее модификации. Описаны базовые модели и их решение с использованием программы Excel. Разработанные программы удобны в использовании, с их помощью можно решать различные задачи, связанные с управлением запасами.

Соколова Инга Сергеевна – студент, Санкт-Петербургский государственный университет; e-mail: st051999@student.spbu.ru, тел.: +7(921)435-05-21

2. Модели и методы планирования и управления запасами. Наиболее распространенная ситуация, с которой сталкиваются производители и продавцы, заключается в том, что уровни запасов со временем истощаются, а затем пополняются новыми партиями товаров. Простая модель, описывающая данную ситуацию, это модель оптимального размера заказа – модель ЕОQ.

Предполагается, что единицы рассматриваемого товара изымаются из запасов непрерывно с известной постоянной скоростью D , т. е. спрос – это количество единиц товара на единицу времени. Кроме того, предполагается, что запасы пополняются, когда это необходимо, партией фиксированного размера (q единиц). Для описания ЕОQ необходимо учитывать затраты: c_0 – затраты оформления при заказе партии, c – цена производства или покупки единицы товара, c_h – расходы на хранение. Задача состоит в том, чтобы определить, когда и на сколько следует пополнять запасы, чтобы минимизировать сумму этих затрат. На рис. 1 изображен график уровня запасов с течением времени. В момент времени, когда уровень запасов подходит к черте минимума, заказывается партия из q единиц, уровень запасов становится q . Затем этот процесс повторяется каждый раз, когда уровень запасов снова падает до установленного уровня.

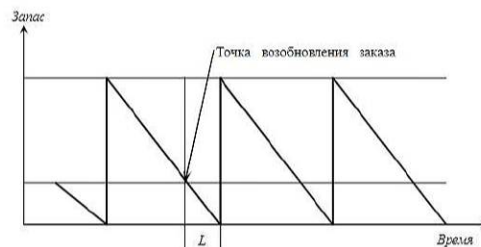


Рис. 1. График уровня запасов

3. Базовая модель ЕОQ. Введем следующие предположения. Пусть определена потребность в товарах на единицу времени и объем заказа q для пополнения запасов поступает, когда уровень запасов падает до определенной отметки.

Общие затраты T получаются из следующих компонентов:

- стоимости производства или заказа за цикл: $c_0 + cq$;
- среднего уровня запасов в течение цикла: $\frac{q}{2}$ и соответствующей стоимости: $\frac{c_h q}{2}$ за единицу времени;

- длины цикла: $\frac{q}{D}$;
- стоимости хранения за цикл: $\frac{c_h q^2}{2D}$;
- общей стоимости за цикл: $c_0 + cq + \frac{c_h q^2}{2D}$.

Таким образом, общая сумма расходов за единицу времени составляет

$$T = \frac{c_0 + cq + c_h \frac{q^2}{2D}}{\frac{q}{D}} = \frac{Dc_0}{q} + Dc + \frac{c_h q}{2}.$$

Значение q , которое минимизирует T , есть

$$q^* = \sqrt{\left(\frac{2Dc_0}{c_h}\right)}. \quad (1)$$

Формула (1) – это хорошо известная формула Уилсона–Харриса.

4. Некоторые модифицированные варианты ЕОQ. В этом разделе представлены две модификации ЕОQ – модели с дефицитом и со скидками, а также их объединение.

4.1. Детерминированная многономенклатурная модель управления запасами. Многие предприятия должны иметь дело одновременно с множеством продуктов. При поставке многономенклатурными партиями с длиной цикла (интервалом повторного заказа) τ размер партии продукта i в составе многономенклатурной партии $q_i = D_i \frac{\tau}{T}$ или, приняв $T = 1$, $q_i = \tau D_i$.

Тогда общая стоимость запаса

$$TC(\tau) = \sum_{i=1}^n c_i D_i + C^0 \frac{1}{\tau} + \sum_{i=1}^n c_i^h \frac{\tau D_i}{2}.$$

Оптимальная длина цикла $\tau^* = \sqrt{\frac{2c_0}{C^h D}}$. Оптимальный размер заказа продукта в составе многономенклатурной партии $q_i^* = \tau^* D_i$. Тогда общая стоимость запаса

$$TC(\tau^*) = \sum_{i=1}^n c_i D_i + C^0 \frac{1}{\tau^*} + \sum_{i=1}^n c_i^h \frac{q_i^*}{2}.$$

При поставках объемов q_i^* по параллельным каналам

$$TC(\overline{q_1^*, q_n^*}) = \sum_{i=1}^n c_i D_i + \sum_{i=1}^n c_i^0 \frac{D_i}{q_i^*} + \sum_{i=1}^n c_i^h \frac{q_i^*}{2}$$

имеем

$$C^0 = \sum_{i=1}^n c_i^0 \frac{\tau^* D_i}{q_i^*} = \sum_{i=1}^n c_i^0 \rightarrow C^0 = \sum_{i=1}^n c_i^0.$$

4.2. Детерминированная модель с учетом скидок на количество. В базовой модели предполагалось, что стоимость единицы товара одинакова независимо от количества в партии. Модель со скидками заменяет это предположение следующей модификацией: стоимость товара теперь зависит от его количества в партии. В частности, появляется стимул для заказа большей партии.

Если стоимость покупки единицы товара принимает значения

$$c = \begin{cases} c, & 0 \leq q < q_1, \\ c_i, & q_i \leq q < q_{i+1}, \quad i = \overline{1, k}, \\ c > c_i > c_{i+1}, & i = \overline{1, k-1}, \end{cases}$$

то общие затраты за единицу времени

$$T_i = \frac{Dc_0}{q} + Dc_i + \frac{c_h q}{2}, \quad i = \overline{1, k}.$$

4.3. Детерминированная модель с запланированным дефицитом. Одна из проблем материально-технического снабжения – возникновение нехватки товара. Модель с запланированным дефицитом решает эту проблему, достаточно только ввести новое предположение: когда возникает дефицит, покупатели ждут, пока продукт станет доступным. Их заказы выполняются сразу после поступления нужного количества товара на склад.

Пусть c_b – стоимость дефицита на единицу нехватки товара, $q - s$ – уровень запасов после добавления партии, s – уровень запасов перед добавлением партии. Общие затраты теперь получаются из следующих компонентов:

- стоимости производства или заказа за цикл: $c_0 + cq$;
- продолжительности периода отсутствия дефицита: $\frac{q-s}{D}$;
- продолжительности периода наличия дефицита: $\frac{s}{D}$;
- длины цикла: $\frac{q}{D}$;

- среднего уровня запасов: $\frac{(q-s)\frac{q-s}{D}}{2\frac{q}{D}} = \frac{(q-s)^2}{2q}$;
- среднего уровня дефицита: $\frac{s\frac{s}{D}}{2\frac{q}{D}} = \frac{s^2}{2q}$;
- затраты дефицита за цикл: $\frac{c_b s}{2} \frac{s}{q} = \frac{c_b s^2}{2q}$.

Таким образом, общие расходы за единицу времени

$$T = \frac{Dc_0}{q} + Dc + \frac{c_h(q-s)^2}{2q} + \frac{c_b s^2}{2q}.$$

В данной модели две переменные (s и q), оптимальные значения (s^* и q^*) находятся из равенств

$$\frac{\partial T}{\partial s} = \frac{(c_h + c_b)s}{q} - c_h = 0, \quad \frac{\partial T}{\partial q} = -\frac{c_0 D}{q^2} + \frac{c_h}{2} - \frac{(c_h + c_b)s^2}{2q^2} = 0,$$

$$s^* = \sqrt{\frac{2c_0 D}{c_b} \frac{c_h}{c_b + c_h}}, \quad q^* = \sqrt{\frac{2c_0 D}{c_h}} \sqrt{\frac{c_b + c_h}{c_b}}.$$

Оптимальная длина цикла

$$t^* = \frac{q^*}{D} = \sqrt{\frac{2c_0}{Dc_h}} \sqrt{\frac{c_b + c_h}{c_b}}.$$

5. Реализация моделей управления запасами в Excel. Все реализации представленных моделей находятся в открытом доступе [8] (см. рис. 2–4).

Входные данные						
	Товар 1	Товар 2	Товар 3	Размер заказа		
3	Цена покупки, c_p	2	3	4	0	200
4	Цена покупки, c_p	1,96	2,96	3,96	200	500
5	Цена покупки, c_p	1,92	2,92	3,92	500	
6	Стоимость хранения, c_{h1}	0,4	1,4	2,4		
7	Стоимость хранения, c_{h1}	0,392	1,392	2,392		
8	Стоимость хранения, c_{h1}	0,384	1,384	2,384		
9	Стоимость оформления, c^{MO}	20				
10	Количество рабочих дней	300				
11	Спрос, D	500	501	502		
Выходные данные						
	Товар 1	Товар 2	Товар 3			
14	Длина цикла, t	0,137809828244583				
17	Оптимальные размеры заказа, q^{MO}	224	120	91		
20	Оптимальные размеры заказа, с учетом диапазона	200	120	91		
23	Количество партий, p	41				
26	Ежегодные затраты хранения, $TC^{MO}(t)$	40	84	109,2		
29	Ежегодные затраты оформления	50	83,5	110,3297		
32	Ежегодные затраты на покупку	224	180	182		
37	Общие затраты	314	347,5	401,5297		
40	min затраты	314	347,5	401,5297		

Рис. 2. Модель уровня запасов со скидками на количество

Входные данные			
	Товар 1	Товар 2	Товар 3
Цена покупки, c_{po}	2	3	4
Стоимость хранения, c_{xh}	0,4	1,4	2,4
Стоимость оформления, c^{MO}	20		
Количество рабочих дней, w	300		
Спрос, D	500	501	502
Штраф за дефицит товара, c_{z}	1,5	2,5	3,5
Выходные данные			
	Товар 1	Товар 2	Товар 3
1 Длина цикла, t	0,173931326241553		
2 Оптимальные размеры заказа, q^{MO}	86,96566	87,13969	87,31353
3 Количество партий, p	52		
4 Размер дефицита, s	18,30856	31,28088	35,51737
5 Продолжительность периода отсутствия дефицита, t_1	0,137314	0,111494	0,10318
6 Продолжительность периода наличия дефицита, t_2	0,036617	0,062437	0,070752
7 Ежегодные затраты хранения, $T C^{MO}(t)$	17,39313	60,99772	104,7762
8 Общие затраты хранения	183,16707966498		
9 Общие затраты оформления	114,987911793556		
10 Ежегодные затраты на покупку	1000	1503	2008
11 Общие затраты на покупку	4511		
12 Ежегодные затраты дефицита и хранения	13,73142	39,1011	62,15539
13 Общие затраты дефицита и хранения	114,987911793556		
14 Общие затраты	1128,719	1657,089	2185,143
15 Затраты, $T C$	4740,97582368711		

Рис. 3. Модель уровня запасов с дефицитом

Входные данные				
	Товар 1	Товар 2	Товар 3	Размер заказа
Цена покупки, c_{po}	2	3	4	0, 200
Цена покупки, c_{po}	1,96	2,96	3,96	200, 500
Цена покупки, c_{po}	1,02	2,02	3,02	500
Стоимость хранения, c_{xh}	0,4	1,4	2,4	
Стоимость хранения, c_{xh}	0,392	1,392	2,392	
Стоимость хранения, c_{xh}	0,384	1,384	2,384	
Стоимость оформления, c^{MO}	20			
Количество рабочих дней	300			
Спрос, D	500	501	502	
Штраф за дефицит товара, c_{z}	1,5	2,5	3,5	
Выходные данные				
	Товар 1	Товар 2	Товар 3	
Длина цикла, t	0,137809828244683			
Оптимальные размеры заказа, q^{MO}	224	120	91	
Оптимальные размеры заказа, с учетом диапазона	200	120	91	
Количество партий, p	41			
Размер дефицита, s	14,5063	24,78457	28,1412	
Продолжительность периода отсутствия дефицита, t_1	0,370987	0,190051	0,12522	
Продолжительность периода наличия дефицита, t_2	0,029013	0,04947	0,05606	
Ежегодные затраты хранения, $T C^{MO}(t)$	40	64	109,2	
Ежегодные затраты оформления	50	83,5	110,33	
Ежегодные затраты на покупку	1000	1503	2008	
Ежегодные затраты дефицита и хранения	485,2656	58750,66	3446741	
Общие затраты	1535,266	60337,16	3448860	

Рис. 4. Модель уровня запасов с дефицитом и скидками

6. Выводы. В работе рассмотрены три модели управления запасами с учетом скидок, дефицита и объединения этих двух моделей, построенных на базе модели EOQ. Описан алгоритм, позволяющий за минимальное количество операций определить оптимальный объем партии, время между поставками, количество партий при наличии нескольких уровней цен на продукцию и дефицита. Приведены программы Excel, с помощью которых можно анализировать различные ситуации. Например, сравнивать существующую политику компании с примененной моделью управления запасами и ответить на вопросы: следует ли компании выбрать скидку, предлагаемую поставщиком; нужно ли произвести товар или передать его на аутсорсинг; должны ли компании идти на дефицит или нет.

Литература

1. Monks J. G. Operations Management. 3rd Edition, Theory and Problems. New York: McGraw-Hill Book Co., 1987. 719 p.
2. Harris F. W. Operations and cost // Factory Management. 1915. P. 48–52.
3. Wilson R. H. A Scientific routine for stock control // Harvard Business Review. 1934. P. 114–128.
4. Hadley G., Whitin T. M. Analysis of Inventory Systems. Englewood Cliffs: Prentice-Hall, 1963. 464 p.
5. Piasecki D. Inventory Management Explained. Saving warehouse space. Hawthorne: GlassFrogBooks, 2009. 341 p.
6. Рыжиков Ю. И. Теория очередей и управления запасами. СПб.: Питер, 2001. 376 с.
7. Долгов А. Я., Козлов В. К., Уваров С. А. Логистический менеджмент фирмы – концепция, методы и модели. Учебное пособие. СПб.: Бизнес-пресса, 2005. 384 с.
8. Реализация моделей управления запасами в Excel [Электронный ресурс]: URL:<https://github.com/Prokhorova2077/InventoryManagement> (дата обращения: 18.03.21).

УДК 519.177

Сташков Е. А.

Анализ задачи достижения консенсуса для ориентированных графов в дискретном времени

Рекомендовано к публикации доцентом Громовым Д. В.

1. Введение. В связи с активным развитием сетевых вычислительных систем в последние десятилетия большой интерес вызывает исследование динамики распределенных вычислений. Частным случаем такой задачи является проблема достижения консенсуса в распределенной системе. Следует отметить, что наряду с распределенными вычислениями, методы исследования динамики достижения консенсуса в сложных сетях активно изучаются в рамках теории сложных социальных систем [1].

В данной статье рассматривается модель достижения консенсуса в сетевой системе, структура взаимодействий в которой описывается направленным графом. Время в этой модели полагается дискретным. Показано, что скорость сходимости алгоритма определяется параметром ε , определяющим вклад состояний соседних вершин в алгоритме модификации состояния некоторой вершины. Определена верхняя граница на значение параметра ε , при которой алгоритм сходится.

2. Определения. Граф Γ описывается как упорядоченный набор $\Gamma = (V, E)$, где $V = \{1, \dots, n\}$ – множество вершин, $E \subset V \times V$ – множество ребер. Количество вершин и количество ребер обозначаются $N_V = |V|$ и $N_E = |E|$ соответственно. Ребро графа, соединяющее вершины j и i , обозначается (i, j) . Граф называется неориентированным, если набор (i, j) полагается неупорядоченным, т. е. $(i, j) \in E \Leftrightarrow (j, i) \in E$, и ориентированным (орграфом) в противном случае. Степень вершины неориентированного графа находится как количество ребер, инцидентных вершине. В случае ориентированного графа мы имеем входящую и исходящую степени вершины, которые равны количеству входящих и исходящих ребер соответственно.

Сташков Егор Алексеевич – студент, Санкт-Петербургский государственный университет; e-mail: st076827@student.spbu.ru, тел.: +7(911)405-42-93

2.1. Матричное представление графов. Матрица смежности – это матрица A размерности $[N_V \times N_V]$, в которой элемент $a_{ij} = 1$ при наличии ребра между вершинами i и j или $a_{ij} = 0$, если соответствующие вершины не связаны:

$$a_{ij} = \begin{cases} 1, & (i, j) \in E, \\ 0, & (i, j) \notin E. \end{cases}$$

Матрица Кирхгофа или лапласиан – матрица L размерности $[N_V \times N_V]$, которая для неориентированного графа определяется как

$$L = D - A,$$

где D – диагональная матрица, содержащая степени вершин d_i на диагонали. Если граф Γ ориентированный, то существует лапласиан $L^{out} = D^{out} - A$ по исходящим вершинам и лапласиан $L^{in} = D^{in} - A$ по входящим вершинам.

2.2. Связность графа. Ориентированным путем называется последовательность вершин и дуг $S = (i_1, (i_2, i_1), i_2, (i_3, i_2), \dots, i_k)$. Две вершины i' и i'' орграфа Γ называются связанными, обозначается $i' \rightsquigarrow i''$, если существует направленный путь, связывающий эти две вершины. Граф Γ называется сильно или ориентированно связным, если для любых двух вершин $i', i'' \in V$ выполняется $i' \rightsquigarrow i''$. В дальнейшем мы будем рассматривать только сильно связные графы.

