

ISSN 2305-9052 (Print)
ISSN 2410-7034 (Online)

ВЕСТНИК



ЮЖНО-УРАЛЬСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА

BULLETIN

OF THE SOUTH URAL
STATE UNIVERSITY

СЕРИЯ

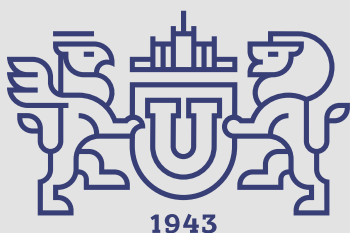
**ВЫЧИСЛИТЕЛЬНАЯ
МАТЕМАТИКА
И ИНФОРМАТИКА**

2020, том 9, № 4

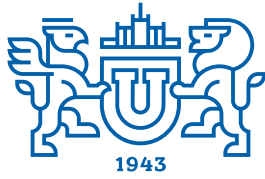
SERIES

**COMPUTATIONAL
MATHEMATICS
AND SOFTWARE ENGINEERING**

2020, volume 9, no. 4



ВЕСТНИК



ЮЖНО-УРАЛЬСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА

2020
Т. 9, № 4

ISSN 2305-9052 (Print)
ISSN 2410-7034 (Online)

СЕРИЯ

«ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»

Решением ВАК включен в Перечень научных изданий,
в которых должны быть опубликованы результаты диссертаций
на соискание ученых степеней кандидата и доктора наук

Учредитель — Федеральное государственное автономное образовательное учреждение
высшего образования «Южно-Уральский государственный университет
(национальный исследовательский университет)»

Тематика журнала:

- Вычислительная математика и численные методы
- Математическое программирование
- Распознавание образов
- Вычислительные методы линейной алгебры
- Решение обратных и некорректно поставленных задач
- Доказательные вычисления
- Численное решение дифференциальных и интегральных уравнений
- Исследование операций
- Теория игр
- Теория аппроксимации
- Информатика
- Искусственный интеллект и машинное обучение
- Системное программирование
- Перспективные многопроцессорные архитектуры
- Облачные вычисления
- Технология программирования
- Машинная графика
- Интернет-технологии
- Системы электронного обучения
- Технологии обработки баз данных и знаний
- Интеллектуальный анализ данных

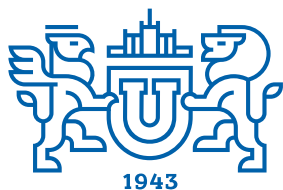
Редакционная коллегия

Л.Б. Соколинский, д.ф.-м.н., проф., *гл. редактор*
В.П. Танана, д.ф.-м.н., проф., *зам. гл. редактора*
М.Л. Цымблер, д.ф.-м.н., доц., *отв. секретарь*
Г.И. Радченко, к.ф.-м.н., доц.
Я.А. Краева, *техн. секретарь*

Редакционный совет

С.М. Абдуллаев, д.г.н., профессор
А. Андреяк, PhD, профессор (Германия)
В.И. Бердышев, д.ф.-м.н., акад. РАН, *председатель*
В.В. Воеводин, д.ф.-м.н., чл.-кор. РАН

Дж. Донгарра, PhD, профессор (США)
С.В. Зыкин, д.т.н., профессор
Д. Маллманн, PhD, профессор (Германия)
А.В. Панюков, д.ф.-м.н., профессор
Р. Продан, PhD, профессор (Австрия)
А.Н. Томилин, д.ф.-м.н., профессор
В.Е. Третьяков, д.ф.-м.н., чл.-кор. РАН
В.И. Ухоботов, д.ф.-м.н., профессор
В.Н. Ушаков, д.ф.-м.н., чл.-кор. РАН
М.Ю. Хачай, д.ф.-м.н., профессор
А. Черных, PhD, профессор (Мексика)
П. Шумяцкий, PhD, профессор (Бразилия)



BULLETIN

OF THE SOUTH URAL
STATE UNIVERSITY

2020

Vol. 9, no. 4

SERIES

“COMPUTATIONAL
MATHEMATICS AND SOFTWARE
ENGINEERING”

ISSN 2305-9052 (Print)
ISSN 2410-7034 (Online)

Vestnik Yuzhno-Ural'skogo Gosudarstvennogo Universiteta.
Seriya “Vychislitel'naya Matematika i Informatika”

South Ural State University

The scope of the journal:

- Numerical analysis and methods
- Mathematical optimization
- Pattern recognition
- Numerical methods of linear algebra
- Reverse and ill-posed problems solution
- Computer-assisted proofs
- Numerical solutions of differential and integral equations
- Operations research
- Game theory
- Approximation theory
- Computer science
- Artificial intelligence and machine learning
- System software
- Advanced multiprocessor architectures
- Cloud computing
- Software engineering
- Computer graphics
- Internet technologies
- E-learning
- Database processing
- Data mining

Editorial Board

L.B. Sokolinsky, South Ural State University (Chelyabinsk, Russia)
V.P. Tanana, South Ural State University (Chelyabinsk, Russia)
M.L. Zymbler, South Ural State University (Chelyabinsk, Russia)
G.I. Radchenko, South Ural State University (Chelyabinsk, Russia)
Ya.A. Kraeva, South Ural State University (Chelyabinsk, Russia)

Editorial Council

S.M. Abdullaev, South Ural State University (Chelyabinsk, Russia)
A. Andrzejak, Heidelberg University (Germany)
V.I. Berdyshev, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)
J. Dongarra, University of Tennessee (USA)
M.Yu. Khachay, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)
D. Mallmann, Julich Supercomputing Centre (Germany)
A.V. Panyukov, South Ural State University (Chelyabinsk, Russia)
R. Prodan, University of Innsbruck (Innsbruck, Austria)
P. Shumyatsky, University of Brasilia (Brazil)
A. Tchernykh, CICESE Research Center (Mexico)
A.N. Tomilin, Institute for System Programming of the RAS (Moscow, Russia)
V.E. Tretyakov, Ural Federal University (Yekaterinburg, Russia)
V.I. Ukhobotov, Chelyabinsk State University (Chelyabinsk, Russia)
V.N. Ushakov, Institute of Mathematics and Mechanics, Ural Branch of the RAS (Yekaterinburg, Russia)
V.V. Voevodin, Lomonosov Moscow State University (Moscow, Russia)
S.V. Zykin, Sobolev Institute of Mathematics, Siberian Branch of the RAS (Omsk, Russia)

Содержание

КОНЦЕПЦИЯ ПОСТРОЕНИЯ ЦИФРОВОГО ДВОЙНИКА ГОРОДА С.А. Иванов, К.Ю. Никольская, Г.И. Радченко, Л.Б. Соколинский, М.Л. Цымблер	5
ЧИСЛЕННОЕ МОДЕЛИРОВАНИЕ НЕУСТОЙЧИВЫХ РЕЖИМОВ ОРЕГОНАТОРА Л.А. Прокудина, С.У. Турлакова	24
О ЗАДАЧЕ ПО ВОССТАНОВЛЕНИЮ КОЭФФИЦИЕНТОВ-ФУНКЦИЙ ВРЕМЕНИ ИСТОЧНИКОВ СПЕЦИАЛЬНОГО ВИДА В ПАРАБОЛИЧЕСКОМ УРАВНЕНИИ А.Б. Рагимов	33
ДВУМЕРНАЯ МОДЕЛЬ ФАСЕТОЧНОГО ЗРЕНИЯ Р.С. Федянина, Л.Б. Соколинский	48
ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ ОСЕСИММЕТРИЧНЫХ СТРУЙНЫХ ТЕЧЕНИЙ НА ОСНОВЕ ТУРБУЛЕНТНОЙ МОДЕЛИ ν_t -92 М.Э. Мадалиев	67
ОБЗОР СОВРЕМЕННЫХ СИСТЕМ ОБРАБОТКИ ВРЕМЕННЫХ РЯДОВ Е.В. Иванова, М.Л. Цымблер	79