3. Постановка задачи. Предположим, что мы имеем дело с распределенной системой, состоящей из n участников (агентов), причем каждый участник контактирует только с некоторым ограниченным набором своих соседей. Структура взаимодействий в такой системе представляется в виде информационного графа Γ и описывается матрицей смежности A . Если два участника могут обмениваться информацией в обоих направлениях, то информационный граф будет являться неориентированным, иначе – ориентированным.

В начальный момент времени i -й участник распределенной системы знает число $x_i(0)$ (например, полученное в результате измерения какой-то величины), которое может модифицироваться по некоторому правилу $x_i(k+1) = f_i(\mathbf{x}(k))$, где $x_i(k)$ – значение величины, известной i -му участнику распределенной системы в момент времени k , а вектор $\mathbf{x}(k) = \mathbf{x}$ – совокупность всех измерений участников в момент времени k . Необходимо найти такое правило, чтобы все

x_i приняли со временем одно и то же значение, в идеале – среднее арифметическое от начальных. Описанная задача имеет два аспекта: достижимость консенсуса – все x_i сходятся в пределе к одинаковому значению и несмещенность консенсуса – этот предел равен среднему значению $\frac{1}{n} \sum_{i=1}^n x_i(0)$. Мы будем рассматривать задачу достижения консенсуса:

$$\lim_{k \rightarrow \infty} x_i(k) = \bar{x} \quad \forall i = \overline{1, n}.$$

4. Консенсус в ориентированных графах. Предположим, что все участники используют следующую схему: на $(k + 1)$ -м шаге i -й участник вычисляет значение $x_i(k + 1)$ как значение $x_i(k)$ на предыдущем шаге минус корректирующая величина, которая определяется как взвешенная сумма разностей значения $x_i(k)$ и значений $x_j(k)$ участников, связанных с i -м участником. Формально это можно записать следующим образом:

$$x_i(k + 1) = x_i(k) - \varepsilon \sum_{j=1}^n a_{ij} (x_i(k) - x_j(k)), \quad (1)$$

где a_{ij} – это компоненты матрицы смежности, которые позволяют учесть тот факт, что участник i может контактировать только с ограниченным набором агентов. Выражение (1) может быть компактно переписано в векторно-матричной форме как

$$\mathbf{x}(k + 1) = (\mathbf{E} - \varepsilon \mathbf{L}^{in}) \mathbf{x}(k) = \mathbf{P} \mathbf{x}(k). \quad (2)$$

Одной из ключевых проблем при выборе схемы достижения консенсуса является определение значения веса ε . Решение этой задачи опирается на использование теорем Гершгорина [2] и Перрона – Фробениуса [3].

4.1. Условие достижения консенсуса. Выберем в качестве ε из (2) величину $\frac{1}{\Delta}$, где $\Delta > d_{max}^{in}$. Тогда матрица \mathbf{P} будет неотрицательна, т. е. каждый ее элемент будет неотрицательным. В результате получается схема

$$\mathbf{x}(k + 1) = \left(\mathbf{E} - \frac{1}{\Delta} \mathbf{L}^{in} \right) \mathbf{x}(k) = \mathbf{P} \mathbf{x}(k), \quad (3)$$

где L^{in} – матрица Кирхгофа по входящим вершинам, матрица

$$P = \begin{pmatrix} 1 - \frac{d_1^{in}}{\Delta} & \frac{a_{12}}{\Delta} & \dots & \frac{a_{1N_V}}{\Delta} \\ \frac{a_{21}}{\Delta} & 1 - \frac{d_2^{in}}{\Delta} & \dots & \frac{a_{2N_V}}{\Delta} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{a_{N_V 1}}{\Delta} & \frac{a_{N_V 1}}{\Delta} & \dots & 1 - \frac{d_{N_V}^{in}}{\Delta} \end{pmatrix}.$$

Матрица P – неотрицательная, примитивная (диагональные элементы ненулевые) и стохастическая по строкам. Из стохастичности следует, что она имеет правый собственный вектор $\mathbf{e} = (1, \dots, 1)$, соответствующий собственному числу $\sigma^* = 1$. С помощью теоремы Гершгорина можно показать, что все собственные числа λ_i матрицы P удовлетворяют неравенству $|\lambda_i| \leq 1$. Соответственно, спектральный радиус $\rho(P) = 1$. Матрица P удовлетворяет всем условиям теоремы Перрона – Фробениуса, значит

$$\lim_{k \rightarrow \infty} \frac{P^k}{\rho^k(P)} = \lim_{k \rightarrow \infty} P^k = \mathbf{v}\mathbf{w}^T,$$

где в качестве правого собственного вектора \mathbf{v} матрицы P , соответствующего $\sigma^* = 1$, можно взять вектор \mathbf{e} , а левый собственный вектор \mathbf{w} удовлетворяет условию нормировки $\mathbf{w}^T \mathbf{e} = 1$. Таким образом, консенсус достигается, однако условие несмещенности в общем случае не выполняется:

$$\lim_{k \rightarrow \infty} \mathbf{x}(k) = \lim_{k \rightarrow \infty} P^k \mathbf{x}(0) = \mathbf{e}\mathbf{w}^T \mathbf{x}(0) = (\mathbf{w}^T \mathbf{x}(0)) \cdot \mathbf{e} = \langle \mathbf{w}, \mathbf{x}(0) \rangle \cdot \mathbf{e}.$$

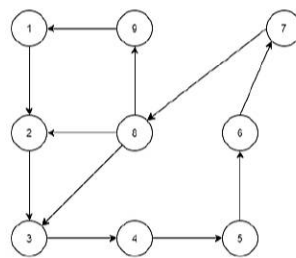


Рис. 1. Пример ориентированного графа

4.2. Результаты численного эксперимента. Для иллюстрации полученных теоретических результатов рассмотрим задачу достижения консенсуса применительно к связному ориентированному графу, изображенному на рис. 1.

Предположим, что начальные состояния равны $x_i = i - 1$, $i = \overline{1, 9}$. Продемонстрируем результат работы алгоритма (3) для значений $\varepsilon = \frac{1}{3}$ и $\varepsilon = \frac{1}{10}$. Все вычисления и построения графиков выполнены с помощью пакета MATLAB. Результаты численного моделирования представлены на рис. 2.

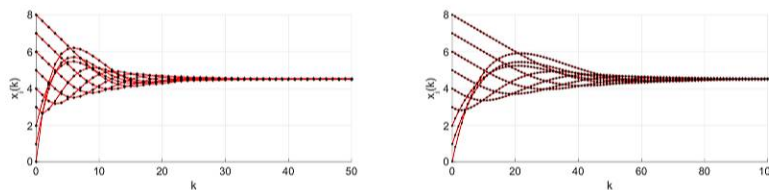


Рис. 2. Демонстрация сходимости алгоритма достижения консенсуса

Можно заметить, что для меньшего значения ε скорость сходимости существенно уменьшается. В то же время, увеличение ε выше теоретического порога может привести к появлению неустойчивости.

5. Заключение. В статье представлен алгоритм достижения консенсуса в распределенной системе, структура которой описывается направленным графом. Найдено условие на параметр ε , гарантирующее сходимость алгоритма. Полученные результаты проиллюстрированы с помощью численного моделирования.

Литература

1. Губанов Д. А., Новиков Д. А., Чхартишвили А. Г. Социальные сети: модели информационного влияния, управления и противоборства. М.: Физматлит, 2010. 228 с.
2. Воеводин В. В., Кузнецов Ю. А. Матрицы и вычисления. М.: Наука, 1984. 318 с.
3. Хорн Р., Джонсон Ч. Матричный анализ. М.: Мир, 1989. 655 с.

УДК 519.87

Третьякова С. С.

О структуре решения задачи оптимального управления объемами производства

Рекомендовано к публикации доцентом Громовым Д. В.

1. Введение. В классической теории оптимального управления, как правило, рассматриваются задачи без смены режима. Тем не менее в динамических задачах с множественными режимами возникают новые особенности, которых нет в стандартных постановках. В работе рассмотрена модель управления объемами производства [1], в которой в некоторый момент времени происходит изменение верхнего предела допустимого уровня загрязнений, создаваемых производством. Для данной задачи изучено поведение оптимальных управлений и оптимальных траекторий, с помощью подхода, основанного на расширении классического принципа максимума – гибридный принцип максимума, описанный в [2, 3].

2. Постановка задачи. Пусть имеется некоторое промышленное производство. Предполагается, что объем производства пропорционален загрязнениям. Количество выбросов в атмосферу, в свою очередь, ограничено сверху. Стратегия заключается в выборе управления для максимизации выигрыша.

Динамика изменения объема загрязнений описывается следующим дифференциальным уравнением:

$$\dot{x} = u, \quad u \in [0, b], \quad x(0) = x_0. \quad (1)$$

Интегральный выигрыш игрока имеет вид

$$J(x_0, T, u) = \int_0^T \left(\left(b - \frac{1}{2}u \right) u - dx \right) dt \rightarrow \max_u.$$

Предположим, что в момент времени $\tau_s \in [0, T]$ верхняя граница на управление меняется на \bar{b} , т. е. для $t \in [\tau_s, T]$ управление удовлетворяет ограничению $u(t) \in [0, \bar{b}]$.

Третьякова Софья Станиславовна – студент, Санкт-Петербургский государственный университет; e-mail: docsophy@gmail.com, тел.: +7(904)633-51-99

3. Общий подход к решению. Рассмотрим базовый случай, когда верхняя граница множества допустимых управлений не изменяется. Оптимальное управление и соответствующая ему траектория находятся с помощью принципа максимума Понтрягина.

Гамильтониан имеет вид

$$H = \left(b - \frac{u}{2}\right) u - dx(t) + \psi u,$$

где ψ – сопряженная переменная. Тогда

$$\frac{\partial H}{\partial u}(x, u, \psi) = b - u + \psi, \quad (2)$$

$$\frac{\partial^2 H}{\partial u^2}(x, u, \psi) = -1.$$

Это значит, что гамильтониан H выпуклый, следовательно, достигает своего максимума. Приравнявая равенство (2) к нулю, получаем выражение для оптимального управления

$$u^*(t) = b + \psi(t). \quad (3)$$

Каноническая система имеет вид

$$\begin{cases} \dot{x} = u^*, \\ \dot{\psi} = d. \end{cases} \quad (4)$$

Поскольку терминальное состояние $x(T)$ свободно, выполняется условие $\psi(T) = 0$. Решая второе уравнение системы (4), получаем $\psi = d(t - T)$, а следовательно, оптимальное управление (3) выражается как $u^*(t) = b + d(t - T)$. Очевидно, что оптимальное управление монотонно возрастает как функция от времени, принимая максимальное значение в точке $t = T$, $u^*(T) = b$, а для $t < T$, $u^*(t) < b$. С учетом нижнего ограничения окончательное выражение для оптимального управления принимает вид

$$u^*(t) = \begin{cases} 0, & t \in [0, T - \frac{b}{d}], \\ b + d(t - T), & t \in [T - \frac{b}{d}, T]. \end{cases}$$

4. Случай со сменой верхнего предела. Поскольку оптимальное управление не превышает b для любых значений $t \in [0, T]$,

увеличение верхней границы уровня выбросов $b \rightarrow \bar{b}$ не будет приводить к изменению оптимального управления. Однако, если на втором интервале ограничение на управление будет меньше, чем на первом, т. е. $(\bar{b} < b)$, то управление примет вид

$$u^*(t) = \begin{cases} b + d(t - T), & t \in [0, \theta), \\ \bar{b}, & t \in [\theta, T], \end{cases}$$

где $\theta \geq \tau_s$ определяется из условия $b + d(\theta - T) = \bar{b}$. Откуда имеем $\theta = T + \frac{\bar{b}-b}{d}$.

Положим $b + d(\tau_s - T) > 0$, т. е. на интервале $[\tau_s, T]$ оптимальное управление не достигает нижней границы. Рассмотрим две возможные ситуации:

1. *Владельцу предприятия неизвестно, что верхняя граница изменится.* Тогда он несет потери в размере $\int_{\theta}^T (b + d(t - T) - \bar{b}) dt$.

При таком исходе его поведение до момента смены условий ничем не будет отличаться от случая без изменения верхнего предела. Воспользовавшись этой логикой, можно найти составное уравнение оптимального управления. В конце первого участка в точке τ_s мы получим значение управления либо больше, чем новая граница, либо меньше. Исходя из этого, мы введем новую переменную ζ . Если значение $u^*(\tau_s) < \bar{b}$, то $\zeta = \theta = T + \frac{\bar{b}-b}{d}$. В противном случае, $\zeta = \tau_s$. Таким образом оптимальное управление

$$u_1^*(t) = \begin{cases} 0, & t \in [0, T - \frac{b}{d}], \\ b + d(t - T), & t \in (T - \frac{b}{d}, \zeta], \\ \bar{b}, & t \in (\zeta, T]. \end{cases} \quad (5)$$

Траектория, соответствующая этому управлению, изображена на рис. 1. Выражение для оптимальной траектории имеет вид

$$x_1^*(t) = \begin{cases} x_0, & t \in [0, T - \frac{b}{d}], \\ (b - dT)t + \frac{d}{2}t^2 + C_1, & t \in [T - \frac{b}{d}, \zeta], \\ \bar{b}t + C_2, & t \in (\zeta, T], \end{cases}$$

где $C_1 = x_0 - bT + \frac{b^2}{2d} + \frac{dT^2}{2}$ и $C_2 = C_1 + (b - dT)\zeta + \frac{d}{2}\zeta^2 - \bar{b}\zeta$.

Возьмем набор переменных $T = 25$, $b = 20$, $d = 1$, $x_0 = 10$, $\bar{b} = 5$, $\tau_s = 15$ и построим график. На рис. 1 видно, что поскольку в момент

τ_s управление $u^*(\tau_s) > \bar{b}$, то управление претерпевает скачок, чтобы соответствовать установленному ограничению.

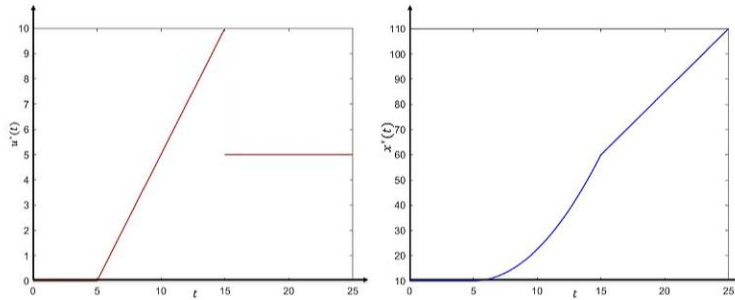


Рис. 1. Оптимальное управление $u_1^*(t)$ и траектория $x_1^*(t)$

2. Владелец предприятия знает о грядущем изменении $b \rightarrow \bar{b}$. Для вычисления оптимального управления необходимо рассмотреть задачу оптимального управления отдельно на двух интервалах, начиная со второго. На интервале $[\tau_s, T]$ определим оптимальное значение функционала

$$J_2(x, u) = \int_{\tau_s}^T \left(\left(b - \frac{1}{2}u \right) u - dx \right) dt,$$

где $u(t)$ определяется как сужение (5) на исследуемый интервал, а $x(t)$ находится из решения (1) с начальным условием $x(\tau_s) = x_s$.

Получим значение выигрыша игрока на втором интервале J_2^* , которое при фиксированном τ_s будет зависеть только от x_s . Теперь рассмотрим первый интервал $[0, \tau_s]$. В момент времени τ_s конец траектории не будет фиксирован, но так как необходимо учитывать выигрыш на втором участке, рассмотрим общий выигрыш в смешанной форме (интегральный и терминальный в момент времени τ_s):

$$J(x_0, T, u) = \int_0^{\tau_s} \left(\left(b - \frac{1}{2}u \right) u - dx \right) dt + J_2^*(x_s) \rightarrow \max.$$

Для того чтобы записать сопряженные уравнения и условия

трансверсальности, найдем производную терминального выигрыша

$$\frac{\partial J_2^*}{\partial x_s} = 2(b - \bar{b}) - 2d(T - \tau_s).$$

Используя условие трансверсальности в задаче с терминальным выигрышем, получаем значение ψ на конце первого интервала: $\psi(\tau_s) = 2(b - \bar{b}) - 2d(T - \tau_s)$.

Составное уравнение управления в этом случае запишется следующим образом:

$$u_2^*(t) = \begin{cases} 0, & t \in [0, \theta - \frac{\bar{b}}{d}], \\ 3b - 2\bar{b} + d(t - 2T + \tau_s), & t \in (\theta - \frac{\bar{b}}{d}, \theta], \\ \bar{b}, & t \in (\theta, T], \end{cases}$$

где $\theta = 2T - \tau_s + \frac{3\bar{b} - 3b}{d}$.

Решая (1), найдем оптимальную траекторию

$$x_2^*(t) = \begin{cases} x_0, & t \in [0, \theta - \frac{\bar{b}}{d}], \\ (3b - 2(\bar{b} + dT) + d\tau_s)t + \frac{d}{2}t^2 + C_1, & t \in (\theta - \frac{\bar{b}}{d}, \theta], \\ \bar{b}t + C_2, & t \in (\theta, T]. \end{cases}$$

Подводя итоги, возьмем набор переменных $T = 25, b = 20, d = 1, x_0 = 100, \bar{b} = 10, \tau_s = 10$ и совместим графики двух случаев, чтобы наглядно увидеть разницу. На рис. 2 пунктирной линией отмечен график случая, в котором грядущие изменения были ожидаемы, а сплошной, соответственно, график случая, когда неизвестно об изменении верхней границы.

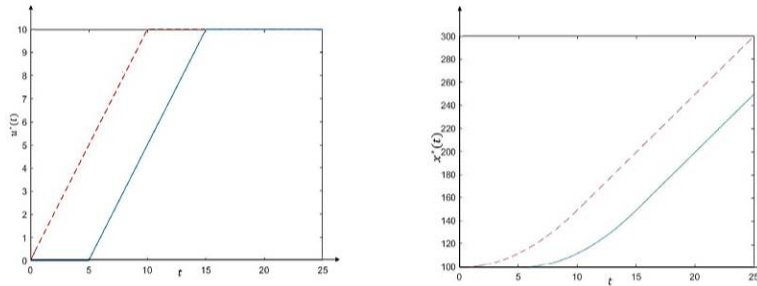


Рис. 2. Сравнение оптимальных управлений u_i^* и оптимальных траекторий x_i^*

Заметим, что в ситуации, когда заранее известно о смене верхней границы выбросов, оптимальное управление быстрее выходит на нужный максимально допустимый уровень, а траектория достигает более высокого значения, чем в ситуации, когда ориентируются только на данные, которые имеются в настоящий момент.

5. Заключение. В данной работе рассмотрены два случая изменения верхнего ограничения допустимого уровня выбросов, а также изучены поведения оптимальных управлений и оптимальных траекторий.

Литература

1. Gromova E. The Shapley value as a sustainable cooperative solution in differential games of three players // *Recent Advances in Game Theory and Applications, Static and Dynamic Game Theory: Foundations and Applications*. 2016. P. 67–91.
2. Bondarev A., Gromov D. On the structure and regularity of optimal solutions in a differential game with regime switching and spillovers // *Dynamic Economic Problems with Regime Switches. Dynamic Modeling and Econometrics in Economics and Finance*. Springer. 2021. Vol. 25. P. 187–208.
3. Gromov D., Gromova E. On a class of hybrid differential games // *Dynamic Games and Applications*. Springer. 2017. Vol. 7. No 2. P. 266–288.

УДК 517.977.8

Чеботарева А. А., Павлова Е. Д., Цепелева Р. В.

Линейно-квадратичная дифференциальная игра управления объемами вредных выбросов с функционалом в форме Больца

Рекомендовано к публикации профессором Громовой Е. В.

1. Введение. В работе изучается одна модель управления объемами вредных выбросов, в которой, в отличие от [1, 2], выигрыш является смешанным, т. е. имеет не только интегральную, но и терминальную часть, соответствующую штрафам за загрязнение среды в последний момент игры. Изучается, как величина штрафа влияет на оптимальные управления игроков, а также вводятся две новые для дифференциальных игр числовые характеристики, позволяющие проанализировать потери игроков в случае отказа от кооперации и в случае ошибочного представления о параметрах игры.

2. Постановка задачи. Рассмотрим теоретико-игровую модель управления объемами вредных выбросов, основанную на работе [1]. Пусть в игре участвуют $n = 3$ игрока. Подразумевается, что у каждого игрока есть промышленное производство на своей территории. Объемы производства прямо пропорциональны объемам загрязнения окружающей среды. Под стратегией игрока i понимается выбор скорости загрязнения в единицу времени, $u_i \in [0, b_i]$.

Общий уровень загрязнения $x(t)$ изменяется в соответствии с дифференциальным уравнением

$$\dot{x} = u_1 + u_2 + u_3, \quad x(t_0) = x_0. \quad (1)$$

Игра начинается в момент t_0 из состояния x_0 и заканчивается в момент T . Будем решать задачу в классе программных управлений $u_i(t)$, где $t \in [t_0, T]$. Предположим, что в терминальный момент иг-

Чеботарева Ангелина Андреевна – студент, Санкт-Петербургский государственный университет; e-mail: angel.xe99@gmail.com, тел.: +7(981)773-93-81

Павлова Екатерина Денисовна – студент, Санкт-Петербургский государственный университет; e-mail: e.d.pavlova@spbu.ru, тел.: +7(999)724-39-74

Цепелева Рита Вячеславовна – студент, Санкт-Петербургский государственный университет; e-mail: st062153@student.spbu.ru, тел.: +7(981)698-55-88

ры игроки выплачивают штраф, прямо пропорциональный уровню загрязнений $x(T)$. Тогда выигрыш игрока i , $i = \overline{1, 3}$, представляет собой смешанный функционал Больца

$$K_i(x_0, T - t_0, u) = \int_{t_0}^T \left(\left(b_i - \frac{1}{2} u_i(t) \right) u_i(t) - d_i x \right) dt - D_i x(T), \quad (2)$$

где $u = \{u_1, u_2, u_3\}$, $b_i \geq 0$ – коэффициент, равный отношению общего дохода от производства i -й компанией к объему общего загрязнения соответствующей компании; $d_i \geq 0$ – расходы i -го игрока на устранение единицы общего загрязнения; $D_i \geq 0$ – величина штрафа на единицу загрязнений. Далее будем предполагать, что $t_0 = 0$.

3. Кооперативный и некооперативный вариант игры. Игроки изначально договариваются об использовании таких оптимальных управлений, которые будут максимизировать суммарный выигрыш

$$\sum_{i=1}^3 K_i(x_0, T, u) \rightarrow \max_u, \quad (3)$$

так что $x(t)$ подчиняется (1). Управления, максимизирующие (3), обозначим $u^{\{1,2,3\}}(t) = \{u_1^{\{1,2,3\}}(t), u_2^{\{1,2,3\}}(t), u_3^{\{1,2,3\}}(t)\}$, а соответствующую им траекторию – $x^{\{1,2,3\}}(t)$. Ранее доказано [3], что для линейно-квадратичных задач типа (1), (2) такое решение в классе программных стратегий существует и единственно.

Введем систему обозначений:

$$\begin{aligned} b_{123} &= b_1 + b_2 + b_3, & D_{123} &= D_1 + D_2 + D_3, \\ d_{123} &= d_1 + d_2 + d_3, & D_{ij} &= D_i + D_j. \end{aligned}$$

Нетрудно проверить, что согласно принципу максимума Понтрягина [4] оптимальные управления будут иметь вид

$$u_i^{\{1,2,3\}}(t) = b_i - (T - t) d_{123} - D_{123}, \quad i = 1, 2, 3.$$

Введем дополнительные ограничения на параметры модели, гарантирующие, что полученные здесь и далее для некоторых других вспомогательных задач оптимизации управления являются допусти-

мыми и принадлежат компакту $[0, b_i]$:

$$D_i \in [0, \min(b_1 - D_{123}, b_2 - D_{123}, b_3 - D_{123})],$$

$$d_i \in [0; \frac{\min(b_1, b_2, b_3) - D_{123}}{T} - d_{123}], \quad i = 1, 2, 3.$$

Для некооперативного варианта игры аналогично можно вычислить равновесие по Нэшу при помощи принципа максимума Понтрягина. Согласно [3] для класса линейно-квадратичных игр оно существует и единственно. Соответствующие равновесию по Нэшу управления имеют вид

$$u_i^{NE}(t) = b_i - (T - t) d_i - D_i, \quad i = 1, 2, 3.$$

Аналогично находим траекторию, подставляя полученные управления в (1). Теперь вычислим значение выигрыша игрока $i = 1$ в равновесии по Нэшу, подставив полученные управления в (2):

$$K_1^{NE} = K_1(x_0, T, u^{NE}) =$$

$$= -x_0(Td_1 + D_1) + D_1T(D_{123} - b_{123} + Td_{123}) + \frac{T(b_1^2 - D_1^2)}{2} + (4)$$

$$+ \frac{T^2(d_1D_{23} - d_1b_{123} - D_1d_{123})}{2} + \frac{T^3(d_1^2 + 2d_1d_2 + 2d_1d_3)}{6}.$$

Для остальных игроков $i = 2, 3$ значение выигрышей в равновесии по Нэшу получается циклической перестановкой индексов в (4).

4. Значение информации. Одной из характеристик важности владения верной информацией о параметрах модели является так называемое значение информации (value of information [5]). Не вдаваясь в стохастический контекст данного понятия, вычислим потери игроков в некооперативном варианте игры для случая, когда они ошиблись относительно условий, в которых развивается конфликтно-управляемый процесс. Предположим, что игроки не знают, что в последний момент игры им будет назначен терминальный штраф $D_i x(T)$, т. е. в терминах нашей игры они пред-

полагают, что $D_i = 0$, хотя на самом деле это не так. Тогда функционал выигрыша для каждого игрока i , $i = \overline{1, 3}$, имеет вид

$$\overline{K}_i(x_0, T, u) = \int_{t_0}^T \left(\left(b_i - \frac{1}{2} u_i(t) \right) u_i(t) - d_i x \right) dt. \quad (5)$$

Тогда игроки будут использовать следующие управления в равновесии по Нэшу:

$$u_i^{*NE}(t) = b_i - (T - t) d_i, \quad i = 1, 2, 3.$$

Поскольку на самом деле функционалом выигрыша игроков является не (5), а (2), подставим полученные управления в (2) и вычислим значение этого функционала:

$$\begin{aligned} K_1(x_0, T, u^{*NE}) &= \frac{T^3 d_1 (d_{123} + d_{23})}{6} + \\ &+ \frac{T^2 (D_1 d_{123} - d_1 b_{123})}{2} - T \left(D_1 b_{123} + d_1 x_0 - \frac{b_1^2}{2} \right) - D_1 x_0. \end{aligned}$$

Для того чтобы оценить характеристику «значение информации», рассмотрим разность:

$$\begin{aligned} VI &= K_1(x_0, T, u^{NE}) - K_1(x_0, T, u^{*NE}) = \\ &= \frac{T (D_1 (D_{123} + D_{23}) + T d_1 D_{23})}{2}. \quad (6) \end{aligned}$$

5. Ценность кооперации. Проанализируем ситуацию, в которой игрок отказывается от участия в коалициях для максимизации общего выигрыша (3). Вычислим значение выражения $K_1^{NE} - Sh_1$, где компонента вектора Шепли, соответствующая компоненте дельта игрока 1, есть величина

$$\begin{aligned} Sh_1(x_0, T) &= \frac{1}{3} [V(\cdot, \{1, 2, 3\}) - V(\cdot, \{2, 3\}) + V(\cdot, \{1\}) + \\ &+ \frac{1}{6} [V(\cdot, \{1, 2\}) - V(\cdot, \{2\}) + V(\cdot, \{1, 3\}) - V(\cdot, \{3\})]. \end{aligned}$$

Тогда значение потери в случае отказа от кооперации для игрока 1

имеет вид

$$\begin{aligned}
 VC = K_1^{NE} - Sh_1 = & \left(\frac{D_3}{3} + \frac{Td_3}{3} \right) x_0 + \frac{D_3 T b_1}{3} - \frac{T b_3^2}{6} - \frac{T^3 d_3^2}{18} - \\
 & - \frac{T^3 d_{12} d_3}{9} + \frac{T^2 (b_{123} d_3 - d_3 D_{123} - D_3 d_{12})}{6} - \frac{T D_3 (D_{123} + D_{12} + b_{23})}{3}.
 \end{aligned} \tag{7}$$

Сравнение введенных в (6), (7) характеристик VI , VC может быть проведено на основе фиксирования отдельных параметров модели и вариации остальных.

6. Заключение. В работе введены две новые характеристики для дифференциальных игр, отвечающие за потери в случае плохой информированности игроков и в случае отказа от кооперации. Данные характеристики вычислены на примере линейно-квадратичной дифференциальной игры с функционалом в виде Больца.

Литература

1. Gromova E. The Shapley value as a sustainable cooperative solution in differential games of three players // Recent Advances in Game Theory and Applications, Static and Dynamic Game Theory: Foundations and Applications. 2016. P. 67–91.
2. Барсук П. И., Громова Е. В. Исследование свойств одного кооперативного решения в дифференциальной игре управления вредными выбросами // Процессы управления и устойчивость. 2020. Т. 7. № 1. С. 361–366.
3. Basar T., Olsder G. J. Dynamic noncooperative game theory. Publisher: Society for Industrial and Applied Mathematics, 1999. 519 p.
4. Понтрягин Л. С. Математическая теория оптимальных процессов. М.: Наука, 1961. 392 с.
5. Raiffa H., Schlaifer R. Applied Statistical Decision Theory. New York: Wiley, 1961. 356 p.

UDC 519.83

Dan W.

Analysis of static game and repeated game in tax collection and management

Recommended for publication by senior lecturer Kumacheva S. Sh.

1. Introduction. In the process of tax collection and management, the tax authority should reduce tax evasion by the taxpayer, and the taxpayer will choose to evade tax if they have the opportunity. It forms a zero-sum game with perfect competition [1]. Xuehan Leng (2012) analyzed the optimal inspection probability of the tax authority in a repeated game [2]. It is inherently not conducive to maximizing social welfare. Thus, after the tax authority rewards non-tax evaders g , we consider the static game and repeated game's perspective, and analyse the behavior of tax authority and the taxpayer, and propose corresponding measures to reduce the probability of tax evasion.

2. The model. In the game, the tax authority is regarded as player 1 and the taxpayer is regarded as player 2. There are two strategies to choose, player 1 may perform the inspection or not; player 2 may evade tax or not. Define $G = (X_i, Y_i, U)$, X_i and Y_i represent the i -th strategies of players, $i = 1, 2$, where $X = (X_1, X_2) = (\text{inspect}, \text{non-inspect})$, $Y = (Y_1, Y_2) = (\text{tax evasion}, \text{non-tax evasion})$, $U = (U_1, U_2)$, where U_1, U_2 represent the payoff of player 1 and player 2 respectively. The following model assumptions are made.

- **Assumption 1.** Player 2's pre-tax salary is Y which is accounting for profit. The tax rate is r . Assuming tax is levied at flat rate unified tax rate based on a percentage, tax payable of player 2 is rY , and tax evasion amount of player 2 is $a \in [0, rY]$. The cost avoid paying tax of player 2 is b , $0 \leq b < a$, otherwise, according to the principle of rational people who follow the maximize benefits, player 2 would rather not choose to evade tax.
- **Assumption 2.** If player 1 finds tax evasion during the inspection process, it will impose a certain fine. f is the ratio of the amount of

Dan Wang – undergraduate student, Yanan University;
e-mail: 2107683623@qq.com, phone: +8615319428312

finer imposed by player 1 to the amount of tax evasion, the amount of fines imposed by player 1 after discovering tax evasion is af [3].

- **Assumption 3.** If there is the behavior of tax evasion of player 2, player 1 can always find out it. The inspection cost of player 1 is c , $0 \leq c < a + af$ and $0 \leq c \leq b$, otherwise, according to the principle of rational people, player 1 will choose not to inspect.
- **Assumption 4.** Player 1 awards player 2 who do not evade tax, denoted by q which counts tangible or intangible rewards.

Table 1. Payoff matrix for players

	Y_1	Y_2
X_1	$(rY + af - c, Y - rY - af - b)$	$(rY - c - q, Y - rY + q)$
X_2	$(rY - a, Y - (rY - a) - b)$	$(rY, Y - rY)$

In the payoff matrix, there is no pure-strategy Nash equilibrium.

3. Taxation analysis in static game. According to the existence theorem of Nash equilibrium [4], there must be a mixed-strategy Nash equilibrium.

Let the mixed-strategy of two players be $H = (p, 1 - p)$, $p \in [0, 1]$; $J = (e, 1 - e)$, $e \in [0, 1]$. The probability that player 1 chooses to inspect is p , the probability of non-inspecting is $1 - p$, the probability of player 2 evasion is e , and the probability of not evading tax is $1 - e$.

- The expected payoff of evaded tax of player 2:
 $U_e = p(Y - rY - af - b) + (1 - p)(Y - rY + a - b)$
- The expected payoff of non-tax evasion of player 2:
 $U_{1-e} = p(Y - rY + q) + (1 - p)(Y - rY)$
- The expected payoff of inspection of player 1:
 $U_p = e(rY + af - c) + (1 - e)(rY - c - q)$
- The expected payoff of non-inspecting of player 1:
 $U_{1-p} = e(rY - a) + (1 - e)rY$

Simultaneous formulas:

$$U_p = U_{1-p}, U_e = U_{1-e}$$

Thus we can get expressions containing variables p^* and e^* :

$$p^* = \frac{a - b}{a + af + q},$$

$$e^* = \frac{c + q}{a + af + q}.$$

Therefore mixed strategy of player 1 is

$$\left(\frac{a-b}{a+af+q}, \frac{af+q+b}{a+af+q} \right).$$

Mixed strategy of player 2 is

$$\left(\frac{c+q}{a+af+q}, \frac{a+af-c}{a+af+q} \right).$$

Consider the choices of player 1 and player 2 in different situations:

- If $p > p^*$, player 2 should choose not to evade tax, i.e. $e = 0$;
- If $p < p^*$, player 2 should choose to evade tax, i.e. $e = 1$;
- If $e > e^*$, player 1 should choose to inspect, i.e. $p = 1$;
- If $e < e^*$, player 1 should choose not to inspect, i.e. $p = 0$.