Contents

DIGITAL TWIN OF A CITY: CONCEPT OVERVIEW S.A. Ivanov, K.Yu. Nikolskaya, G.I. Radchenko, L.B. Sokolinsky, M.L. Zymbler	5
NUMERICAL SIMULATION OF UNSTABLE OREGONATOR REGIMES L.A. Prokudina, S.U. Turlakova	24
ON THE PROBLEM OF RESTORING TIME COEFFICIENT-FUNCTIONS OF SPECIAL TYPE SOURCE IN PARABOLIC EQUATION A.B. Rahimov	33
TWO-DIMENSIONAL MODEL OF FACET VISION R.S. Fedyanina, L.B. Sokolinsky	48
NUMERICAL RESEARCH ν_t -92 TURBULENCE MODEL FOR AXISYMMETRIC JET FLOW M.E. Madaliev	67
OVERVIEW OF MODERN TIME SERIES MANAGEMENT SYSTEMS E.I. Ivanova, M.L. Zymbler	79



This issue is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

КОНЦЕПЦИЯ ПОСТРОЕНИЯ ЦИФРОВОГО ДВОЙНИКА ГОРОДА

© 2020 С.А. Иванов, К.Ю. Никольская, Г.И. Радченко,
Л.Б. Соколинский, М.Л. Цымблер

*Южно-Уральский государственный университет
(454080 Челябинск, пр. им. В.И. Ленина, д. 76)*

*E-mail: saivanov@susu.ru, nikolskaiaki@susu.ru, gleb.radchenko@susu.ru,
leonid.sokolinsky@susu.ru, mzym@susu.ru*

Поступила в редакцию: 15.09.2020

В данной статье представлена концепция построения цифрового двойника города. Городское хозяйство представляет собой сложную многовекторную систему, создание единого цифрового двойника которой в настоящее время является трудно решаемой задачей. Авторами предлагается эволюционный подход к решению этой задачи, в соответствии с которым на единой программно-аппаратной платформе последовательно строятся цифровые двойники отдельных элементов городской среды. Эти цифровые двойники связываются в единую кооперативную систему, позволяющую одним цифровым двойникам использовать данные, производимые другими цифровыми двойниками. В статье дается определение и архитектура такой системы. Описываются классы моделей, которые могут использоваться для создания цифровых двойников. Особое внимание уделяется нейросетевым моделям и моделям для анализа данных. Рассматривается информационная инфраструктура цифрового двойника города, включающая в себя сенсорные сети, механизмы очистки данных и туманные вычисления.

Ключевые слова: цифровой двойник, умный город, городское управление, нейронные сети, интеллектуальный анализ данных, сенсоры, туманные вычисления.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Иванов С.А., Никольская К.Ю., Радченко Г.И., Соколинский Л.Б., Цымблер М.Л. Концепция построения цифрового двойника города // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 5–23. DOI: 10.14529/cmse200401.

Введение

Сегодня главной движущей силой социально-экономического развития становятся информационные технологии. Инновационные способы сбора и анализа данных постепенно занимают место устоявшихся механизмов управления городом. В отличие от статистических выборок, которые успевают устареть к моменту их анализа, «большие данные» могут обрабатываться в режиме реального времени, что повышает качество и скорость принятия решений. «Большие данные» в области городского управления дополняют традиционные типы информации о городе и расширяют сферу их применения. Так, благодаря «большим данным» стал возможен мониторинг поведенческих моделей и анализ городского образа жизни на пересечении таких привычных категорий как население, экономическое развитие, застройка и инфраструктура и др. Цифровая революция и повсеместное распространение интернета породили новый феномен — Data Driven City (город, управляемый данными) [3] и близкий ему термин Smart City (Умный город). Умный город можно определить как стратегический подход к интеграции данных и цифровых технологий для обеспечения устойчивости, благосостояния граждан и экономического развития городской среды [40]. Концепция Умного города определяет концепцию пространства, в котором ключевые компоненты городской инфраструктуры — окружающая среда, борьба

с чрезвычайными ситуациями, управление дорожным движением и энергетика — интегрированы таким образом, что их функции и возможности могут быть легко объединены как друг с другом, так и с новыми системами [33].

В Министерстве строительства России совместно с МГУ им. М.В. Ломоносова разработан индекс цифровизации городского хозяйства «IQ городов» в рамках проекта «Умный город», который реализуется в рамках двух национальных проектов — «Жилье и городская среда» и «Цифровая экономика». «IQ городов» рассчитывается по десяти направлениям (городское управление, умное ЖКХ, инновации для городской среды, умный городской транспорт, интеллектуальные системы общественной и экологической безопасности, туризм и сервис, интеллектуальные системы социальных услуг, экономическое состояние и инвестиционный климат, инфраструктура сетей связи) и содержит 47 показателей [2]. Одним из важных индикаторов «IQ городов» является наличие цифрового двойника города.

Концепция «Цифрового двойника» (ЦД) обеспечивает создание и поддержку виртуальных моделей объектов и процессов реального мира. Подход цифровых двойников ориентируется на возможность получения и эффективной обработки потоков данных, собираемых автоматически посредством распределенных сенсорных систем «Интернета вещей» (Internet of Things, IoT). «Цифровой двойник» города постепенно наполняется данными реального города, собираемыми в реальном времени из развернутой IoT-инфраструктуры и городских информационных систем. «Цифровой двойник», анализируя информацию о состоянии городской инфраструктуры, динамике движения людей и транспорта, их взаимозависимость, а также их колебания во времени и пространстве, способен прогнозировать изменения состояния систем городского хозяйства и предлагать оптимальные решения. Кроме того, независимо от текущего состояния, цифровой двойник позволит отвечать на вопросы вида «что, если», помогая аналитикам понять, как города, оснащенные интеллектуальными технологиями, будут функционировать в тех или иных экономических, экологических и социальных условиях, и определить факторы, способствующие возможным сбоям [32]. В статье [4] рассматриваются результаты анализа, проведенного Градостроительным институтом пространственного моделирования и развития «Гипрогор Проект», в котором были рассмотрены 10 наиболее комплексных решений по созданию цифровых двойников городов. В список вошли Сингапур, Амаравати, Бостон, Ньюкасл, Джайпур, Хельсинки, Роттердам, Стокгольм, Ренн и Антверпен. Как отмечается в работе, Россию также ждет неотвратимый процесс цифровизации городов, так как это ведет к существенному росту качества жизни граждан. Среди российских решений, приближающихся к видению «Умного города» как цифрового сервиса, обеспечивающего качественно новый уровень сервиса для жителей города, можно отметить проект «Умные города Росатома» [5]. В рамках этого проекта, на базе единой ГИС-платформы сформированы решения по предоставлению жителям сервисов по поиску актуальной информации, отслеживанию работы городских служб, сбора обратной связи о текущих городских проблемах и др.

Целью данной статьи является исследование существующих технологий построения цифровых двойников городов. Для этого нам необходимо определить понятие цифрового двойника города, проанализировать методы моделирования, которые могут быть применены для построения цифровых двойников, а также систематизировать информационную инфраструктуру, которая должна обеспечивать функционирование цифрового двойника города. Статья имеет следующую структуру. В разделе 1 дается определение цифрового двойника города, описывается его архитектура и функциональные требования. В раз-

деле 2 рассматриваются классы моделей, применяемые при создании цифровых двойников. В разделе 3 дано описание информационной инфраструктуры цифрового двойника города, включающее в себя сенсорные сети, очистку данных и туманные вычисления. В заключении суммируются полученные результаты.

1. Цифровой двойник города: определение, структура, функционал

Цифровой двойник (ЦД) — это интегрированная мульти-физическая, мульти-масштабная вероятностная симуляция сложного объекта, которая использует физические, математические, имитационные и другие модели, для того чтобы получить как можно более достоверное представление о соответствующем реальном объекте на основе анализа данных, поступающих из сенсорных сетей и других источников [21].

Цифровой двойник города — это комплекс взаимосвязанных цифровых двойников, представляющих определенные аспекты функционирования и развития городской среды. Эти цифровые двойники поддерживают возможности тонкой подстройки и синхронизации с реальным состоянием городской инфраструктуры посредством данных, поступающих из различных источников в режиме реального времени [32, 38]. Основой для эффективного функционирования цифрового двойника города является непрерывный поток данных, генерируемых различными источниками в цифровой инфраструктуре умного города. Как пример таких источников можно выделить следующие (см. рис. 1).



Рис. 1. Архитектура цифрового двойника города

- *Данные о передвижении жителей города*, включая информацию о перемещении частного, коммерческого и общественного транспорта, а также загруженности улиц, собираемые посредством различных механизмов (данные о транзакциях «единого проездного билета», результаты мониторинга дорожного движения и др.).
- *Информация о физических параметрах городской среды*, получаемых в реальном времени с массивов интеллектуальных датчиков (как частных, так и общественных), позволяет осуществлять мониторинг и анализ таких параметров как температура и влажность воздуха, количество взвешенных частиц и химический состав воздуха, шумовая загрязненность, радиационный фон, химический состав воды и др. в привязке к географической позиции.

- *Данные с камер наружного наблюдения* позволяют проводить интеллектуальный анализ таких характеристик городской среды, которые невозможно или затруднительно собрать другими средствами (загруженность автомобильных и пешеходных дорог, загрязненность и качество дорожной сети, идентификация отдельных объектов и событий в общем потоке).
- *Данные из открытых источников* (такие как открытые государственные порталы и службы, данные о метеорологических условиях, открытая отчетная информация субъектов бизнеса и др.) позволяют обогатить модели интеллектуального анализа данных.

Собранные данные позволят обеспечить функционирование комплекса цифровых двойников города. Примерами таких цифровых двойников могут служить [13]:

- *цифровой двойник городской инфраструктуры*, представляющий собой интерактивную 3D-модель зданий, сооружений, инженерных коммуникаций и другой городской инфраструктуры;
- *цифровой двойник транспортной сети*, обеспечивающий мониторинг и прогнозирование развития ситуации транспортной доступности, эффективности работы общественного транспорта и др.;
- *цифровой двойник городской экологии*, предоставляющий механизмы мониторинга и прогнозирования экологического состояния городской среды, включая качество почвы, воды, воздуха и др.;
- *цифровой двойник энергетики* и др.

Цифровой двойник города обеспечивает следующие возможности:

- мониторинг текущего состояния городской среды;
- оперативное реагирование на возникновение чрезвычайных ситуаций;
- оценку эффективности проектных решений;
- выявление источников потенциальных рисков;
- прогнозирование развития ситуации с учетом исторических данных.

Цифровой двойник города — это чрезвычайно сложное, комплексное решение, формирование которого может и должно реализовываться поэтапно, путем эволюционного развития и интеграции частных решений, направленных на решение наиболее острых («точечных») проблем. Анализ показывает, что такого рода решения сегодня начинают применяться повсеместно во всех странах мира, не только в передовых мегаполисах, но и в городах большого и среднего размера. Использование таких систем совместно с технологиями «интернета вещей» позволяет решать целый ряд проблем на качественно ином уровне [1]:

- контроль над уровнем загрязнения и анализ регулирующего воздействия путем обследования состояния окружающей среды;
- микроклиматические прогнозы погоды с опорой на городскую сеть датчиков;
- повышение эффективности и снижение расходов за счет вывоза и переработки мусора по необходимости, а не по графику;
- улучшение ситуации на дорогах и экономия топлива за счет умных светофоров и разметки
- рациональное потребление электроэнергии благодаря городскому освещению по необходимости;

- оптимизация снегоуборочных работ благодаря поступающим в реальном времени данным о ситуации на дорогах, погодных условиях и ближайших снегоуборочных машинах;
- умная система полива в парках и общественных местах, учитывающая погодные условия и текущее состояние;
- умные камеры наблюдения для отслеживания преступных деяний и автоматизированная система оповещений AMBER Alert в реальном времени;
- умные парковки, помогающие автоматически подобрать лучшее парковочное место;
- мониторинг износа и состояния мостов, улиц и городской инфраструктуры, направленный на своевременное обслуживание и продление срока службы.

Одним из примеров формирования цифрового двойника может служить пример города Такамацу (Япония, население 420 тыс. человек). В качестве пилотного проекта, власти города внедрили 2 цифровых двойника: цифровой двойник мониторинга и предотвращения чрезвычайных ситуаций (в частности, наводнений) и цифровой двойник туристической привлекательности города [26].

Цифровой двойник для мониторинга и предотвращения чрезвычайных ситуаций основывается на сборе и анализе данных с датчиков уровня воды, расположенных по всей площади города, и обеспечивает мониторинг риска наводнения каждого из районов города в реальном времени. Система также обеспечивает мониторинг состояния убежищ для жителей города путем сбора информации с датчиков влажности и потребления электроэнергии в каждом из убежищ. Мобильное приложение, доступное каждому жителю города, позволяет не только заблаговременно оповестить жителей района, находящихся в зоне риска, но и предоставить всю необходимую информацию на случай экстренной ситуации, включая путь до ближайшего функционирующего убежища.

В плане цифрового двойника туристической привлекательности города был внедрен мониторинг перемещения арендованных велосипедов (как наиболее привлекательного метода перемещения по городу для туристов). Сопоставление гео-трека перемещения арендованных велосипедов с данными о туристе (возраст, пол, гражданство и др.) позволяют сформировать карту наиболее важных туристических локаций города, спланировать и оценить эффективность маркетинговых кампаний по продвижению туристических достопримечательностей (включая поддержку мульти-язычной среды и др.).

2. Классы моделей

ЦД используют различные подходы для моделирования реальных объектов и технологических процессов, включая методы статистического и интеллектуального анализа данных, методы вычислительного моделирования, такие как метод конечных элементов и др. [29]. Каждый из этих методов предъявляет особые требования к необходимым вычислительным ресурсам. Например, методы интеллектуального анализа данных требуют хранилищ большого объема с высокой пропускной способностью для сбора и получения доступа к аналитическим данным, а также высокой масштабируемости вычислительной системы для их обработки; для применения методов машинного обучения требуются узлы с установленными графическими ускорителями, а для моделей, использующей метод конечных элементов, требуются центральные процессоры с высокой производительностью и большие объемы оперативной памяти [9]. В этом разделе рассматриваются различные классы моделей, применяемые для создания цифровых двойников.

Физическая модель обеспечивает компьютерное моделирование физических процессов, протекающих во времени. Построение таких моделей на практике связано с использованием современных пакетов моделирования, например, таких, как ANSYS. Основные физические модели, используемые для построения цифровых двойников:

- модели на основе уравнения Навье—Стокса (моделирование гидро-газодинамических процессов);
- модели на основе метода конечных элементов (моделирование конструкций из твердых материалов).

Пример применения физической модели для построения цифрового двойника можно найти в статье [31]. В данной статье представлены результаты численного моделирования рассеивания в атмосфере загрязнителей воздуха, образующихся при отводе дымовых газов, образующихся при сгорании топлива на ТЭС. Модель использует уравнение Навье—Стокса и реализована с использованием пакета ANSYS CFX-CFD.

Оптимизационная модель обеспечивает поиск оптимумов целевой функции при наличии ограничений с использованием математических методов. Структура оптимизационной модели состоит из целевой функции, области допустимых решений и системы ограничений, определяющих эту область. Главной задачей оптимизационного моделирования является нахождение экстремума функций при ограничениях в форме уравнений и неравенств.

Основные подходы:

- линейное программирование;
- смешанное целочисленное программирование (Mixed Integer Programming);
- нелинейное программирование.