The figure 1 indicates the mixed-strategy Nash equilibrium: (p^*, e^*) .

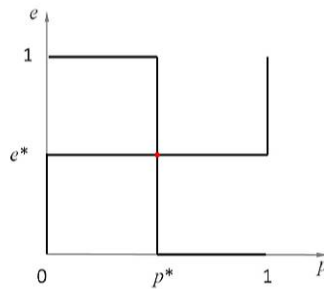


Figure 1. Mixed-strategy under the Nash equilibrium

Static game analysis conclusion:

- To confirm that the incentive factor q has influence, the derivative of e^* with respect to q : $e^{*'}(q) = \frac{a+af-c}{(a+af+q)^2}$, as $rY + af - c < rY - a$, $a + af - c < 0$, so $e^{*'}(q) < 0$, thus the model proves that theoretically people will respond to incentives, and positive incentives will reduce the probability of tax evasion, negative incentives contrary.

4. Taxation analysis in repeated game. Based on the assumptions, let k is the expected times of inspection, T is total times of game. So that the expected times of inspection are $k = p^*T$, the expected cost of inspection is $N = ck$. The factors affecting the total expected cost of inspection are the sum of the expected cost of inspection and the product, which include the expected times of inspection, rewards, probability of inspection and probability of player 2's not-evading taxes.

The total expected cost of inspection:

$$ck + qTp^*(1 - e^*) = \frac{[(1 + f)(a - b)(c + q)]}{(a + af + q)^2}.$$

At a given level, we only consider the impact of q on player 1 and player 2. In the following example, take arbitrary values for a , b , c , T and f , and these values only need to satisfy $0 \leq b < a$, $0 \leq c < a + af$ and $0 \leq c \leq b$. If $a = 50$, $b = 30$, $c = 20$, $T = 100$, and $f = 2$, Figure 2 is obtained.

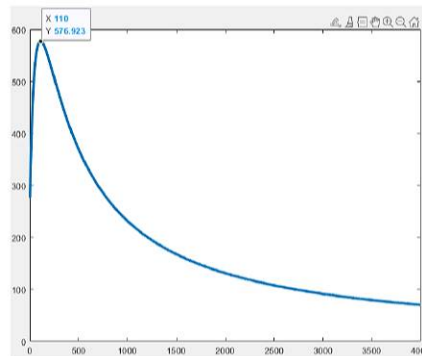


Figure 2. The abscissa represents rewards, and the ordinate represents the cost of inspection. This curve represents that the cost of inspection firstly increases and then decreases as rewards change

- If rewards $0 < q < 110$, q is positively correlated with the cost of inspection of player 1;

- If rewards $q = 110$, the cost of inspection of player 1 reaches the maximum value of 576.923;
- If rewards $q > 110$, q is negatively related to the cost of inspection player 1.

Repeated game analysis conclusion:

- The remuneration is low, and it effectively reduces tax evasion, but it increases the cost of inspection because the payoff of tax evasion of player 2 is larger than rewards while rewards q are low. According to the principle of rational people, player 2 will choose to evade tax, and the amount of tax evasion and the cost of tax evasion will increase, and the cost of inspection will also increase.
- If the incentive payoff is large, the cost of inspection will decrease as rewards increase. As player 1 gives away more rewards, the payoff of tax evasion is less than rewards. According to the principle of rational people, player 2 will choose not to evade tax. It will also reduce the cost of tax evasion and the cost of inspection.

5. Conclusion and inspiration. In static games, we consider if the rewards will incent player 2. In repeated game, being amply rewards will reduce the cost of inspection of player 1 and cost of tax evasion of player 2. The whole society will reduce deadweight losses, and social welfare will increase relatively. For player 1, it is necessary to increase incentives for player 2 who do not evade tax. For player 2, they should improve their tax compliance, and file tax return consciously, and maintain their reputation and image. This is a win-win situation for both self-interests and social interests.

References

1. Robinson J. What is perfect competition? // The Quarterly Journal of Economics. 1934. Vol. 49. No 1. P. 104–120.
2. Xuehan L. Taxation and Enterprise Game Analysis. Liaoning University. 2012.
3. Han Y., Jie C. A game analysis of taxation-analysis of citizens tax awareness // Market Modernization. 2005. Vol. 34. No 7. P. 106–107.
4. Nash J. Equilibrium points in N-person games // Proceeding of the national academy of sciences. 1950. Vol. 36. No 1. P. 48–49.

UDC 519.2

Fernández-Arias E., López-Barrientos J. D.,
Moreno-Ruíz-Esparza A.

A contingent claims approach for computing the premium to ensure an investment in highways

1. Introduction. Black and Scholes [1] developed a framework to find fair prices of financial options. Later on, Merton *et al.* [2] reformulated it and used it to find the price of a (so-called) “contingent claims”. For this case, we define a contingent claim payoff “premium” upon the value of another variable (number of cars or vehicles) is less or equal to a pre-established number at least twice a year.

In the present context, a company has offered to pay the administrator of a certain highway a premium of E if the average daily number of cars travelling on the highway touches (or exceeds) at least twice a year a barrier of B cars per month. We model the situation as a variant of the down-and-out and asset-or-nothing options, and we use the Monte Carlo simulation method (cf. [3, Chapter 15.2]) to find the fair value of the investment. We present algorithms for the European and American modalities of this instrument.

We assume that the monthly daily average volume in the highway is given by the random variable

$$S_t = S_0 \exp \left[\left(\mu - \frac{1}{2} \sigma^2 \right) t + \sigma W_t \right], \quad (1)$$

where $\mu \in \mathbb{R}$, $\sigma > 0$ and $W_t \sim \mathcal{N}(0, t)$. Note that there is no seasonal adjustment in this expression.

For the European mode, we consider the payoff function:

$$\Lambda(S_T) = \begin{cases} E & \text{if } \bar{S}_t, \bar{S}_r \geq B \text{ for } t, r \in [0; T], \\ 0 & \text{in other case,} \end{cases}$$

Fernández-Arias Ernesto – graduate student, Universidad Anáhuac México;
e-mail: ernesto.fernandezar@anahuac.mx, phone: +52(55)91980682

López-Barrientos José Daniel – full-professor, Universidad Anáhuac México;
e-mail: daniel.lopez@anahuac.mx, phone: +52(55)5627-02108506

Moreno-Ruíz-Esparza Alberto – full-professor, Universidad Anáhuac México;
e-mail: albertomorenor@anahuac.mx, phone: +52(55)5627-02108244

where \bar{S}_t and \bar{S}_r represent the monthly daily average volume for two given months $t \neq r$.

We call this option “down-and-out *with double arrival*, and asset-or-nothing” of European type. We justify this name because it is an instrument analogous to an option that makes its payment at the end of the period. Furthermore, when looking for a function of the daily force to touch or exceed a pre-specified barrier, we say that the model is congruent with the down-and-out. However, the payment offered by the instrument is of the asset-or-nothing type.

2. Methodology. We use the developments presented in [3, Chapter 19.6] to reproduce the pseudo code that calculates a 95% confidence interval of the fair price of the down-and-out call option *with double arrival*, and asset-or-nothing of European type in Algorithm 1. We write with an *emphasized* font the critical points of the algorithm.

Now we go step-by-step through the Algorithm 1. For each $i = 1, \dots, M$, line 3 gives a starting value to a counter that will indicate how many times on the horizon the average influx reaches or exceeds the B barrier (see lines 8–10). Lines 5–6 calculate a path of *geometric Brownian motion* (see [4], and [5, Example 5.1.1]). Line 7 gives us the seasonal adjusted version (see [6, p.106]) of the daily average calculated from the trajectory S_{j+1} . After having traversed the horizon, lines 12–16 show whether or not the premium E should be paid, and they give us the present value of the payment. Finally, lines 18 and 19 give us the mean and variance of our simulations. Line 20 returns the 95% confidence interval for the fair value of our portfolio.

We suggest the selection of parameters $M = 100\,000$, $T = 1$, $L = 12$, a discount rate of 7.42% (see [7]), $E = \$200\,000\,000.00$, $B = 22\,500$. To calibrate the initial value of the average daily flow, we refer to [8] and the next section.

In particular, if we take $r = 0$ and $E = 1$, then, the value a_M referred to by line 18 in Algorithm 1 is the probability that the premium E will be paid.

2.1. American exercise. For the case in which the option is exercised when the average daily number of users reaches/crosses the barrier for the second time, we propose the Algorithm 2. Note that, in this case, we assume that the payment is made *one month after* the condition is met.

If there are two arrival times $t, r \in [0; 1]$ to the set $\{\bar{S}_t : \bar{S}_t \geq B\}$, the

Algorithm 1: Computation of a 95%-confidence interval for the fair price of the down-and-out option *with double arrival*, and of the Euro-Asian-type asset-or-nothing option

Data: Number of simulations to perform M ; time horizon T ; barrier to cross/reach at least twice B ; number of evaluations of the barrier L ; monthly daily average volume at the initial time S_0 ; strike price E ; discount rate r ; drift coefficient μ and volatility σ of the growth rate of the monthly daily average volume; vector of seasonal coefficients R .

Result: 95%-confidence interval for the value of the instrument at time $t = 0$.

```

1 Compute the step size  $\Delta t \leftarrow \frac{T}{L}$ ;
2 for  $i \leftarrow 1$  to  $M$  do
3    $I \leftarrow 0$ ;
4   for  $j \leftarrow 0$  to  $L - 1$  do
5     Compute a realization of the random variable  $\xi_j \sim \mathcal{N}(0, 1)$ ;
6      $S_{j+1} \leftarrow S_j \exp \left[ \left( \mu - \frac{\sigma^2}{2} \right) \Delta t + \sigma \sqrt{\Delta t} \xi_j \right]$ ;
7      $\bar{S}_{j+1} \leftarrow L \cdot \frac{R_j}{\sum_{k=1}^L R_k} \cdot S_{j+1}$ ;
8     if  $\bar{S}_{j+1} \geq B$  then
9        $I \leftarrow I + 1$ ;
10    end
11  end
12  if  $I \geq 2$  then
13     $V_i \leftarrow e^{-rT} E$ ;
14  else
15     $V_i \leftarrow 0$ ;
16  end
17 end
18  $a_M \leftarrow \frac{1}{M} \sum_{i=1}^M V_i$ ;
19  $b_M^2 \leftarrow \frac{1}{M-1} \sum_{i=1}^M (V_i - a_M)^2$ ;
20 return  $\left[ a_M - 1.96 \frac{b_M}{\sqrt{M}}; a_M + 1.96 \frac{b_M}{\sqrt{M}} \right]$ ;

```

payoff we consider is given by $\Lambda\left(S_{t \vee r + \frac{1}{12}}\right) = E$. Otherwise, $\Lambda(S_t) = 0$ for all $t \in [0; 1]$.

The main difference between Algorithms 1 and 2 is the cycle of lines 4–11 of the Algorithm 1 is replaced by that of lines 5–13 in the Algorithm 2. Furthermore, if this last cycle is broken in the k -th period of the horizon, lines 14–18 of Algorithm 2 contemplate the realization of the payment E exactly at the beginning of the $(k + 1)$ -th period.

3. Model calibration. In this section we will find μ and σ for lines 6 and 7 of Algorithms 1 and 2, respectively. Later we will use the information available in [8] to calculate the initial value S_0 used in those same lines.

3.1. Calibration of drift and volatility. First, it is valuable to know that the expressions provided by lines 6 and 7 of the Algorithms 1 and 2 are exponential approximations of the solution of

$$dS_t = \mu S_t dt + \sigma S_t dW_t, \quad (2)$$

which is, in turn, a stochastic version of (1). We establish this last assertion in the form of a Theorem, which we take from [5, Example 5.1.1].

Theorem 1. *The expression (1) is a solution of the stochastic differential equation (2).*

Proof. Apply Itô's differential rule with $g(t, x) = \ln x$ for $x > 0$ and $t > 0$, and thus get:

$$d(\ln S_t) = \frac{1}{S_t} dS_t + \frac{1}{2} \left(-\frac{1}{S_t^2} \right) (dS_t)^2.$$

Substituting (2) in dS_t , along with Itô's isometric formula (see [5, Lemma 3.1.5 and Corollary 3.1.7]) yields:

$$d(\ln S_t) = \frac{dS_t}{S_t} - \frac{1}{2S_t^2} \sigma^2 S_t^2 dt = \frac{dS_t}{S_t} - \frac{1}{2} \sigma^2 dt. \quad (3)$$

From here, it follows that

$$\frac{dS_t}{S_t} = d(\ln S_t) + \frac{1}{2} \sigma^2 dt.$$

Algorithm 2: Computation of a 95%-confidence interval of the fair price of the down-and-out option *with double arrival*, and asset-or-nothing Asian-type with American exercise

Data: Number of simulations to perform M ; time horizon T ; barrier to cross/reach at least twice B ; number of evaluations of the barrier L ; monthly daily average volume at the initial time S_0 ; strike price E ; discount rate r ; drift coefficient μ and volatility σ of the growth rate of the monthly daily average volume; vector of seasonal coefficients R .

Result: 95%-confidence interval for the value of the investment at time $t = 0$.

```

1 Compute the step size  $\Delta t \leftarrow \frac{T}{L}$ ;
2 for  $i \leftarrow 1$  to  $M$  do
3    $I \leftarrow 0$ ;
4    $j \leftarrow 1$ ;
5   while  $I < 2 \wedge j \leq L$  do
6     Compute a realization of the random variable  $\xi_j \sim \mathcal{N}(0, 1)$ ;
7      $S_{j+1} \leftarrow S_j \exp \left[ \left( \mu - \frac{\sigma^2}{2} \right) \Delta t + \sigma \sqrt{\Delta t} \xi_j \right]$ ;
8      $\bar{S}_{j+1} = L \cdot \frac{R_j}{\sum_{k=1}^L R_k} \cdot S_{j+1}$ ;
9     if  $\bar{S}_{j+1} \geq B$  then
10       $I \leftarrow I + 1$ ;
11    end
12     $j \leftarrow j + 1$ ;
13  end
14  if  $I \geq 2$  then
15     $V_i \leftarrow e^{-r \frac{k}{L}} E$ ;
16  else
17     $V_i \leftarrow 0$ ;
18  end
19 end
20  $a_M \leftarrow \frac{1}{M} \sum_{i=1}^M V_i$ ;
21  $b_M^2 \leftarrow \frac{1}{M-1} \sum_{i=1}^M (V_i - a_M)^2$ ;
22 return  $\left[ a_M - 1.96 \frac{b_M}{\sqrt{M}}; a_M + 1.96 \frac{b_M}{\sqrt{M}} \right]$ ;

```

Integration of both sides gives:

$$\int_0^t \frac{dS_r}{S_r} = \int_0^t d(\ln S_r) + \frac{1}{2}\sigma^2 \int_0^t dr. \quad (4)$$

Note that, by (2), we have that

$$\int_0^t \frac{dS_r}{S_r} = \mu t + \sigma W_t,$$

while elementary calculus yields

$$\int_0^t d(\ln S_r) + \frac{1}{2}\sigma^2 \int_0^t dr = \ln \frac{S_t}{S_0} + \frac{1}{2}\sigma^2 t.$$

Using (4) and reducing terms yields the result. \square

Now we will show that the approximations of lines 6 and 7 of Algorithms 1 and 2, respectively, are equivalent to the expression (1). The case of the geometric Brownian motion is well known, and it is a particular case of the so-called Euler–Maruyama method (see [4]).