Примером применения оптимизационных моделей является оптимизация программы озеленения улиц города [43]. В данном исследовании предпринимается попытка определить модель планирования, которая определяет местоположение и тип зеленых насаждений на основе их множественных эффектов (например, охлаждение и улучшение связности), и рассчитывает стоимость реализации с использованием мета-эвристических алгоритмов оптимизации. Еще одним примером может служить оптимизация дорожного строительства [10]. В статье решается задача поиска наилучшего плана строительства с учетом предварительных условий и взаимозависимостей пользы от завершения проектов. Задача формулируется как двухуровневая проблема, где объективная функция заключается в минимизации обобщенных затрат, а нижний уровень учитывает выбор маршрута водителями.

Имитационное моделирование представляет собой метод исследования, при котором изучаемая система заменяется компьютерной моделью, имитирующей с достаточной точностью процессы, происходящие в реальной системе, с которой проводятся эксперименты с целью получения информации об этой системе [28]. Яркими представителями имитационных моделей могут служить модели распределения энергоресурсов конечных потребителей [25]. Еще одним ярким примером может служить прототип системы, представленный в статье [14]. Прототип включает в себя трехмерную модель застроенной среды, модель уличной сети с использованием теории и метода космического синтаксиса, моделирование городской мобильности, моделирование ветрового потока и ряд эмпирических количественных и качественных данных с использованием добровольной географической информации (volunteered geographic information). Кроме того, городской цифровой двойник был реализован в платформе визуализации для виртуальной реальности.

2.1. Модели на основе интеллектуального анализа данных

Модели на основе *интеллектуального анализа данных (Data Mining)* применяются для обнаружения в данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия стратегически важных решений в различных сферах человеческой деятельности [19]. В рамках построения цифрового двойника города модель на основе интеллектуального анализа данных позволяет решать следующие основные задачи:

- выявление влияния факторов друг на друга;
- выявление степени влияния факторов на показатель;
- прогнозирование значений факторов и показателя.

Таблица

Показания датчиков

Дата	Метеофакторы			Внутренние факторы			Внешние факторы		...	Показатель
	Сила ветра, м/с	Влажность воздуха, %	t° воздуха, $^{\circ}C$	Интенс-ть город. трафика, 1..10	Выбросы промышл. предпр-й города, тыс. тонн	Чистота улиц, 1..10	Выбросы промышл. предпр-й области, тыс. тонн	Въезд-й грязный транспорт, %		
10 апр	5	40	12	5	1,3	4	4	65		2
11 апр	7	60	14	4	0,6	8	2,1	40		1
12 апр	8	65	13	3	0,5	9	1,2	30		0,3
13 апр	10	80	16	10	1,6	3	3,9	85	...	3
...										

Рассмотрим методы решения указанных задач на примере анализа концентрации мелкодисперсных взвешенных частиц в городском воздухе (далее данный показатель обозначается как n и в качестве единиц измерения используется ПДК — предельно допустимая концентрация). Пусть имеется база данных, содержащая показания, приведенные в таблице.

Выявление влияния факторов друг на друга осуществляется с помощью методов *поиска шаблонов (pattern mining)* [6]. *Шаблон* представляет собой устойчиво выполняющееся правило вида «ЕСЛИ *антецедент*, ТО *консеквент*», где в качестве антецедента и консеквента фигурируют непустые и непересекающиеся между собой множества факторов. Пример двух таких правил представлен ниже:

- ЕСЛИ Сила ветра > 10 м/с И Влажность > 65% ТО Грязный транспорт > 75%;
- ЕСЛИ Дата = Понедельник ТО Выбросы пром. предприятий города > 1.5 тыс. тонн И Выбросы пром. предприятий области > 3 тыс. тонн.

Устойчивость выполнения правила определяется посредством двух основных мер: поддержка и достоверность. *Поддержка правила (rule support)* представляет собой вероятность наличия в базе данных записей, в которых присутствует как антецедент, так и консеквент. *Достоверность правила (rule confidence)* представляет собой условную вероятность наличия в базе данных записей, в которых присутствует антецедент, при условии

присутствия в этих записях консеквента. В качестве шаблонов на практике выбирают правила, поддержка и достоверность которых превышают пороговые значения ($minsup$ и $minconf$ соответственно), наперед заданные экспертом в данной предметной области. Для поиска шаблонов основными алгоритмами являются Apriori [7], FP-Growth [22, Eclat [45], а также их многочисленные модификации, и параллельные и распределенные версии.

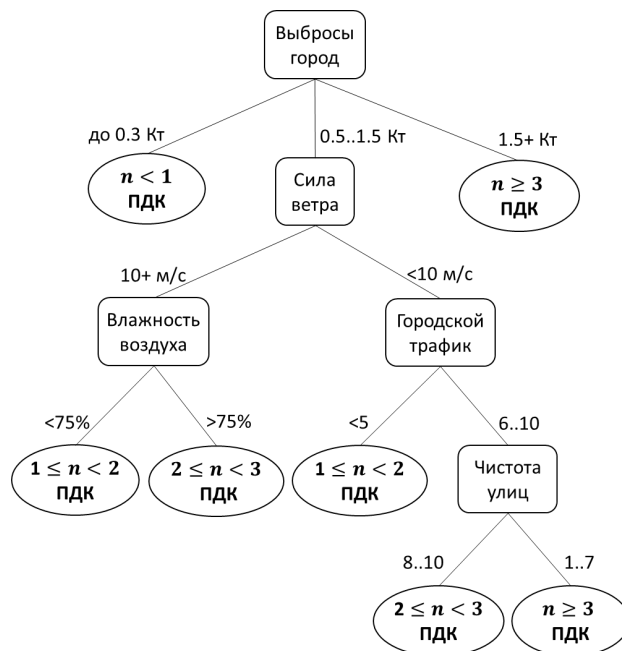


Рис. 2. Пример использования классификационной модели интеллектуального анализа данных

Выявление степени влияния факторов на показатель может быть выполнено как построение иерархии факторов, отражающей степень влияния фактора на показатель: чем выше фактор в иерархии, тем больше его влияние на показатель. Решение данной задачи связано с классификацией данных в базе данных сенсоров на основе деревьев решений [36]. На рис. 2 приведен пример построенного дерева решений, позволяющего классифицировать дни, в которые производились наблюдения, по значению показателя «Концентрация мелкодисперсных взвешенных частиц» со следующими классами значений: «не более нормы», «превышение нормы до одной ПДК», «превышение нормы до двух ПДК», «превышение нормы более двух ПДК».

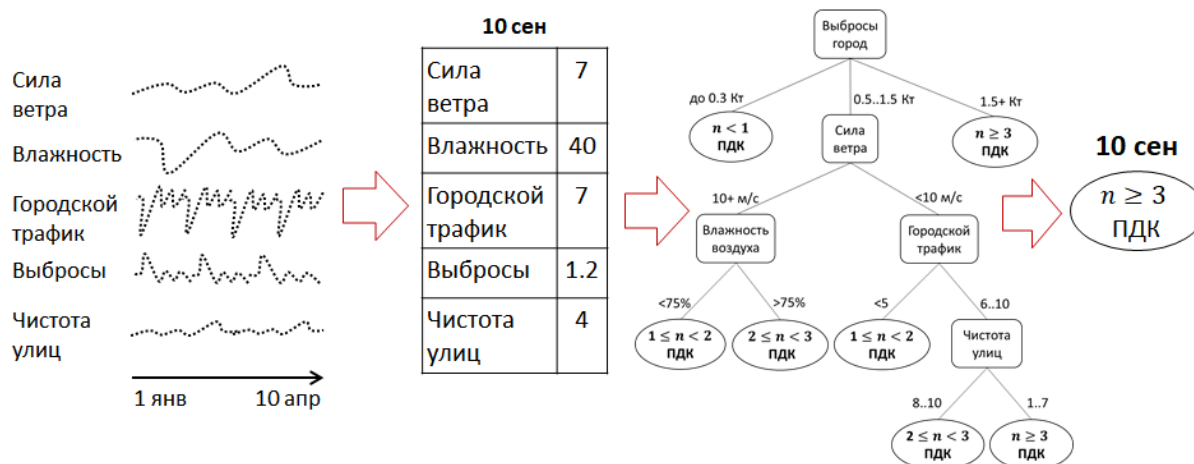


Рис. 3. Модель предсказания предельно допустимой концентрации

Задача *прогнозирования значений факторов и показателя* может быть решена как на базе нейросетевых моделей (см. раздел 2.2), так и в рамках моделей интеллектуального анализа данных. В последнем случае решение задачи предполагает два этапа (см. рис. 3). На первом этапе выполняется прогноз значений факторов на основе регрессионных моделей [15]. На втором этапе, используя полученные прогнозные значения показателей и ранее построенную классификационную модель, вычисляется прогнозное значение показателя.

2.2. Нейросетевые модели

Искусственная нейронная сеть (ИНС) — математическая модель, а также ее программное или аппаратное воплощение, построенные по принципу организации и функционирования биологических нейронных сетей [47]. ИНС представляет собой систему соединенных и взаимодействующих между собой искусственных нейронов. Каждый нейрон подобной сети имеет дело только с сигналами, которые он периодически получает, и сигналами, которые он периодически посылает другим нейронам. В результате обучения нейронная сеть способна выявлять сложные зависимости между входными данными и выходными, а также выполнять обобщение и сможет вернуть верный результат на основании данных, которые отсутствовали в обучающей выборке, а также неполных и/или «зашумленных», частично искаженных данных.

Сверточная нейронная сеть (СНС) — специальная архитектура искусственных нейронных сетей, нацеленная на анализ изображений. Сверточные нейронные сети состоят из нескольких слоев: сверточные слои, субдискретизирующие (subsampling, подвыборка) и слои «обычной» нейронной сети (полносвязные) — персептрона. Первые два слоя (сверточный и субдискретизирующий), чередуясь между собой, формируют входной вектор признаков для персептрона [8]. Сверточный слой включает в себя ядра свертки, которые являются матрицами весов, обрабатывающие предыдущие слои по фрагментам. На данный момент существует множество реализаций сверточных нейронных сетей для решения задач умного города. Приведем некоторые примеры. Detectron2 [42] — программная система Facebook AI Research [16], в которой реализованы современные алгоритмы обнаружения объектов такие, как Faster R-CNN, Mask R-CNN, RetinaNet, DensePose и другие. YOLOv3 [37] — система обнаружения объектов в реальном времени. Благодаря широкому диапазону доступных вариантов можно выбрать версию, наиболее подходящую для приложения. Например, Tiny YOLO [17] — это самый «компактный» вариант, который может работать даже на смартфонах.

Нейросетевые модели применяются для решения задач следующих классов.

- Детекция объектов на изображении. Например, детектирование типа автотранспорта и наличия грязи на нем или мусора, в том числе и нескольких объектов сразу на одном изображении.
- Выявления сложных нелинейных зависимостей между исходными данными и целевыми показателями. Например, построение моделей зависимости концентрации мелкодисперсных частиц от различных факторов, оценка степени влияния факторов.
- Построение рекомендательных систем. Например, систем поддержки принятия решений для управления городским трафиком с целью уменьшения вредных выбросов в атмосферу.

Примером нейросетевой модели для оценки городского трафика на перекрестке является «Интеллектуальная система мониторинга транспортных потоков и дорожно-транспортной инфраструктуры AIMS (Artificial Intelligence Monitoring System)» [39]. Целью проекта являлась разработка и внедрение системы оценки эффективности использования



Рис. 4. Пример работы нейронной сети для распознавания автомобилей

дорожной инфраструктуры, прогнозирования транспортных заторов и суммарных токсичных выбросов от автотранспорта. Использование нейронных сетей глубокого обучения позволяет в режиме реального времени осуществлять сбор, интерпретацию и агрегацию данных по интенсивности и классификации дорожного трафика.

Интеллектуальная система мониторинга транспортных потоков и дорожно-транспортной инфраструктуры позволяет решить сразу ряд сложных задач, таких как сбор, интерпретация и агрегация данных дорожного трафика, выявление недоиспользованных ресурсов в транспортной инфраструктуре, что позволяет сократить капитальные и эксплуатационные затраты. Система позволяет оценить эффективность решений по организации дорожного движения в режиме реального времени (изменения циклов работы светофорных объектов, перенос места нанесения разметки и др.), прогнозировать суммарные выбросы токсичных выхлопных газов от автотранспорта с учетом атмосферных и климатических условий. Также программа предупредит о превышении ПДК в узлах (перекрестках) улично-дорожной сети, что позволит не допустить негативного сценария (увеличив пропускную способность узла, ограничив движения для грузовиков).

Технология с использованием обученных нейросетей, не требует больших затрат на серверное оборудование и видеокамеры. Для мониторинга крупного перекрестка, зачастую, достаточно установки одной камеры уличного видеонаблюдения [39]. Пример работы такой системы приведен на рис. 4.

Еще одним примером использования нейросетевых технологий является задача детектирования мусора на улицах города [44]. Общая структура схемы работы модели показана на рис. 5.

Область переднего плана, совместные тепловые карты и совместные позиции получаются путем вычитания фона и совместной оценки. Затем алгоритм отслеживает пешеходов и обнаруживает объект, который они несут. После обнаружения объекта он отслеживается с помощью трекаera фильтра корреляции, который работает в режиме реального времени. Одновременно, расстояние постепенно моделируется между суставом каждого человека и отслеживаемым объектом. Если обнаружено изменение расстояния между объектом и человеком, алгоритм выдает предупреждение о сбросе мусора. Экспериментальные результаты показали высокую эффективность предложенного метода.

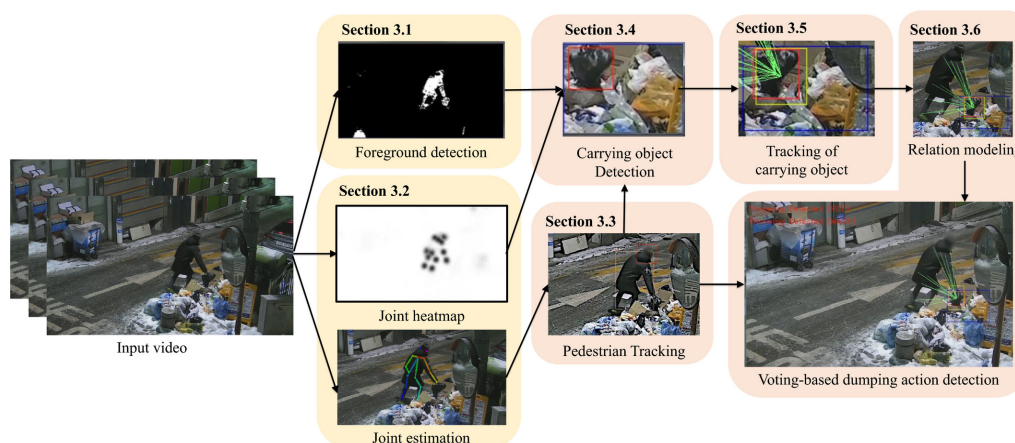


Рис. 5. Пример работы нейронной сети для определения мусора на изображении

3. Информационная инфраструктура цифрового двойника города

Под информационной инфраструктурой цифрового двойника города подразумеваются следующие компоненты:

- датчики, которые собирают информацию;
- сети, которые занимаются передачей данных;
- программные системы, которые занимаются очисткой данных;
- программные комплексы, которые занимаются интеллектуальным анализом данных.

Для того, чтобы ЦД мог обеспечить синхронизацию между состоянием процесса в реальном мире и его виртуальной копией, мы должны обеспечить ему возможность получать, передавать и анализировать поток данных от интеллектуальных конечных устройств Интернета вещей.