Consequence 1. *The approximations of lines 6 and 7 of Algorithms 1 and 2, respectively, are equivalent to the expression (1).*

Proof. Consider (3), and note that, by (2), we have that

$$d(\ln S_t) = \left(\mu - \frac{1}{2}\sigma^2 \right) dt + \sigma dW_t.$$

Letting $\nu := \mu - \frac{1}{2}\sigma^2$, we see that

$$d(\ln S_t) = \nu dt + \sigma dW_t.$$

Taking *sufficiently small* increments makes this expression equivalent to

$$\ln S_{t_{j+1}} - \ln S_{t_j} = \nu \Delta t + \sigma \sqrt{\Delta t} \xi_j,$$

where $\xi_j \sim \mathcal{N}(0, 1)$. Taking exponentials in both sides yields the result. \square

Next we describe how to obtain the parameters $\nu = \mu - \frac{1}{2}\sigma^2$ and σ to use them in the Algorithms 1 and 2. We use (1) to define the daily influx of highway users in the years 2023 and 2028:

$$S_{2023} = S_{2018} \exp \left[5\nu + \sqrt{5} \cdot \sigma Z \right], \quad (5)$$

$$S_{2028} = S_{2018} \exp \left[10\nu + \sqrt{10} \cdot \sigma Z \right], \quad (6)$$

where $Z \sim \mathcal{N}(0, 1)$. Using the tables in section 7.1.3 in [8] yields

$$\mathbb{P}(S_{2023} > 19\,460) = 0.8, \quad (7)$$

$$\mathbb{P}(S_{2028} > 21\,370) = 0.9. \quad (8)$$

Using (5)–(6) we rewrite this information and thus see that:

$$\begin{aligned} \mathbb{P} \left(16\,816 e^{5\nu + \sqrt{5} \cdot \sigma Z} > 19\,460 \right) &= 0.8, \\ \mathbb{P} \left(16\,816 e^{10\nu + \sqrt{10} \cdot \sigma Z} > 21\,370 \right) &= 0.9. \end{aligned}$$

Equivalently:

$$\begin{aligned} \mathbb{P} \left(5\nu + \sqrt{5} \cdot \sigma Z > \ln \frac{19\,460}{16\,816} \right) &= 0.8, \\ \mathbb{P} \left(10\nu + \sqrt{10} \cdot \sigma Z > \ln \frac{21\,370}{16\,816} \right) &= 0.9. \end{aligned}$$

Considering that $\Phi(z) := \mathbb{P}(Z \leq z)$ is the distribution function of a Normal standard random variable, we see that:

$$\begin{aligned} \Phi \left(\frac{\ln \frac{19\,460}{16\,816} - 5\nu}{\sqrt{5} \cdot \sigma} \right) &= 0.2, \\ \Phi \left(\frac{\ln \frac{21\,370}{16\,816} - 10\nu}{\sqrt{10} \cdot \sigma} \right) &= 0.1. \end{aligned}$$

The following linear system in the unknowns ν and σ stems from here:

$$\begin{aligned} \frac{\ln \frac{19\,460}{16\,816} - 5\nu}{\sqrt{5} \cdot \sigma} &= -0.841621233572915, \\ \frac{\ln \frac{21\,370}{16\,816} - 10\nu}{\sqrt{10} \cdot \sigma} &= -1.2815515655446. \end{aligned}$$

The solution of the system is

$$\nu = 0.097507044, \quad \sigma = 0.181466025.$$

The decision to take the twentieth and tenth percentiles at (7) and (8), respectively, is contingent on our desire to reflect the effect of the great variability of the data on the inflow to cause of the uncertainty caused by the phenomenon of the 2020 pandemic, but it is deliberate. If we take, for example, the percentiles in the inverted sense, (that is $\mathbb{P}(S_{2023} > 19\,076) = 0.9$ and $\mathbb{P}(S_{2028} > 21\,891) = 0.8$) and apply an analogous procedure to the one just described, the results are

$$\nu = 0.027375338, \quad \sigma = 0.003760598.$$

A robust criterion is necessary in this regard. In fact, an expert criterion makes us assume that the value of ν will be 33% higher than that reported, due to our estimate of the recovery of the Mexican economy during the coming year. Thus, the values we will use are

$$\nu^* = 0.130009067, \quad \sigma = 0.003760598.$$

On the other hand, *we do not* recommend trying to solve the resulting system for any of the other random variables available in section 7 of [8] (*i.e.* S_{2033} , S_{2038} and S_{2043}), since our hypothesis about the distribution referred to in (1) may not apply in some scenario so far on the horizon. Regarding the choice of the median to do the calculations, note that the linear system that we would obtain would be trivial, since one of the equations would have only one variable: σ . For this reason we did not choose this percentile to carry out our calculations.

3.2. Initial flow calibration. We have resolved to calibrate the starting point of our exercise with a simple linear interpolation between the data provided in the tables in section 7 of [8]. We reproduce an extract from the relevant table below.

Year	Flux
2018	16 816
2023	19 076

According to the study [8], the probability that the monthly average of the daily influx is greater than the values that we see in this table, is

greater than 90%. To estimate the value of S_{2020} , we use the traditional linear interpolation procedure.

$$\begin{aligned} S_{2020} &= \frac{S_{2023} - S_{2018}}{2023 - 2018} (2020 - 2018) + S_{2018} \\ &= 17\,720. \end{aligned}$$

It is also possible to use the table that estimates the monthly averages of the daily influx with a reliability of 80%.

Year	Flux
2018	16 816
2023	19 460

Using the linear interpolation formula just cited with this table, we see that it is also possible to use $S_{2020} = 17\,873.6$. However, we believe that our first approximation is more conservative and robust.

As a third scenario, we propose the use of the average capacity table for a scenario with a rate increase due to inflation, with a correction factor due to the effect of the COVID-19 pandemic.

Year	Flux
2018	16 816
2023	19 847

To estimate the value of S_{2020} , we use the traditional linear interpolation procedure.

$$\begin{aligned} S_{2020} &= \frac{S_{2023} - S_{2018}}{2023 - 2018} (2020 - 2018) + S_{2018} \\ &= 18\,028.40. \end{aligned}$$

We finalize the calibration of the average daily source for our model assuming a 90% correction for the effect of the pandemic that plagued the world in 2020. Thus, $S_{2020} = 16\,225.56$.

4. Execution of Algorithm 2. In this section we implement the Algorithm 2, incorporating the calibrations of the last section. In summary, we consider the following parameters.

$M = 100\,000$	$\nu^* = 0.130009067$
$S_{2020} = 16\,225.56$	$\sigma = 0.181466025$
$E = \$200\,000\,000\,000.00$	$B = 22\,500$
$r = 7.42\%$	$T = 1, L = 12$

We also take into account this vector of seasonality factors, which was obtained from the study [9], which collects information from Federal Roads and Bridges:

October	0.89415	November	0.96435
December	1.09077	January	0.94472
February	0.86806	March	0.95401
April	1.14642	May	1.00693
Jun	0.97059	July	1.16154
August	1.05416	September	0.94429

These data give us a fair price for the instrument of \$46 166 959 189.0708, together with the 95% confidence interval given by

$$[\$46\ 164\ 224\ 383.76; \$46\ 169\ 693\ 995.38],$$

a probability of exercise 24.34%, and the 95% confidence interval for this probability of

$$[24.3395229\%; 24.3404771\%].$$

If instead of the above, we use $S_0 = 17\ 720$, as suggested by the first variant in section 3, we obtain a fair price of \$87 175 828 738.84, and a probability of exercise of 46.2860%. The 95% confidence intervals are

$$[\$87\ 174\ 120\ 084.06; \$87\ 177\ 537\ 394.63], \text{ and } [46.28509\%; 46.28691\%],$$

respectively.

Using $S_0 = 17\ 873.60$, as the second variant in section 3.2 suggests, gives us a 95% confidence interval for the fair price of

$$[\$91\ 238\ 207\ 088.94; \$91\ 241\ 954\ 976.54],$$

and a fair price of \$91 240 081 031.74; and a probability of exercise of 48.321%, with a 95% confidence interval of

$$[48.3200529\%; 48.3219471\%].$$

We also note that the use of the variants cited in section 3.1 is trivial. In effect, these variants result in a null price and probability of collection for the three initial levels of S_0 proposed in section 3.2.

References

1. Black F., Scholes M. The pricing of options and corporate liabilities // The Journal of Political Economy. 1973. Vol. 81. No 3. P. 637–654.
2. Merton R. C., Brennan M. J., Schwartz E. S. The valuation of American put options // The Journal of Finance. 1977. Vol. 32. No 2. P. 449–462.
3. Higham D. An introduction to financial option valuation. New York: Cambridge University Press, 2004. 296 p.
4. Higham D. An algorithmic introduction to numerical simulation of stochastic differential equations // SIAM Review. 2001. Vol. 43. No 3. P. 525–546.
5. Øksendal B. Stochastic Differential Equations: An Introduction with applications. Springer Science & Business Media, 2003. 406 p.
6. Wright P. H., Paquette R. J. Highway Engineering. New York: John Wiley and Sons Ltd, 1987. 727 p.
7. Herrasti A. Spreadsmat_v01. Publisher: C4P1T4L, 2020.
8. Paucar D., Mercado J., López R. et al. Actualización de la Demanda para la Autopista Naucalpan-Toluca. Publisher: Black Rock, 2017.
9. Herrasti A. Factor de Estacionalidad. Publisher: C4P1T4L, 2020.

UDC 519.83

Gao J., Parilina E.

Average-oriented opinion dynamics with the last moment of observation

1. Introduction. Interpersonal communication changes opinions of the society members. A model of opinion dynamics with two principals or two centers of influence, where the influence matrix is given, is introduced in [1], where existence conditions of the limit opinion are obtained. The competitive models of opinion dynamics are considered in [2], where the dynamic game of opinion formation is linear-quadratic. The authors found equilibria that represent the degree of influence of different information structures. A model of opinion dynamics with two agents and one or two centers of influence is proposed in [3] and the optimal strength of influence is found. The Nash equilibrium found in [3] helps to reduce the cost of having social network members close to the target opinion. In the paper, the player aims in finding minimal total costs and making the society opinion close to the target one when observation is possible only at the end of the time horizon. We characterize the optimal strategy of the player using the Euler equation approach [4] and make numerical simulations to demonstrate theoretical results.

2. The model of opinion dynamics with the terminal observation. There are two agents with corresponding opinions $x_1(t), x_2(t) \in \mathbb{R}^1$. The agent 1 is influenced by the player's control $u(t) \in U \subset \mathbb{R}^1$. The dynamics of the agents' opinion is given by the following equations:

$$x_1(t+1) = x_1(t) + a_1 \left(\frac{x_1(t) + x_2(t)}{2} - x_1(t) \right) + u(t),$$

$$x_2(t+1) = x_2(t) + a_2 \left(\frac{x_1(t) + x_2(t)}{2} - x_2(t) \right),$$

Gao Jingjing – student, Saint Petersburg State University; e-mail: GJJ127@outlook.com, phone: +7(812)428-42-86

Parilina Elena – professor, Saint Petersburg State University; e-mail: e.parilina@spbu.ru, phone: +7(812)428-42-86

The work of the second author was supported by the Russian Science Foundation under grant, no. 17-11-01079

with initial condition

$$x_1(0) = x_1^0, \quad x_2(0) = x_2^0, \quad (1)$$

where $a_1, a_2 \in \mathbb{R}^+$ represent the strength of influence of the average opinion on the agent's opinion.

We can rewrite equations of dynamics in the following form:

$$x_1(t+1) = x_1(t) + \frac{a_1}{2}(x_2(t) - x_1(t)) + u(t), \quad (2)$$

$$x_2(t+1) = x_2(t) + \frac{a_2}{2}(x_1(t) - x_2(t)). \quad (3)$$

The player minimizes a functional that is

$$J(u) = \sum_{t=0}^{T-1} \delta^t (cu^2(t)) + \delta^T \left((x_1(T) - s)^2 + (x_2(T) - s)^2 \right), \quad (4)$$

where $c > 0$ is the unit control costs, $s > 0$ is a target player's opinion. Having the function in the form, we assume that the player does not take into account the difference between $x_1(t), x_2(t)$ and the target opinions all the time periods, but he is interested in a final result.

3. Main result. In this section we characterize the optimal control of the player using Euler-equation approach.

Theorem 1. *Let $\{u^*(t) : t = 0, \dots, T-1\}$ be the optimal strategy minimizing Functional (4) subject to initial conditions (1) and state dynamics equations (2) and (3) and $\{(x_1^*(t), x_2^*(t)) : t = 0, \dots, T\}$ be the corresponding state trajectory, then the optimal strategy $u^*(t), t = 0, \dots, T-1$ is*

$$u^*(t) = z^*(t+1) - Az^*(t)$$

and corresponding optimal state trajectory $(x_1^*(t), x_2^*(t)), t = 1, \dots, T$ satisfies the equations:

$$\begin{cases} cA\delta^2 z(t+1) - Bz(t) + Cz(t-1) - Acz(t-2) = 0, \\ \quad t = \overline{2, T-1}, \\ - \left(cA - \frac{a_2\delta}{2} \right) z(T) + \left(A^2c + \frac{c}{\delta} \right) z(T-1) - A\frac{c}{\delta} z(T-2) \\ \quad = -\delta a_2 (x_2(T) - s), \\ \frac{c}{\delta} (z(T) - Az(T-1)) + z(T) + x_2(T) - s = 0, \\ x_2(t+1) = x_2(t) + \frac{a_2}{2} z(t), \quad t = \overline{1, T-1}, \end{cases} \quad (5)$$

where $z^*(t) = x_1^*(t) - x_2^*(t)$, $A = 1 - \frac{a_1+a_2}{2}$, $B = cA\delta + c\delta + cA^2\delta$, and $C = c + cA\delta + cA^2\delta$.

Proof. We consider a new variable $z(t) = x_1(t) - x_2(t)$, $t = \overline{0, T}$. From state equations (2) and (3) substituting $z(t)$, we obtain the new state equations:

$$z(t+1) = Az(t) + u(t), \quad (6)$$

$$x_2(t+1) = x_2(t) + \frac{a_2}{2}z(t), \quad (7)$$

where $A = 1 - \frac{a_1+a_2}{2}$, with initial condition

$$z(0) = x_1^0 - x_2^0, \quad x_2(0) = x_2^0.$$

We find expression of $u(t)$ from (6):

$$u(0) = z(1) - Az(0),$$

$$u(t) = z(t+1) - Az(t) = q(z(t), z(t+1)), \quad t = \overline{0, T-1}.$$

Then we substitute expression to $\sum_{t=0}^T \delta^t g_t(x(t), x(t+1))$, we can get

$$\begin{aligned} J(z, x_2) &= c(z(1) - Az(0))^2 + \sum_{t=1}^{T-1} \delta^t [c(z(t+1) - Az(t))^2] \\ &\quad + \delta^T [(z(T) + x_2(T) - s)^2 + (x_2(T) - s)^2]. \end{aligned}$$

Minimizing $J(z, x_2)$ under equations (6) and (7), we form the Lagrange function:

$$L(z, x_2, k) = J(z, x_2) + \sum_{t=1}^{T-1} k_t \left(x_2(t+1) - x_2(t) - \frac{a_2}{2}z(t) \right).$$

The first order conditions should be $\frac{\partial L(z, x_2)}{\partial z(t)} = 0$, $t = \overline{1, T}$ and $\frac{\partial L(z, x_2)}{\partial x_2(t)} = 0$, $t = \overline{1, T}$.

Using Euler equation approach, we get the system

$$\begin{cases} -\frac{cA}{\delta}z(t-1) + z(t) \left(\frac{c}{\delta} + cA^2 \right) - cAz(t+1) \\ \quad = \frac{a_2}{4}k_t\delta^{-t}, \quad t = \overline{1, T-1}, \\ \frac{c}{\delta}(z(T) - Az(T-1)) + z(T) + x_2(T) - s = 0, \quad t = T, \\ k_{t-1} - k_t = 0, \quad t = \overline{2, T-1}, \\ \delta^t [2z(t) + 4(x_2(t) - s)] + k_{t-1} = 0, \quad t = T, \end{cases} \quad (8)$$

with initial condition $z(0) = x_1^0 - x_2^0, x_2(0) = x_2^0$.

Excluding k_t from system (8), we can get

$$\begin{cases} cA\delta^2 z(t+1) - Bz(t) + Cz(t-1) - Acz(t-2) = 0, \\ t = \overline{2, T-1}, \\ -\left(cA - \frac{a_2\delta}{2}\right) z(T) + \left(A^2c + \frac{c}{\delta}\right) z(T-1) - A\frac{c}{\delta}z(T-2) \\ = -\delta a_2(x_2(T) - s), \\ \frac{c}{\delta}(z(T) - Az(T-1)) + z(T) + x_2(T) - s = 0, \end{cases} \quad (9)$$

where $B = cA\delta + c\delta + cA^2\delta$, and $C = c + cA\delta + cA^2\delta$. The theorem is proved.

4. Numerical simulation. To demonstrate theoretical results we consider the following example. Let $a_1 = 0.2, a_2 = 0.6, \delta = 1, c = 2$ and initial opinions be $x_1(0) = 0.7, x_2(0) = 0.2$. For time horizon $T = 10$ and target opinion $s = 0.4$, the optimal state and optimal control trajectories are presented in Table 1. The optimal value for functional (4) is 0.068.

t	0	1	2	3	4	5
$x_1(t)$	0.7000	0.4714	0.4159	0.4024	0.3991	0.3984
$x_2(t)$	0.2000	0.3500	0.3864	0.3953	0.3974	0.3979
$z(t)$	0.5000	0.1214	0.0295	0.0071	0.0017	0.0005
$u(t)$	-0.1786	-0.0433	-0.0106	-0.0026	-0.0005	-0.0004
t	6	7	8	9	10	
$x_1(t)$	0.3980	0.3984	0.3976	0.3984	0.3989	
$x_2(t)$	0.3981	0.3980	0.3982	0.3980	0.3981	
$z(t)$	-0.0001	0.0004	-0.0006	0.0004	0.0008	
$u(t)$	0.00046	0.00084	0.00076	0.00056		

Table 1. Optimal state trajectory and control

We introduce optimal state and strategy trajectories in figure 1 and figure 2, where the target opinion $s = 0.4$.

From figure 1, it can be seen that the opinions of both agents reach the target opinion under the influence of the player very fast. In figure 2, the player's influence on the agents tends to be stable.

5. Conclusion. A model of average-oriented opinion dynamics with one center of influence and two agents is proposed. A player aims in making the opinion of the society closer to the target one at the terminal time period minimizing his costs on influence. We find the player's optimal influence power when there is a unique center of the social network. The Euler equation approach is used to find the optimal strategy. The numerical simulations verify the theoretical results.

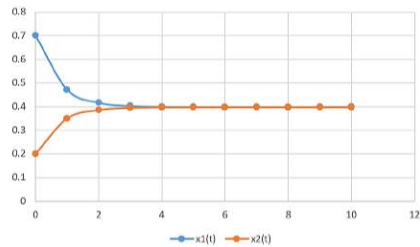


Figure 1. Optimal state trajectories (blue $x_1(t)$, red $x_2(t)$).