3.1. Сенсорные сети

Для сбора информации могут применяться как существующие инфраструктуры датчиков, так и разворачиваться новые сенсорные кластера. Можно выделить следующие типовые виды источников данных:

- видеокамеры;
- датчики давления;
- датчики влажности;
- датчики температуры;
- датчики загрязнения воздуха;
- геолокационные данные транспортных средств;
- информация с проездных карт пассажиров и др.

Выделяют несколько уровней связи, через которые проходят данные, собираемые с сенсорных сетей. Одним из вариантов такой классификации является разбиение каналов связи по зоне охвата на уровни полевых, локальных и глобальных вычислительных сетей [13].

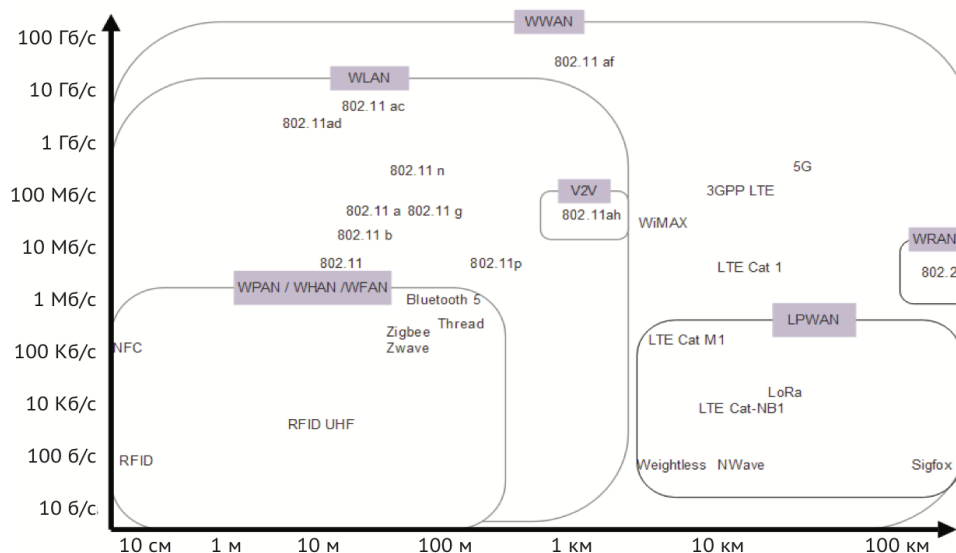


Рис. 6. График сравнения скорости и дальности передачи данных различными каналами связи [13]

Полевой уровень связи обеспечивает передачу данных на расстояния от десятков сантиметров до сотни метров, от места их генерации до первого узла на краю сети, отвечающего за их сбор и дальнейшую ретрансляцию. Для обмена данными на этом уровне применяются низкоскоростные маломощные информационные каналы. Среди коммуникационных протоколов данного уровня можно отметить протоколы для промышленных сетей Modbus и HART, которые ориентированы на работу на базе проводных сетей типа RS-232/RS-485 и «Токовая петля». В последнее десятилетие также значительно расширился набор технологий, обеспечивающих возможности беспроводной передачи данных от сенсорных систем. Среди наиболее известных беспроводных протоколов полевого уровня можно назвать: Bluetooth; WirelessHART; ZigBee (стандарт IEEE 802.15.4); Z-Wave; NFC; RFID и другие (см. рис. 6).

Уровень локальных вычислительных сетей обеспечивает передачу данных в рамках локальной сети, развернутой в непосредственной близости от источников данных (на расстояниях от сотен метров до километра). Сегодня на этом уровне чаще всего используются стандартные сетевые решения, базирующиеся на стеке коммуникационных протоколов TCP/IP. Для организации проводных сетей используются решения на базе семейства технологий Ethernet, а беспроводные сети реализуются на базе семейства протоколов Wi-Fi (IEEE 802.11).

На уровне глобальных вычислительных сетей обеспечивается передача данных в рамках сети Интернет. За функционирование данного уровня отвечают телекоммуникационные операторы, в том числе операторы сетей сотовой связи, спутниковой связи, маломощных глобальных сетей (Low-Power Wide-Area Network, LPWAN, см. рис. 6). Для организации передачи и эффективной обработки данных в глобальных сетях обычно применяются транспортные протоколы интернета для IoT и сетевых устройств, обеспечивающие возможности синхронной либо асинхронной обработки данных, такие как MQTT, CoAP, HTTP [1]. Для проводной передачи данных применяются решения магистральных телекоммуникационных линий, такие как Gigabit Ethernet, EPON и GPON. Для беспроводной передачи данных IoT могут быть организованы маломощные глобальные сети, в том числе на базе протокола LoRa. Отдельно стоит отметить возможности использования сетей сотовой связи для обеспечения прямого соединения устройств IoT с узлами в сети Интернет на основе технологий 2G (GSM), 3G, LTE, 5G.

3.2. Очистка данных

На протяжении жизненного цикла сенсоров данные, выдаваемые ими, могут «загрязняться». Загрязнение поступающих данных проявляется в виде некорректных, аномальных или пропущенных значений. Причиной загрязнения могут быть человеческий фактор, отказ сенсора, сбои в линиях связи, плановое техобслуживание сенсора и др. Загрязненные данные обесценивают их последующее применение в моделях на основе интеллектуального анализа данных и нейронных сетей. Для использования в цифровом двойнике города данные нуждаются в очистке [41]. Очистка данных, поступающих с сенсоров, подразумевает следующие основные операции: поиск аномалий и восстановление пропущенных значений.

Поиск аномалий предполагает нахождение промежутков и соответствующих значений временного ряда, которые существенно не похожи на значения во всех остальных промежутках этого ряда [12]. *Восстановление пропущенных значений* предполагает генерацию синтетических значений временного ряда вместо отсутствующих или ошибочных показаний на основе ретроспективного анализа значений временного ряда данного сенсора и/или значений временных рядов сенсоров, географически/логически близких к данному [11]. Решение указанных задач может быть выполнено как методами интеллектуального анализа данных, так и на основе нейросетевых моделей [27, 35].

При создании цифрового двойника города важной компонентой системы является *модуль очистки данных*, который обеспечивает поиск аномалий и восстановление пропущенных значений в потоках данных, поступающих с сенсоров системы. Работа модуля очистки данных организуется в соответствии со следующими основными принципами. Во-первых, перед поступлением в ЦОД поток данных каждой группы сенсоров должен обрабатываться в соответствии со своими правилами очистки. Во-вторых, очищенный поток данных каждой группы сенсоров должен регулярно контролироваться в соответствии со своими правилами верификации.

Правило очистки определяет область корректных значений для каждой группы сенсоров (например, минимальное и максимальное значения показаний, зависимость показаний сенсора данной группы от показаний сенсоров других групп и др.). *Правило верификации* для данной группы сенсоров определяет периодичность и способ оценки правдоподобия значений, генерируемых синтетически, вместо пропущенных или аномальных. Для оценки правдоподобия используются различные меры (например, нормализованная средняя абсолютная ошибка, средняя квадратическая ошибка и др. [23]) и соответствующие группе сенсоров предельные значения этих мер. Низкая степень правдоподобия синтетических значений, выявленная в результате верификации, является поводом для пересмотра соответствующих методов и алгоритмов, реализованных в модуле очистки данных.

3.3. Туманные вычисления

Для обработки информации, поступающей от множества источников интернета вещей (IoT) целесообразно применение концепции туманных вычислений.

Туманные вычисления (ТВ, англ. fog computing) — это многоуровневая модель-расширение облачных вычислений (англ. cloud computing), облегчающая развертывание распределенных приложений и услуг, учитывающих сетевые задержки, на, так называемых, *туманных узлах* (англ. fog nodes) (физических или виртуальных), располагающихся между умными конечными устройствами и централизованными (облачными) сервисами. *Туманные узлы* являются контекстно-зависимыми и поддерживают единую систему управления данными и организации связи. Они могут быть организованы в кластеры

вертикально (для поддержки изоляции), горизонтально (для поддержки федераций сервисов) или в привязке к сетевой близости от конечных умных устройств. Туманные вычисления минимизируют время сетевого отклика поддерживаемых приложений, а также обеспечивают конечные устройства локальными вычислительными ресурсами и, при необходимости, сетевым подключением к централизованным сервисам [24].