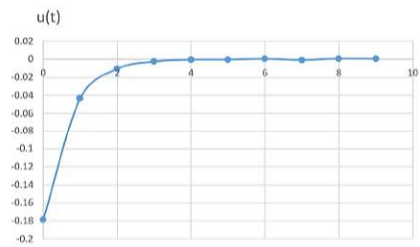


Figure 2. Optimal strategy trajectory $u(t)$.

References

1. Bure V. M., Parilina E. M., Sedakov A. A. Consensus in a social network with two principals // Automation and Remote Control. 2017. Vol. 78. No. 8. P. 1489–1499.
2. Sedakov A. A., Zhen M. Opinion dynamics game in a social network with two influence nodes // Vestnik of Saint Petersburg University. Applied Mathematics. Informatics. Control processes. 2019. Vol. 15. No. 1. P. 118–125.
3. Mazalov V., Parilina E. M. The Euler-equation approach in average-oriented opinion dynamics // Mathematics. 2020. Vol. 8. No. 3. Art. No 355.
4. Gonzalez-Sanchez D., Hernandez-Lerma O. Discrete-time stochastic control and dynamic potential games: the Euler-equation approach. Series: SpringerBriefs in Mathematics. Springer, 2013. 83 p.

UDC 311.312

He Y.

Analysis of the most important factors of China's air quality index

Recommended for publication by professor Bure V. M.

1. Abstract. There are six types of linear regression analysis that available in statistics which are simple linear regression, multiple linear regression, logistic regression, ordinal regression, multinomial regression, and discriminant analysis [1]. In our research, we use multiple linear regression analysis of the factor to predict the China air quality index in different years. The main objective of this paper is to find the important factors of the China air quality index. To ensure that the final set of explanatory variables is optimal. We also use stepwise regression to get the final model [2]. Data that has been used in this research were taken from the National Bureau of Statistics [3]. Then the data will be analyzed by using RStudio and Excel [4].

2. China air quality index. The China air quality index is used by China government agencies to communicate to the public how polluted the air currently is or how polluted it is forecast to become in China.

2.1. Air quality index computation.

$$AQI = \max \{IAQI_1, \dots, IAQI_n\},$$

where AQI is air quality index, $IAQI_n$ is the individual air quality index. N is from 1 to 6, represent the different pollutant projects. Pollutant item 1 is sulfur dioxide, pollutant item 2 is nitrogen dioxide, pollutant item 3 is particulate matter 10 micrometers, pollutant item 4 is carbon monoxide, pollutant item 5 is ozone, pollutant item 6 is particulate matter 2.5 micrometers.

2.2. Individual air quality index computation.

$$IAQI_n = \frac{IAQI_{Hi} - IAQI_{Lo}}{BP_{Hi} - BP_{Lo}}(C_n - BP_{Lo}) + IAQI_{Lo},$$

He Yang – student, Saint Petersburg State University; e-mail: hy1186867324@outlook.com, phone: +7(965)794-39-49

where $IAQI_n$ is the sub air quality index of pollutant project n , n represents pollutant project; C_n is the mass concentration value of the pollutant item n ; BP_{H_i} is the high value of the pollutant concentration limit close to C_n in the pollutant item concentration limit table corresponding to the air quality sub-index; BP_{L_o} is the low value of the pollutant concentration limit close to C_n in the pollutant item concentration limit table corresponding to the air quality sub-index; $IAQI_{H_i}$ is the air quality sub-index corresponding to BP_{H_i} in the pollutant item concentration limit table corresponding to the air quality sub-index. $IAQI_{L_o}$ is the air quality sub-index corresponding to BP_{L_o} in the pollutant item concentration limit table corresponding to the air quality sub-index. Except for $IAQI_n$ and C_n other items were from technical regulation on ambient air quality index [5].

3. Logarithm China air quality index model. We try to establish the following econometric model:

$$y_{i,t} = f(x_{1,i,t}, \dots, x_{10,i,t}), \quad i = 1, \dots, 31, \quad t = 2013, \dots, 2019,$$

where i – different provincial capital cities in China; t – different years; $\hat{y}_{i,t}$ is Logarithm of air quality index in different regions at different years; $x_{1,i,t}$ is annual average concentration of SO_2 in different regions at different years ($\mu g/m^3$); $x_{2,i,t}$ is annual average concentration of NO_2 in different regions at different years ($\mu g/m^3$); $x_{3,i,t}$ is annual average concentration of PM_{10} in different regions at different years ($\mu g/m^3$); $x_{4,i,t}$ – 95th percentile daily average concentration of CO in different regions at different years ($\mu g/m^3$); $x_{5,i,t}$ – 90th percentile daily maximum 8 hours average concentration of O_3 in different regions at different years ($\mu g/m^3$); $x_{6,i,t}$ is annual average concentration of $PM_{2.5}$ in different regions at different years ($\mu g/m^3$); $x_{7,i,t}$ is annual average temperature in different regions at different years ($^{\circ}C$); $x_{8,i,t}$ is annual average humidity in different regions at different years (%); $x_{9,i,t}$ is annual average precipitation in different regions at different years (mm); $x_{10,i,t}$ is annual average number of sunshine in different regions at different years (hour).

The first step is to build the logarithm China air quality index model by checking assumptions of data. There are three assumptions that should be checked which is normality, linearity, and heteroscedasticity.

From the first assumption, all of the variables in our research must be a normal distribution. The normal distribution can be seen by histogram graph, plot Q-Q, kurtosis, and skewness. If the distribution of data is

not normal we need transformation. In our research, all data are normal distribution. Then, multiple linear regression should have a linear relationship between the response variable and controlled variables.

The third assumption in multiple linear regression is any data should free from heteroscedasticity. Heteroscedasticity will happen whatever there is an interruption in the model that is not fulfilled. If the important variable in the model is missing, hence heteroscedasticity will happen.

Table 1. The dynamics of select final model by backward elimination stepwise

Variable	R	R^2	Adj R^2	F-value	P(F)
1,2,3,4,5,6,7,8,9,10	0.997	0.995	0.993	487.35	$1.28 \cdot 10^{-21}$
1,2,3,4,5,6,7,9,10	0.997	0.995	0.994	569.28	$5.20 \cdot 10^{-23}$
2,3,4,5,6,7,9,10	0.997	0.995	0.994	667.07	$2.01 \cdot 10^{-24}$
2,3,5,6,7,9,10	0.997	0.995	0.994	792.40	$7.21 \cdot 10^{-26}$
2,3,5,6,7,9	0.997	0.995	0.994	960.40	$2.34 \cdot 10^{-27}$
2,3,5,6,7	0.997	0.995	0.994	1192.90	$6.89 \cdot 10^{-29}$
2,5,6,7	0.997	0.995	0.995	1508.87	$2.30 \cdot 10^{-30}$
2,5,6	0.997	0.995	0.995	1897.39	$9.07 \cdot 10^{-32}$

3.1. Applied logarithm China air quality index model in 2019 years. As we see in Table 1, the first row is the initial model, the value of R^2 shows that there are 99.5 percent changes in the response variable (LAQI) occurs because of changes in the combination of ten controlled variables. The value of adjusted R^2 equal to 0.993 which means the regression equation determines that 99.5 percent of the variance of the dependent variable is within the observed value. Multiple correlation coefficient R equal to 0.997 is close 1, which means dependent variable y and controlled variables (x_1, \dots, x_{10}) have highly related.

From the Fisher test, corresponding to P (F-test) is $1.28 \cdot 10^{-21}$ less than 0.05, we define the coefficient level as 0.05, so we conclude the model is significant.

We also combine F-test and T-test to get the final model, from the table in each step we delete an independent variable. As we see, in Table 1 we delete a big number, corresponding to delete a not significant variable, to get maximal F-test value. This is a dynamic process. The stop condition is all variables are significant. In each step, we also need to check the residual is the normal distribution and the autocorrelation coefficient is zero by hypothesis test. We use Kolmogorov–Smirnov, Shapiro–Wilk, Anderson-darling test to test residual normality. We also use the Darbin–Watson to test the residual autocorrelation coefficient is zero or not. Compared with the traditional multiple linear regression model, we don't test multicollinearity, because in our research

we just consider the relationship between independent variables and the dependent variable.

In 2019, we get the final model are as follows:

$$y = 3.18 + 1.47 \cdot 10^{-3}x_2 + 8.85 \cdot 10^{-3}x_5 - 1.46 \cdot 10^{-3}x_6.$$

3.2. Applied logarithm China air quality index model in 2013–2018 years. We use the same method to get the final model in the 2013–2018 years. The results are as follows:

In 2013 (P(F)= $5.10 \cdot 10^{-15}$)

$$y = 4.48 - 0.15x_1 + 0.11x_6 - 9.30 \cdot 10^{-3}x_7 - 9.70 \cdot 10^{-3}x_8 + 1.80 \cdot 10^{-4}x_9.$$

In 2014 (P(F)= $1.47 \cdot 10^{-12}$)

$$y = 4.17 - 8.35 \cdot 10^{-3}x_2 + 8.56 \cdot 10^{-3}x_6 - 0.11x_8 + 2.34 \cdot 10^{-4}x_9.$$

In 2015 (P(F)= $2.03 \cdot 10^{-14}$)

$$y = 3.58 - 1.90 \cdot 10^{-3}x_1 + 4.99 \cdot 10^{-3}x_5 + 6.99 \cdot 10^{-3}x_6 - 8.15 \cdot 10^{-3}x_7.$$

In 2016 (P(F)= $1.01 \cdot 10^{-15}$)

$$y = 3.46 - 1.75 \cdot 10^{-3}x_1 + 6.23 \cdot 10^{-3}x_5 + 5.62 \cdot 10^{-3}x_6 - 0.14x_7 + 7.47 \cdot 10^{-5}x_9.$$

In 2017 (P(F)= $6.77 \cdot 10^{-20}$)

$$y = 3.31 + 7.41 \cdot 10^{-3}x_5 + 2.25 \cdot 10^{-3}x_6.$$

In 2018 (P(F)= $2.77 \cdot 10^{-31}$)

$$y = 3.18 + 2.12 \cdot 10^{-3}x_2 + 8.67 \cdot 10^{-3}x_5 - 1.39 \cdot 10^{-3}x_6.$$

4. Result and discussion. We take our research results in a Table 1. As we see, the different columns represent different controlled variables. To be more specific, we set each column in different colors. The far left is the year corresponding to the final model, the far right is the number of control variables in the final model, and the middle is the control variable in the final model. The bottom number represents the influence factor of the control variable.

Therefore, from Figure 1 we could easily find that the controlled variable x_6 (ozone) is one of the most important factors in China's air

quality index. Followed by x_5 ($PM_{2.5}$). I represent the different capital cities of China in different years.

5. Conclusion. In our research, we use the multiple linear regression analysis of ten controlled variables which one or more controlled variables are the most important factors, including six pollutant items and four meteorological factors to predict China's air quality index. From the result above, we conclude that the $PM_{2.5}$ and ozone are the most important factors of China's air quality index.

	x11	x21	x31	x41	x51	x61	x71	x81	x91	x101	
2013											5
2014											4
2015											4
2016											5
2017											2
2018											3
2019											3
	3	3	0	0	5	7	3	2	3	0	

Figure 1. Regression result

References

1. Harrell F. E., Jr. Regression Modeling Strategies. Springer series in statistics. New York: Springer-Verlag, 2001. 596 p.
2. Bure V. M., Parilina E. M. Probability Theory and Mathematical Statistics. SPb.: Lan, 2013. 416 p.
3. National Bureau of Statistics [Internet resource]: URL:<http://www.stats.gov.cn/tjsj/ndsj> (date: 15.12.20).
4. Bure V. M., Parilina E. M., Sedakov A. A. Methods of Applied Statistics in R and Excel. SPb.: Lan, 2016. 152 p.
5. Ambient Air Quality Index (AQI) technical regulations. Ministry of Environmental Protection of the People's Republic of China. [Internet resource]: URL:<http://kjs.mee.gov.cn/hjbhbz/bzwb/jcffbz> (date: 22.02.2021).

UDC 519.233.5

Qi D.

Research on factors affecting differences in investment attractiveness based on HLM model

Recommended for publication by professor Bure V. M.

1. Abstract. This article analyzes the influencing factors and formation mechanism of investment attractiveness differences from the national and provincial levels, which is of great significance to promoting economic development and improving people's living standards. Based on the Hierarchical Linear Model's principles and methods and based on the survey data of mainland China's provinces and provincial administrative regions, empirical analysis is carried out on the influencing factors and formation mechanisms of investment attractiveness differences. We discovered that the investment gravitational difference has a complicated mechanism of action, and it is jointly solved from multiple levels and perspectives such as total import and export trade, taxation, and per capita income.

2. The model.

2.1. Overview of Hierarchical linear Model. The design of the sections using commands Hierarchical Linear Model (HLM) was proposed by Bryk and Raudenbush in 1992. It is based on the ordinary linear regression analysis model and can process individual and group-level data simultaneously. When processing the seal data, first establish a regression equation with the first level (individual level) explanatory variables, then use the intercept and slope in the equation as dependent variables, and then use the second level (group level) explanatory variables as independent variables, perform a second regression. The basic form is

$$L_1 : Y_{ij} = \beta_{0j} + \beta_{1j}X_m + r_{ij}, \quad (1)$$

$$L_2 : \beta_{0j} = \gamma_{00} + \gamma_{01}W_n + \mu_{0j}, \quad (2)$$

$$\beta_{1j} = \gamma_{10} + \gamma_{11}W_n + \mu_{1j}. \quad (3)$$

Qi Dongfang – graduate student, Saint Petersburg State University; e-mail: st073409@student.spbu.ru, phone: +7(981)810-21-63

In formulas (1) to (3): L_1 and L_2 represent two levels respectively; Y_{ij} is the dependent variable, i represents each unit of the first level, j represents each unit of the second level; X_m, W_n are the first and second levels, respectively level explanatory variables, m and n ask the number of independent variables at the first and second levels respectively; $r_{ij}, \mu_{0j}, \mu_{1j}$ are random effects, indicating the unexplained part of the independent variable; intercept β_{0j} is the average value of Y_{ij} , that is, the group average—the result of the centering processing of the number or the total average; β_{1j} represents the regression slope.

In this article, the maximum likelihood estimation method is used to estimate the regression equations' parameters at each level through the HLM software.

2.2. Model construction. Taking “investment attractiveness” as the explained variable, to eliminate the variance of the data and smooth the data, it is necessary to perform logarithmic processing on the income data and construct a multi-level linear model shown in table 1. To use a multi-level linear model to study the income differences of rural households, we first need to test the model's usability.

1. As described in Model 1, this model is ANOVA model and does not include any independent variables. It only has random effects at the individual level. It is to determine how much of the overall variation of the investment attractiveness logarithm $\ln y$ is caused by the group level to test the multi-level model's feasibility for research.

ANOVA model:

$$L_1 : \ln Y_{ij} = \beta_{0j} + r_{ij}, \quad L_2 : \beta_{0j} = \gamma_{00} + \mu_{0j}, \quad (4)$$

where $\ln Y_{ij}$ is the logarithm of investment attractiveness; β_{0j} is the average value of the j -th two-story unit $\ln Y_{ij}$; r_{ij} is the variation of the j -th two-story unit $\ln Y_{ij}$; γ_{00} is the total average of all two-level units $\ln Y_{ij}$; μ_{0j} is the second layer Random component of the equation.

2. Model 2 is a random coefficient regression model, which only contains individual-level variables, but allows the intercept and regression coefficients of the individual-level equations to vary randomly between each group level to measure the individual level's effect on investment attractiveness.

Random coefficient regression model:

$$\begin{aligned}
L_1 : \ln Y_{ij} &= \beta_{0j} + \beta_{nj}X_n + r_{ij}, \\
L_2 : \beta_{0j} &= \gamma_{00} + \mu_{0j}, \\
\beta_{nj} &= \gamma_{n0} + \mu_{nj},
\end{aligned} \tag{5}$$

where X_n is the n -th independent variable of the first level; β_{nj} is the regression slope of the n -th independent variable of the first level; r_{nj} is the average value of β_{nj} ; μ_{nj} is the random component of β_{nj} .

3. Model 3 adds group-level variables to the intercept term based on controlling individual-level variables, measures the group's direct effect on individuals, and allows individual-level variables to vary randomly between groups. We regard it as a particular type of complete model form:

$$\begin{aligned}
L_1 : \ln Y_{ij} &= \beta_{0j} + \beta_{nj}X_n + r_{ij}, \\
L_2 : \beta_{0j} &= \gamma_{00} + \gamma_{0m}W_m + \mu_{0j}, \\
\beta_{nj} &= \gamma_{n0} + \mu_{nj},
\end{aligned} \tag{6}$$

where W_m is the m independent variables of the second level, and r_{0m} is the regression slope of the m -th independent variable of the second level.

4. Model 4 is a complete model, through the result of the interaction of the combination of individuals to explore the difference in investment attractiveness process, that is, individuals to the investment attractiveness of indirect adjustment, the second level of variable coefficient symbol and the first level of corresponding variable coefficient symbol the same play a strengthening role, on the contrary, play a weakening role:

$$\begin{aligned}
L_1 : \ln Y_{ij} &= \beta_{0j} + \beta_{nj}X_n + r_{ij}, \\
L_2 : \beta_{0j} &= \gamma_{00} + \gamma_{0m}W_m + \mu_{0j}, \\
\beta_{nj} &= \gamma_{n0} + \gamma_{nm}W_m + \mu_{nj},
\end{aligned} \tag{7}$$

where r_{nm} is the slope used by the m -th independent variable in the second layer to explain the slopes of the respective variables in the first layer, reflecting the cross-level interaction.