В то время как *облачные вычисления* обеспечивают потенциально безграничные ресурсы для решения задач, требующих существенных вычислительных ресурсов, туманные вычисления обеспечивают:

- предоставление вычислительных ресурсов в непосредственной близости от конечных устройств;
- предобработку данных перед их отправкой в облако;
- решение задач, которые требуют минимального времени отклика.

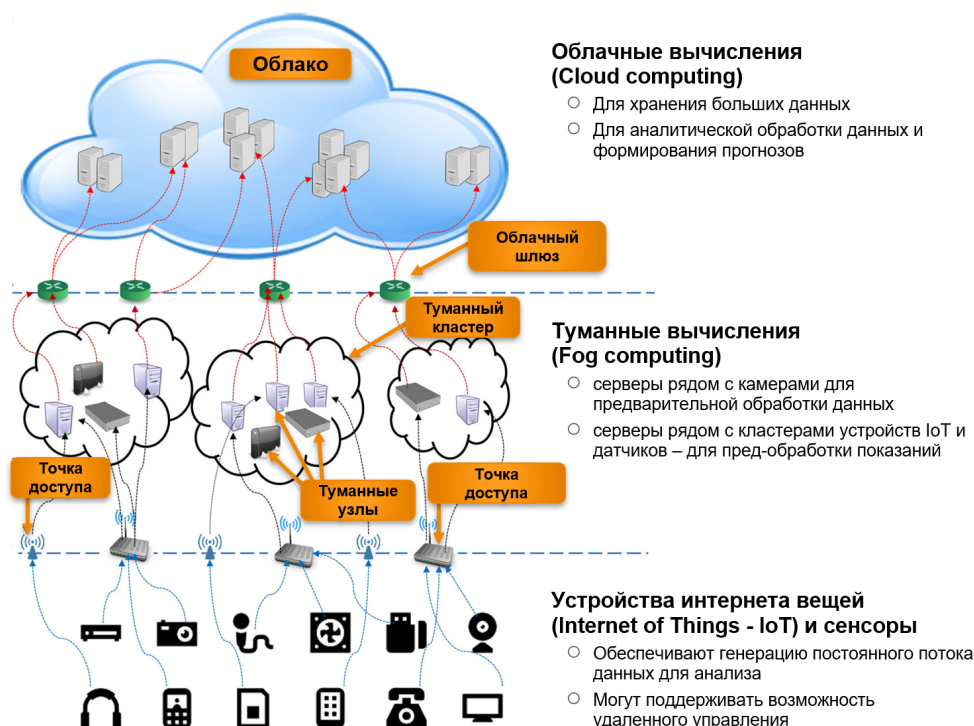


Рис. 7. Модель туманных вычислений

Примером использования туманных вычислений для организации эффективной обработки данных, может служить задача предобработки данных с камер видеонаблюдения. Сегодня в городе могут размещаться сотни камер наружного наблюдения, обеспечивающие покрытие сервисами видеомониторинга большой площади объектов, имеющих особый интерес для управления городским хозяйством. Если для их анализа применяются ресурсы единого центра обработки данных (облака), то задача передачи потока видеоданных по магистральным каналам от такого большого массива камер может быть значительно затруднена. Для решения этой проблемы приходится прибегать к определенным компромиссам, включающим в себя:

- сильное сжатие видеопотока;
- уменьшение разрешения передаваемой видеoinформации;
- ограничение количества передаваемых кадров в секунду и др.

Такие решения приводят к тому, что интеллектуальная модель, применяемая в облачной системе для анализа поступающих данных, может испытывать недостаток качественной

информации для обеспечения эффективной работы. Возможным решением этой проблемы является развертывание сети туманных узлов в непосредственной близости с источниками видеоданных. Эти узлы потребляют и проводят предварительный анализ видеоданных в оригинальном качестве, обеспечивая решение таких задач предобработки видеоданных, как:

- сегментация и выделение объектов интереса;
- удаление фона;
- интеллектуальное сжатие видеопотока.

Это позволяет значительно снизить объем и повысить качество данных, передаваемых в облако, повышая качество работы системы интеллектуального анализа данных [20, 30, 34].

На рис. 7 изображена общая модель передачи и обработки данных от сенсоров и устройств интернета вещей (IoT) при реализации концепции туманных вычислений. На нижнем уровне модели располагаются устройства IoT, обеспечивающие генерацию потоков данных. Это данные различной природы, различного формата и различной скорости генерации. Устройства IoT напрямую связаны с точками доступа, обеспечивающими их соединение с сетью Интернет.

В непосредственной близости к точкам доступа развернута инфраструктура кластеров туманных узлов, обеспечивающих предобработку поступающих потоков данных. Согласно [18], порядка 40% вычислительной нагрузки по обработке данных систем интернета вещей будет обеспечено на туманных узлах. Также, в силу географической близости к узлам IoT, туманные узлы обеспечивают низкую латентность при необходимости двусторонней коммуникации с конечными узлами.

Предобработанные данные передаются для дальнейшей обработки в облако. В отличие от туманных узлов, инфраструктура облачных вычислений подразумевает предоставление по требованию практически безграничных ресурсов для реализации вычислительных задач.

Заключение

В рамках данной статьи были проанализированы существующие технологии построения цифровых двойников городов. Было рассмотрено определение и ключевые компоненты, составляющие цифровой двойник города, примеры решений, которые уже сегодня можно считать первыми шагами к построению полноценного цифрового двойника. Нами были рассмотрены ключевые методы моделирования, реальных объектов и технологических процессов, которые находят свое применение при создании цифровых двойников. Отдельно, на практических примерах, были рассмотрены модели на основе интеллектуального анализа данных, и нейросетевые модели. Также, нами было дано описание ключевых элементов информационной инфраструктуры цифрового двойника города и были рассмотрены ключевые элементы этой инфраструктуры, такие как сенсорные сети, совмещенные с системами очистки сырых данных. Отдельно было отмечено, что для успешного построения такой системы необходимо формирование информационной среды на основе концепции модели туманных вычислений, так как задачи предварительной обработки потоков данных часто требуют существенных вычислительных ресурсов на предобработку данных, расположенных в непосредственной близости от источников.

Исследование выполнено при финансовой поддержке Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022) и Российского фонда фундаментальных исследований (проекты №№ 20-07-00092-а, 20-07-00140-а и 18-07-01224-а).