3. Result analysis.

3.1. ANOVA model analysis. The analysis results are shown in table 1. In the variance analysis, the between-group variance is 1,364 and is statistically significant at the 0.1% level, indicating that there are significant differences between the groups, with obvious stratification characteristics. The intra-group correlation coefficient $ICC = \text{Var}(U_0) / [\text{Var}(U_0) + \text{Var}(R)] = 1.364 / (1.364 + 0.370) = 78\%$ indicates that 78% of the differences between groups are caused by the group level, which is not negligible. Therefore, the multi-layer linear model can be used for analysis.

Table 1. Parameter Estimation

Random Effect	Standard Deviation	Variance Component	df	Chi-square	P-value
Intrept1, U_0	1.168	1.364	21	2032.411	0.000
level-1, R	0.608	0.370			

Table 2. HLM analysis of investment attractiveness

Fixed effect	Model 2		Model 3		Model 4	
	coefficient	standard error	coefficient	standard error	coefficient	standard error
x_1	-0.103	0.076	-0.155**	0.05	-0.089	0.050
x_2	1.559***	0.363	1.700***	0.436	0.975***	0.159
x_3	0.002***	0.0003	0.0002***	0.0003	0.00052***	0.0003
x_4			0.498***	0.117	1.818**	0.428
x_5			0.517***	0.107	-1.418**	0.466
x_1x_4			0.242*	0.116	0.374**	0.141
x_1x_5			-0.478***	0.136	-0.673***	0.154
x_2x_4			-0.136	0.147	1.392**	0.522
x_2x_5			0.292	0.193	-2.334***	0.572
x_3x_4			-0.008	0.008	-0.0015*	0.007
x_3x_5			-0.029**	0.011	-0.0015	0.009
Var(R)	0.712***		0.098***		0.468***	
level1	0.336		0.365		0.310	
Var(R)						

In table 2 “*” means $P < 0.05$, “**” means $P < 0.01$, “***” means $P < 0.001$; x_1 is cash income per capita; x_2 – volume of work performed for the type of activity “Construction”; x_3 – GRP(Gross Regional Product); x_4 – taxes levied; x_5 – total import and export trade.

3.2. Random coefficient regression model analysis. In the analysis of Model 2, Model 3, and Model 4, the paper expresses the explanatory variables’ variance components at the individual level.

It only shows the changes in the variance between groups and the first-level residuals. The fixed effect represents the influence effect of the independent variables of level 1 and level 2 on the investment attractiveness difference. The random effect means the investment attractiveness difference that the independent variable does not explain. The positive influence on investment attractiveness is x_1 , but it is not significant. The ones that have a significant negative impact on investment attractiveness are x_2 and x_3 .

3.3. Complete mode analysis. The direct effect of x_4 and x_5 is the significant positive effect of x_4 and x_5 . Under the cross-level interaction: x_4 has a moderately significant strengthening effect on the investment attractiveness of x_2 , indicating that the enhancement of x_4 can promote the influence of x_2 on investment attractiveness. In the same way, x_4 also played a highly significant strengthening effect on x_2 . And x_5 has a high significant weakening effect on x_1 and x_2 , and x_4 has a low significant weakening effect on x_3 .

4. Conclusion. The difference in investment attractiveness is the result of a combination of multiple factors. It is neither an individual regional issue nor a pure national policy issue, but a regional economic development, policy system, historical background and many other interrelated issues. All aspects are resolved together.

References

1. Bure V. M., Parilina E. M. Probability Theory and Mathematical Statistics. First Edition, SPb: "Lan", 2013, 416 p.
2. National Bureau of Statistics [Internet resource]: URL:<http://www.stats.gov.cn/tjsj/ndsj> (date: 15.02.21).
3. Qi D. Study of the investment attractiveness of China's regions // Control processes and stability. 2020. Vol. 7. No 1. P. 423–428.

UDC 519.2

Zhu W., Zhou G., Han M.

**The correlation study of average prices
of commodity housing in regional cities in China
based on archimedean copula function**

Recommended for publication by professor Krylatov A. Yu.

1. Introduction. Due to the steady increase in the average price of Chinese commodity housing, many scholars tend to analyze the factors of commodity housing and its time series, but there are insufficient studies on the correlation between the average housing prices of various provinces and cities. In this paper, we focus on the tail dependence which among five coastal developed and five inland less-developed provinces using the *archimedean copula* function.

2. Bivariate copula function. We consider two random variables X and Y , their marginal distribution functions are $F(x)$ and $G(y)$. Denote $H(x, y)$ as a joint distribution function of the variables. There must exist a function which can connect H with both F and G . It is named *copula*.

Definition 1. Bivariate function [1] $C(u, v)$ is a *bivariate copula function* and satisfies:

(1) Marginal condition:

$$C(u, 0) = C(0, v) = 0, \quad C(u, 1) = u, \quad C(1, v) = v;$$

(2) 2-increasing: For $\forall 0 \leq u_1 \leq u_2 \leq 1, 0 \leq v_1 \leq v_2 \leq 1$, there is

$$V_C([u_1, u_2] \times [v_1, v_2]) = C(u_2, v_2) - C(u_2, v_1) - C(u_1, v_1) \geq 0.$$

Note that $V_C([u_1, u_2] \times [v_1, v_2])$ as the second-order difference of the function C on the rectangle $[u_1, u_2] \times [v_1, v_2]$.

Zhu Wenguo – undergraduate student, Yanan University;
e-mail: 215198068@qq.com, phone: +8613379120501

Zhou Gaojuan – undergraduate student, Yanan University;
e-mail: 2546556117@qq.com, phone: +8615929582563

Han Mingxi – undergraduate student, Yanan University;
e-mail: mingxi_7@outlook.com, phone: +8615353405621

3. Archimedean copula function. First of all

Definition 2. Archimedean copula function expression [2]:

$$C(u_1, \dots, u_2, \dots, u_N) = \varphi^{-1}(\varphi(u_1) + \dots + \varphi(u_n) + \dots + \varphi(u_N))$$

Function $\varphi(\cdot)$ is named the generating function of *archimedean copula*, and satisfies:

$$\sum_{n=1}^N \varphi(u_n) \geq \varphi(0), \quad \varphi(1) = 0, \quad \forall 0 \leq u \leq 1, \quad \varphi'(u) < 0, \quad \varphi''(u) > 0.$$

The introductions of two copula functions are as follows:

Table 1. The introduction of two Copulas

Copula	$\varphi(t)$	$C(u, v; \alpha)$ & $C(u, v; \theta)$	α & θ
Gumbel Copula	$(-\ln t)^\alpha$	$C(u, v; \alpha) = \exp\{-[(-\ln u)^{\alpha} + (-\ln v)^{\alpha}]^{1/\alpha}\}$	$\alpha \in (0, 1]$
Clayton Copula	$t^{-\theta} - 1$	$C(u, v; \theta) = (u^{-\theta} + v^{-\theta} - 1)^{-1/\theta}$	$\theta \in (0, \infty)$

4. Principles and processes. We select GDP per capita of five coastal developed regions and five less-developed regions. We summarize the average price of commodity housing in recent two decades.

4.1. The establishment of the marginal distribution function. Archimedean copula function which performs better in tail dependence. Based on the historical data of commodity housing average prices. Taking Fujian as an example we could get a frequency histogram (figure 1):

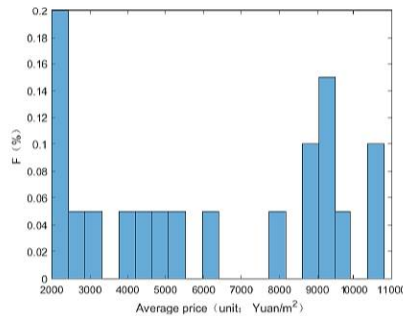


Figure 1. The frequency histogram

Taking Fujian as an example, we could get the mean and variance of three types of distribution. Then we estimate the three parameters when the significant levels 0.05, and we could get the results in table 2 are as follows:

Table 2. Three distribution functions and results

Types	Fuction	parameter	Sum of the Squared Errors	I
Normal	$\frac{1}{\sqrt{2\pi}\sigma} \int_a^x \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] dx$	$\mu = 6.1397$ $\sigma = 3.1916$	$\sum_{i=1}^n (F(x_i) - \bar{F}(x_i))^2$	0.1462
Uniform	$\begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & x \geq b \end{cases}$	$a = 0.2015$ $b = 1.0748$	$\sum_{i=1}^n (G(x_i) - \bar{G}(x_i))^2$	0.2266
Exponential	$\begin{cases} 1 - e^{-\lambda x}, & x \geq 0 \\ 0, & x < 0 \end{cases}$	$\lambda = 1.6288$	$\sum_{i=1}^n (H(x_i) - \bar{H}(x_i))^2$	0.2895

From the above three expressions, we choose the normal distribution to be the marginal distribution of Fujian's data. Similar to other data from the sample, the normal distribution is fit to the marginal distribution of Shanghai, Gansu, Sichuan and Yunnan, the uniform distribution for Zhejiang, and the exponential distribution for the marginal distribution of Jiangsu, Guangdong, Qinghai and Guizhou.

4.2. The establishment of joint distribution function. Taking the data of Fujian and Gansu as an example, let X be the Fujian's dataset and Y be Gansu's dataset, then we select the *Gumbel copula* function and the *Clayton copula* function to estimate the parameters, the steps are as follows.

Taking the *Gumbel copula* function as an example, the likelihood function of sample $x_i, y_i, i = 1, \dots, n$,

$$L_n(\alpha) = \prod_{i=1}^n c(F(x_i, \alpha), G(y_i, \alpha)) f(x_i, \alpha) g(y_i, \alpha),$$

where n represents the sample size. Its log-likelihood estimation:

$$\ln L_n(\alpha) = \sum_{i=1}^n \ln c(F(x_i, \alpha), G(y_i, \alpha)) + \sum_{i=1}^n \ln f(x_i, \alpha) + \sum_{i=1}^n \ln g(y_i, \alpha).$$

Taking the partial derivative of log-likelihood with respect with α . Let it equal 0, then we have: $\frac{\partial \ln L_n(\alpha)}{\partial \alpha} = 0$.

We get α 's maximum likelihood estimation: $\hat{\alpha} = \operatorname{argmax} L_n(\alpha)$. Then we introduce the *empirical copula* function.

Definition 3. Suppose $(x_i, y_i), i = 1, \dots, n$, is the sample originated to two-dimensional overall sample [3]. Denote the empirical distributions of X and Y are $F_n(x)$ and $G_n(y)$. Define the *empirical copula* function as follows:

$$\hat{C}_n(u, v) = \frac{1}{n} \sum_{i=1}^n I_{[F_n(x_i) \leq u]} I_{[G_n(y_i) \leq v]}, \quad u, v \in [0, 1].$$

$I_{[\cdot]}$ represents characteristic function, if $F_n(x_i) \leq u$, then $I_{[F_n(x_i) \leq u]} = 1$; if not, then $I_{[F_n(x_i) \leq u]} = 0$.

Now we consider the squared euclidean distance of *Gumbel copula* function $C_g(u, v)$ and *Clayton copula* function $C_c(u, v)$:

$$d_{C_g}^2 = \sum_{i=1}^n [\hat{C}_n(u_i, v_i) - \hat{C}_g(u_i, v_i)]^2, \quad d_{C_c}^2 = \sum_{i=1}^n [\hat{C}_n(u_i, v_i) - \hat{C}_c(u_i, v_i)]^2.$$

$d_{C_g}^2$ and $d_{C_c}^2$ show the original situation of data of the *Gumbel copula* function $C_g(u, v)$ and the *Clayton copula* function $C_c(u, v)$.

Table 3. The results

Regions	Gumbel Copula	$d_{C_g}^2$	Clayton Copula	$d_{C_c}^2$
Fujian & Jiangsu	6.0698	4.8156	13.3104	4.8491
Fujian & Shanghai	5.7299	4.6475	13.8979	4.7112
Fujian & Zhejiang	8.3275	4.9845	14.4848	4.9091
Fujian & Sichuan	16.1759	5.2240	32.3954	5.2009
...
Gansu & Jiangsu	7.5938	4.9646	11.3559	4.8193
Gansu & Shanghai	9.9405	5.0052	19.0011	4.9610

We refer the minimum squared euclidean distance as the optimal Copula function. According to the table 3, the *Gumbel copula* function fits better between Fujian and Jiangsu, Shanghai, Guangdong, Qinghai, Guizhou, Yunnan; the *Clayton copula* function fits better between Fujian and Zhejiang, Sichuan, also between Gansu and Jiangsu, Shanghai, Zhejiang, Guangdong, Qinghai, Sichuan, Yunnan, Guizhou.

4.3. The tail dependence. Taking Fujian and Gansu as an example, we fit the their corresponding probability density functions based on MATLAB (figure 2).

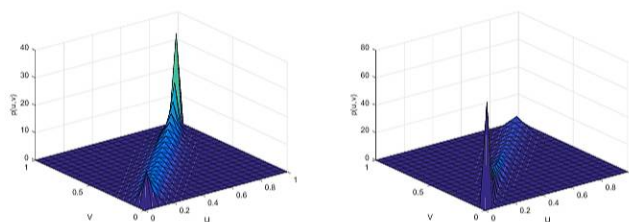


Figure 2. *Gumbel* and *Clayton* copula-corresponding probability density

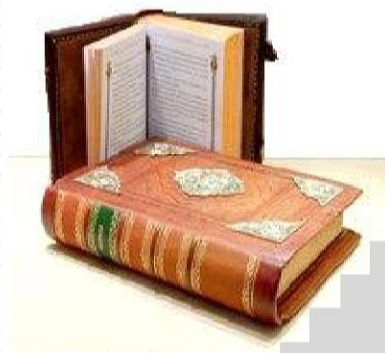
The image of the *Gumbel copula's* probability density function indicates the situation that upper-tail dependence is stronger, and the same case about lower-tail dependence is weaker for the *Clayton copula's*.

5. Conclusion and suggestion. The commodity housing average price of Fujian exhibits the upper-tail dependence with Jiangsu's, Shanghai's, Guangdong's, Gansu's, Qinghai's, Yunnan's and Guizhou's; also exhibits lower-tail dependence with Zhejiang's and Sichuan's. If the policy decision agencies would like to regulate the average price of Jiangsu (Zhejiang) and Shanghai (Sichuan) higher (lower), they could take measures to increase (decrease) the Fujian's. The commodity housing average price of Gansu exhibits the lower-tail dependence with the preceding samples. Then if the policy decision agencies would like to regulate the average price of Jiangsu and Shanghai lower, they could take measures to decrease the Gansu's.

References

1. Roger B. N. An Introduction to Copulas. Springer Series in Statistics. Springer-Verlag New York, 2006. 286 p.
2. Christian G., Jock M. The joy of copulas: bivariate distributions with uniform marginals // The American Statistician. 2012. Vol. 40. No 4. P. 280–283.
3. Xibo L., Zeng W., Yanhua G. The research on sequence of daily return correlation between Shanghai-Shenzhen stock index based on copula function // Mathematics in Practice and Theory. 2015. Vol. 45. No 11. P. 101–108.

*6. Прикладная
математика и
процессы управления
на факультете
ПМ–ПУ СПбГУ*



УДК 517.977

Погожев С. В., Смирнов Н. В.

Памяти профессора Евгения Игоревича Веремея



Евгений Игоревич скончался совершенно неожиданно – 1 марта 2021 г. на 72-м году жизни. Для факультета ПМ–ПУ это большая утрата. Для всех нас он имел непререкаемый авторитет в науке, был добрым и отзывчивым коллегой, организатором научных исследований и учебного процесса, для многих – учителем во всех смыслах этого слова. О таких людях обычно говорят: «Он не ушел в прошлое – он растворился в будущем...». Научное наследие Евгения Игоревича еще требует серьезного осмысления, мы обязательно вернемся к его обсуждению на будущих научных конференциях. Эта же небольшая заметка – дань памяти человеку, посвятившему большую часть своей жизни науке и нашему родному факультету.

Погожев Сергей Владимирович – доцент, Санкт-Петербургский государственный университет; e-mail: s.pogozhev@spbu.ru, тел.: +7(812)428-44-76

Смирнов Николай Васильевич – профессор, Санкт-Петербургский государственный университет; e-mail: n.v.smirnov@spbu.ru, тел.: +7(812)428-41-54

Евгений Игоревич Веремей в 1973 г. окончил Днепропетровский государственный университет и с 1975 г. начал трудовой путь в Ленинградском (Санкт-Петербургском) государственном университете, где последовательно занимал должности младшего, старшего научного сотрудника, доцента, профессора [1]. В 1977 г. Е. И. Веремей поступил в аспирантуру ЛГУ, в 1980 г. защитил кандидатскую диссертацию «Синтез оптимальных регуляторов с учетом требований реализации», в 1995 г. – докторскую диссертацию «Методы и алгоритмы среднеквадратичного многоцелевого синтеза». В 2002 г. по его инициативе на факультете ПМ – ПУ была открыта кафедра компьютерных технологий и систем, которой Евгений Игоревич заведовал до настоящего времени.

Евгений Игоревич Веремей – признанный специалист в области теории управления и ее технических приложений, ученый, широко известный в России и за рубежом. Основным направлением его фундаментальных исследований была разработка математических методов и вычислительных алгоритмов автоматизированного синтеза и компьютерного моделирования цифровых систем управления динамическими объектами. В центре прикладных разработок находились такие сферы, как энергетика и судостроение. Среди его важнейших научных результатов – формирование новой концепции исследования и проектирования многоцелевых цифровых обратных связей с оптимизацией по нормам соответствующих функциональных пространств. В рамках этой концепции созданы многочисленные методы, расчетные схемы и реализующие их программные инструменты. В области энергетики были получены новые результаты по решению задач многоцелевого управления формой и током плазмы в токамаках. В области судостроения разработан широкий спектр аналитических и вычислительных методов анализа, синтеза и моделирования управляющих алгоритмов для систем управления движением морских объектов различных классов. Евгений Игоревич занимался внедрением своих подходов в различные сферы цифровой экономики, включая технологические процессы в робототехнике и нефтехимии. Он являлся автором более 200 научных работ, изданных в России и за рубежом, в том числе 7 монографий и учебных пособий.

Большое внимание Евгений Игоревич уделял учебной и учебно-методической работе. Он создал известную научно-педагогическую школу в области прикладной математики и процессов управления. Под его руководством успешно защищены 17 кандидатских и од-

на докторская диссертация. Евгений Игоревич был одним из разработчиков программ бакалавриата по направлению «Фундаментальная информатика и информационные технологии» и разработчиком актуальной магистерской образовательной программы «Цифровые технологии и системы», являлся членом Президиума ФУМО при Минобрнауки РФ по укрупненной группе направлений 02 «Информатика и компьютерные науки».

Е. И. Веремей активно занимался экспертной деятельностью по соответствующим научным направлениям – входил в состав диссертационных советов при СПбГУ и других учебных и научных организациях, участвовал в работе редколлегий научных журналов («Вестник СПбГУ. Серия 10: Прикладная математика. Информатика. Процессы управления», Cloud of Science, International Journal of Open Information и др.), оргкомитетов международных научных конференций (Computer Technologies in Physical and Engineering Applications, «Современные информационные технологии и ИТ-образование», «Конвергентные когнитивно-информационные технологии» и др.). С 2004 г. Евгений Игоревич Веремей являлся действительным членом международной Академии навигации и управления движением (Санкт-Петербург).

Заслуги Е. И. Веремея, как ученого и преподавателя, отмечены наградами: в 2003 г. – нагрудным знаком «Почетный работник высшего профессионального образования РФ», в 2007 г. – дипломом и премией СПбГУ за педагогическое мастерство.

В рамках ЛП Международной научной конференции аспирантов и студентов «Процессы управления и устойчивость» состоялся круглый стол «Памяти профессора Е. И. Веремея» [2], где коллеги поделились своими воспоминаниями. Светлая память о Евгении Игоревиче сохранится в сердцах всех, кто его знал.

Литература

1. Веремей Евгений Игоревич // Они работали с нами. Официальный сайт факультета ПМ–ПУ [Электронный ресурс]: URL: <http://www.apmath.spbu.ru/ru/info/veremey/> (дата обращения: 24.06.2021).
2. Круглый стол «Памяти профессора Е. И. Веремея». Конференция CPS [Электронный ресурс]: URL: <https://youtu.be/KMdeIhs0VdQ> (дата обращения: 24.06.2021).

Contents

Plenary Session	9
Invited speakers	9
<i>Tamasyan G. Sh.</i> About constructive nonsmooth analysis	11
1. Mathematical theory of control processes	37
<i>Alkin R. V.</i> Decentralized control of the location of objects on a straight line, taking into account their size	37
<i>Belov A. I.</i> Stability criterion for linear system with delay	45
<i>Evtina D. S.</i> On the robust stability of linear systems with delay ..	50
<i>Zhigalov V. S.</i> Conditions for the pointwise controllability of the difference-differential system with linearly increasing delay	55
<i>Zholobov Y. V.</i> The quadcopter emergency landing place predicting	61
<i>Ivanov A. V., Moiseev I. A.</i> A special case of optimal control synthesis in the flight problem	66
<i>Kudryakov D. A.</i> Stability analysis of linear time delay systems with uncertain parameters	74
<i>Mironov A. S.</i> A generalization of the Danskin problem	79
<i>Kastner A., Gröll L., Hagenmeyer V.</i> Abstractions in modeling of switching DC/AC power converters	85
<i>Wachter J., Gröll L., Hagenmeyer V.</i> Realtime testbench for inverters – a sliding mode control experiment	93
2. Mathematical methods in mechanics and physics	101
<i>Alcybeev G. O., Burova I. G.</i> Gas-turbine engine and spline approximations	101
<i>Vakaeva A. B., Shuvalov G. M., Kostyrko S. A.</i> Morphological instability of a nanovoid in a solid during surface diffusion	108
<i>Vinogradova E. M., Doronin G. G.</i> Calculation of the potential distribution in a two-dimensional diode system with a blade-shaped field cathode	113
<i>Dem'yanovich Yu. K., Burova I. G., Doronina A. G., Miroshnitchenko I. D.</i> On approximation by the finite element method in a flat region	117

<i>Zubakhina T. S.</i> Stability analysis of the Runge–Kutta–Chebyshev methods of the second order for equations with delay	126
<i>Kazankov V. K., Kholodova S. E.</i> Investigation and analysis of the equation of the theory of an ideal incompressible electrically conducting fluid	131
<i>Onegov N. A.</i> Analysis of the influence of internal waves on the spill process and on the main characteristics of the fluid in the oil-bearing formation	138
<i>Peglivanova M. M.</i> Corrosion of the plate made of elastic-plastic material with SD effect	144
<i>Rybalkin A. A.</i> Mathematical modeling of a balancing robot on a spherical wheel	149
<i>Hoang V. N.</i> Differential-difference equation with distributed parameters on a graph	155
<i>Shuvalov G. M., Kostyrko S. A.</i> Effect of surface elasticity on morphological stability of ultrathin film surface	161
<i>Schafberger J., Hagenmeyer V., Gröll L.</i> Modeling of a distillation column without measurement data	167

3. Mathematical models of medical and biological systems 175

<i>Bartashuk A. E.</i> Predicting the Gleason score of prostate cancer using an artificial neural network	175
<i>Belavin S. A.</i> Using cluster analysis methods for diagnosing brain diseases	182
<i>Buchina D. A., Goncharova A. B.</i> Processing medical research data using propensity score matching (PSM)	187
<i>Vil' M. Yu., Goncharova A. B.</i> Usage of MATLAB SimBiology application for oncological disease modeling taking into consideration interference competition	192
<i>Larochkin P. V.</i> Research of MRI images of the brain	197
<i>Leonova E. B., Kotina E. D.</i> Displacement field construction for image sequences based on discrete systems	202
<i>Moiseeva V. D.</i> Analysis of one multi-virus epidemic model	207
<i>Petrova O. G.</i> Segmentation of MRI images of the spine using convolutional neural networks	214
<i>Tikhonova E. O.</i> Statistical analysis of changes in intraocular pressure after intravitreal injections	219
<i>Hao Q.</i> Finding the projections of quadratic manifolds	224

Chubarov D. A. Mathematical modeling of the intervertebral disc using the numerical method 233

4. Information and computer technologies 239

Akopyan A. S. The improvement of methods for extraction word alignment in the neural machine translation task 239

Afanasov A. K., Tsoy A. I. WhatsApp backup acquisition and decryption for Android platform 246

Gladyshev T. Yu. Single-image 3D shape reconstruction with deep neural networks 254

Grashin D. M. Determination the texture size for the reconstructed model in Agisoft MetaShape application 259

Zhereb V. V. Implementing RAID algorithms for the ARM processor 265

Ilchenko I. I. Embox OS network namespace implementation 274

Kvanchiani K. Sh. Improving the quality of medical images in the context of tumor segmentation task 279

Knyazev N. A. Determining the gender and age of people from a photograph of a face using neural networks 284

Kulminskiy D. P., Efimov V. V. Semi-supervised segmentation with neural networks for core sample condition assessment 289

Menshikov M. A. Systematization of abstractions to support various programming languages and environments in static code analyzers 294

Nikolaeva I. N. Container images minimization of application and using Docker as an example 303

Pavlova E. D., Chebotareva A. A., Cepeleva R. V. Solving the problem of automated marking of underwater animals using a cascade of neural networks 308

Ruschenko V. V. Development of an algorithm for ordering superimposed contours 313

Savchenko A. V. Comparison of the access control system based on intermediate user groups with the standard access control system without intermediate groups 318

Sermyagin N. V. People detection in a video stream using a motion detector and a convolutional neural network 321

Sorokina S. O., Mikhailova E. G. Hyperparameters optimization of gradient boosting algorithms 326

<i>Tereshchenko D. V.</i> Programming interface for fault-tolerant distributed computing in Python.....	332
<i>Faradzhov A. M.</i> Comparative analysis of DLT solutions based on the PBFT algorithm.....	337
<i>Tsepeleva R. V., Pavlova E. D., Chebotareva A. A.</i> Implementation of the cross-blockchain interacting protocol.....	342
<i>Bai M., Li S.</i> Comparative study of various edge detection operator.....	347
<i>Kimura K., Tu Y., Tanji R., Mozgovoy M.</i> Identifying winning actions in a 3D tennis game with Monte Carlo tree search.....	352
<i>Li Y., Fan F.</i> Design and implementation of online education system based on B2C mode.....	358
<i>Liu Y., Chang G.</i> Implementation of facial mask detection in tensorflow based on improved SSD algorithm.....	363
<i>Qiao W., Zhang J.</i> Image fusion based on transform domain.....	368
<i>Su S., Blekanov I.</i> Neural network methods for messages analysis in user discussions on YouTube.....	372
<i>Tang Z., Pyshkin E.</i> Managing notifications within the context of soft planning: visual approach to requirement analysis.....	377
<i>Yang W., Li P.</i> Research on sentiment analysis of network public opinion based on BI-LSTM.....	387
<i>Yuda K., Otomo K., Ito R., Wakana I., Tanji R., Kamei S., Mozgovoy M.</i> Behavior similarity between human players and built-in AI characters in universal fighting engine.....	392
<i>Zhao C., Blekanov I.</i> Two towers collaborative filtering algorithm for movie recommendation.....	397

5. Control problems in social and economical systems ... 403

<i>Berestyukova V. M.</i> Bayesian equilibrium in Bertrand duopoly ...	403
<i>Bulgakova M. A.</i> The τ value for the network games with pairwise interactions.....	408
<i>Busel V. D., Lazareva A. V.</i> The use of various centrality measures in the problem of optimizing the transmission of information in self-organizing networks.....	413
<i>Kocherov I. P.</i> Analysis of daily COVID-19 infection data.....	418
<i>Magnitskaya N. G.</i> Superadditive expansion of the characteristic function in the investment management game.....	424

<i>Markelova A. Yu., Laisceva M. Yu., Petrosian O. L.</i> Microgrid management with energy storage system and renewable energy generation	430
<i>Nechiporuk A. A.</i> Equivalence of models of opinions dynamics in social networks.....	435
<i>Nikitina A. A.</i> Estimation of model parameters in SQAIRD epidemic model with time intervals.....	440
<i>Papernyuk A. O.</i> Application of ALS in collaborative filtering ...	450
<i>Savin K. A., Gromova E. V.</i> On the properties of characteristic functions in a game with multilateral externalities	455
<i>Sachkov A. V., Shcherbakov G. A.</i> Discussion of potential application of axiomatic choice theory in business	460
<i>Sokolova I. S.</i> Implementation of algorithms for solving problems of multi-product optimization of inventory management with shortages and discounts	466
<i>Stashkov E. A.</i> Analysis of a consense maintaining problem for directed graphs in discrete time	473
<i>Tretiakova S. S.</i> On the solving structure of the optimal control problem of production volumes.....	478
<i>Chebotareva A. A., Pavlova E. D., Tsepeleva R. V.,</i> A linear-quadratic differential game of harmful emissions volumes control with a Bolz's functional	484
<i>Dan W.</i> Analysis of static game and repeated game in tax collection and management	489
<i>Fernández-Arias E., López-Barrientos J. D., Moreno-Ruíz-Esparza A.</i> A contingent claims approach for computing the premium to ensure an investment in highways	494
<i>Gao J., Parilina E.</i> Average-oriented opinion dynamics with the last moment of observation	505
<i>He Y.</i> Analysis of the most important factors of China's air quality index	510
<i>Qi D.</i> Research on factors affecting differences in investment attractiveness based on HLM model.....	515
<i>Zhu W., Zhou G., Han M.</i> The correlation study of average prices of commodity housing in regional cities in China based on archimedean copula function	520

6. Applied mathematics and control processes theory at the AM & CP Faculty of Saint Petersburg State University	525
<i>Pogozhev S. V., Smirnov N. V.</i> In memory of professor Evgeny Igorovich Veremey	525
Contents	528
Author index.....	534

Именной указатель

Акопян А. С.	239	Евтина Д. С.	50
Алцыбеев Г. О.	101	Ефимов В. В.	289
Алькин Р. В.	37	Жереб В. В.	265
Афанасов А. К.	246	Жигалов В. С.	55
Балыкина Ю. Е.	11	Жолобов Е. В.	61
Барташук А. Е.	175	Зубахина Т. С.	126
Белавин С. А.	182	Иванов А. В.	66
Белов А. И.	45	Ильченко И. И.	274
Берестюкова В. М.	403	Казанков В. К.	131
Булгакова М. А.	408	Кванчиани К. Ш.	279
Бурова И. Г.	101, 117	Князев Н. А.	284
Бусел В. Д.	413	Костырко С. А.	108, 161
Бучина Д. А.	187	Котина Е. Д.	202
Вакаева А. Б.	108	Кочеров И. П.	418
Виль М. Ю.	192	Кудряков Д. А.	74
Виноградова Е. М.	113	Кульминский Д. П.	289
Гладышев Т. Ю.	254	Лазарева А. В.	413
Гончарова А. Б.	187, 192	Лаисцева М. Ю.	430
Грашин Д. М.	259	Ларочкин П. В.	197
Громова Е. В.	455	Леонова Е. Б.	202
Демьянович Ю. К.	117	Магнитская Н. Г.	424
Доронин Г. Г.	113	Маркелова А. Ю.	430
Доронина А. Г.	117	Меньшиков М. А.	294

Миронов А. С.	79	Сташков Е. А.	473
Мирошниченко И. Д.	117	Тамасян Г. Ш.	9, 13
Михайлова Е. Г.	326	Терещенко Д. В.	332
Моисеев И. А.	66	Тихонова Е. О.	219
Моисеева В. Д.	207	Третьякова С. С.	478
Нечипорук А. А.	435	Фараджов А. М.	337
Никитина А. А.	440	Хао Ц.	224
Николаева И. Н.	303	Хоанг В. Н.	155
Онегов Н. А.	138	Холодова С. Е.	131
Павлова Е. Д.	308, 342, 484	Цепелева Р. В.	308, 342, 484
Папернюк А. О.	450	Цой А. И.	246
Пегливанова М. М.	144	Чеботарева А. А.	308, 342, 484
Петрова О. Г.	214	Чубаров Д. А.	233
Петросян О. Л.	430	Шувалов Г. М.	108, 161
Погожев С. В.	525	Щербаков Г. А.	460
Рущенко В. В.	313	Vai M.	347
Рыбалкин А. А.	149	Vlekanov I.	372, 397
Савин К. А.	455	Chang G.	363
Савченко А. В.	318	Dan W.	489
Сачков А. В.	460	Fan F.	358
Сермягин Н. В.	321	Fernández-Arias E.	494
Смирнов Н. В.	525	Gao J.	505
Соколова И. С.	466	Gröll L.	85, 93, 167
Сорокина С. О.	326	Hagenmeyer V.	85, 93, 167

Han M.	520	Qi D.	515
He Y.	510	Qiao W.	368
Ito R.	392	Schaßberger J.	167
Kamei S.	392	Su S.	372
Kastner A.	85	Tang Z.	377
Kimura K.	352	Tanji R.	352, 392
Li P.	387	Tu Y.	352
Li S.	347	Wachter J.	93
Li Y.	358	Wakana I.	392
Liu Y.	363	Yang W.	387
López-Barrientos J. D.	494	Yuda K.	392
Moreno-Ruíz-Esparza A.	494	Zhang J.	368
Mozgovoy M.	352, 392	Zhao C.	397
Otomo K.	392	Zhou G.	520
Parilina E.	505	Zhu W.	520
Pyshkin E.	377		

Для заметок

Ежегодное научное издание

ПРОЦЕССЫ УПРАВЛЕНИЯ И УСТОЙЧИВОСТЬ

Том 8(24)

Труды 52-й международной научной конференции
аспирантов и студентов

Редакторы текста на русском языке: *А. С. Ерёмин, Т. Е. Смирнова*

Редакторы текста на английском языке: *А. С. Ерёмин, М. А. Смирнова*

Компьютерная верстка: *М. Н. Смирнов, М. А. Смирнова*

Подписано в печать 24.06.21
Усл. печ. л. 31,2. Тираж 200 экз.

198504, Россия, Санкт-Петербург, Петергоф, Университетский просп., 35