Литература

1. Ли П. Архитектура интернета вещей. М.: ДМК Пресс, 2019. 454 с.
2. Минстрой России. Индекс IQ городов по итогам 2018 года. URL: <https://minstroyrf.gov.ru/docs/57570/> (дата обращения: 14.09.2020).
3. От концепции до прикладных решений. Города, управляемые данными. 2016. URL: https://www.pwc.ru/ru/government-and-public-sector/assets/ddc_rus.pdf (дата обращения: 14.09.2020).
4. Повх Е. Десять цифровых двойников городов. 2020. URL: <https://realty.rbc.ru/news/5e297b079a79478024d54ff6> (дата обращения: 14.09.2020).
5. Умные города Росатома. URL: <https://rosatom.city/> (дата обращения: 14.09.2020).
6. Aggarwal C.C., Han J. Frequent pattern mining. Springer, 2014. 480 p. DOI: 10.1007/978-3-319-07821-2.
7. Agrawal R., Srikant R. Fast algorithms for mining association rules in large databases // Proceedings of 20th International Conference on Very Large Data Bases, VLDB'94 (Santiago de Chile, Chile, September, 12–15, 1994). 1994. P. 487–499.
8. Albawi S., Mohammed T.A., Al-Zawi S. Understanding of a convolutional neural network // The International Conference on Engineering and Technology 2017 (Antalya, Turkey, August, 21–23, 2017). 2017. P. 1–6. DOI: 10.1109/ICEngTechnol.2017.8308186.
9. Al-Jarrah O.Y., Yoo P.D., Muhaidat S., et al. Efficient machine learning for Big Data: A review // Big Data Research. 2015. Vol. 2, no. 3. P. 87–93. DOI: 10.1016/j.bdr.2015.04.001.
10. Bagloee S.A., Sarvi M., Patriksson M., et al. Optimization for roads' construction: selection, prioritization, and scheduling // Computer-Aided Civil and Infrastructure Engineering. 2018. Vol. 33, no. 10. P. 833–848. DOI: 10.1111/mice.12370.
11. Batista G.E.A.P.A., Monard M.C. An analysis of four missing data treatment methods for supervised learning // Appl. Artif. Intell. 2003. Vol. 17, no. 5-6. P. 519–533. DOI: 10.1080/713827181.
12. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey // ACM Comput. Surv. 2009. Article no. 15. DOI: 10.1145/1541880.1541882.
13. Data-driven smart cities: Big Data, analytics and security. 2018. URL: <https://skelia.com/articles/data-driven-smart-cities-big-data-analytics-and-security/> (дата обращения: 14.09.2020).
14. Dembski F., Wössner U., Letzgus M., et al. Urban digital twins for smart cities and citizens: the case study of Herrenberg, Germany // Sustainability. 2020. Vol. 12, art. 2307. DOI: 10.3390/su12062307.
15. Draper N.R., Smith H. Applied regression analysis. Wiley, 1981. 707 p.
16. Facebook AI Research. URL: <https://ai.facebook.com/> (дата обращения: 20.05.2020)
17. Fang W., Wang L., Ren P. Tinier-YOLO: A real-time object detection method for constrained environments // IEEE Access. 2019. Vol. 8. P. 1935–1944. DOI: 10.1109/ACCESS.2019.2961959.
18. Fernández-Cerero D., Fernández-Montesa A., Ortega F.J., et al. Sphere: simulator of edge infrastructures for the optimization of performance and resources energy consumption // Simulation Modelling Practice and Theory. 2020. Vol. 101. P. 101966. DOI: 10.1016/j.simpat.2019.101966.
19. Frawley W.J., Piatetsky-Shapiro G., Matheus C.J. Knowledge Discovery in databases: an overview // Knowledge Discovery in Databases. 1991. P. 1–30.

20. Gardašević G., Berbakov L., Mastilović A. Cybersecurity of Industrial Internet of Things // *Cyber Security of Industrial Control Systems in the Future Internet Environment*. 2020. P. 47–68. DOI: 10.4018/978-1-7998-2910-2.ch003.
21. Glaessgen E., Stargel D. The digital twin paradigm for future NASA and U.S. air force vehicles // *Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC conference on structures, structural dynamics and materials conference (Honolulu, Hawaii, USA, April, 23–26, 2012)*. 2012. P. 1818. DOI: 10.2514/6.2012-1818.
22. Han J., Pei J., Yin Y. Mining frequent patterns without candidate generation // *ACM SIGMOD Record*. 2000. Vol. 29, no. 2. P. 1–12.
23. Hyndman R.J., Koehler A.B. Another look at measures of forecast accuracy // *International Journal of Forecasting*. 2006. Vol. 22, no. 4. P. 679–688. DOI: 10.1016/j.ijforecast.2006.03.001.
24. Iorga M., Feldman L., Barton R., et al. Fog computing conceptual model. 2018. URL: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-325.pdf> (дата обращения: 14.09.2020).
25. Iqbal M.N., Kütt L. End-user electricity consumption modelling for power quality analysis in residential building // *19th Int. Scientific Conf. on Electric Power Engineering, EPE. IEEE*, 2018. P. 1–6. DOI: 10.1109/EPE.2018.8396030.
26. Kazuhiko I., Atsush Y. Building a common smart city platform utilizing FIWARE (case study of Takamatsu city) // *NEC Tech. J.* 2018. Vol. 13, no. 1. P. 28–31.
27. Kim J., Tae D., Seok J. A survey of missing data imputation using generative adversarial networks // *Proc. of the 2020 Int. Conf. on Artificial Intelligence in Information and Communication, ICAIIC 2020*. P. 454–456. DOI: 10.1109/ICAIIIC48513.2020.9065044.
28. Kin W., Chan V. *Foundations of simulation modeling*. Wiley Encyclopedia of Operations Research and Management Science. Wiley, 2011. P. 6408.
29. Korambath P., Wang J., Kumar A., et al. A smart manufacturing use case: furnace temperature balancing in steam methane reforming process via Kepler workflows // *Procedia Computer Science*. 2016. Vol. 80. P. 680–689. DOI: 10.1016/j.procs.2016.05.357.
30. Kuplyakov D.A., Shalnov E.V., Konushin V.S., et al. A distributed tracking algorithm for counting people in video // *Programming and Computer Software*. 2019. Vol. 45, no. 4. P. 163–170. DOI: 10.1134/S0361768819040042.
31. Mărunțălu O., Lăzăroiu G., Manea E.E., et al. Numerical simulation of the air pollutants dispersion emitted by CHP using ANSYS CFX // *World Academy of Science, Engineering and Technology, International Journal of Environmental, Chemical, Ecological, Geological and Geophysical Engineering*. 2015. Vol. 9. P. 1058–1064. DOI: 10.5281/zenodo.1108252.
32. Mohammadi N., Taylor J.E. Smart city digital twins // *Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (Honolulu, HI, USA, Nov., 27 – Dec., 1, 2017)*. IEEE, 2017. P. 1–5. DOI: 10.1109/SSCI.2017.8285439.
33. Nam T., Pardo T.A. Conceptualizing smart city with dimensions of technology, people, and institutions // *Proceedings of the 12th Annual International Digital Government Research Conference: Digital Government Innovation in Challenging Times (College Park, MD, USA, Jun, 12–15, 2011)*. ACM, 2011. P. 282–291. DOI: 10.1145/2037556.2037602.
34. Nikouei S.Y., Chen Y., Aved A., et al. I-ViSE: interactive video surveillance as an edge service using unsupervised feature queries // *IEEE Internet of Things Journal (Early Access)*. 2020. DOI: 10.1109/JIOT.2020.3016825.
35. Osman M.S., Abu-Mahfouz A.M., Page P.R. A survey on data imputation techniques: water distribution system as a use case // *IEEE Access*. 2018. Vol. 6. P. 63279–63291. DOI: 10.1109/ACCESS.2018.2877269.

36. Quinlan J.R. Induction of decision trees // Machine Learning. 1986. Vol. 1, no. 1. P. 81–106. DOI: 10.1023/A:1022643204877.
37. Redmon J., Farhadi A. You only look once: unified: real-time object detection // IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016 (Las Vegas, NV, USA, June, 27–30, 2016). P. 779–789. DOI: 10.1109/CVPR.2016.91.
38. Ruohomaki T., Airaksinen E., Huuska P., et al. Smart city platform enabling digital twin // 2018 International Conference on Intelligent Systems (Funchal-Madeira, Portugal, September, 25–27, 2018). IEEE, 2018. P. 155–161. DOI: 10.1109/IS.2018.8710517.
39. Shepelev V., Aliukov S., Glushkov A., et al. Identification of distinguishing characteristics of intersections based on statistical analysis and data from video cameras // Journal of Big Data. 2020. Vol. 7, no. 1. P. 1–23. DOI: 10.1186/s40537-020-00324-7.
40. Smart cities readiness: smart cities maturity model and self-assessment tool, Scottish cities alliance. 2014. URL: https://www.scottishcities.org.uk/site/assets/files/1103/smart_cities_readiness_assessment_-_guidance_note.pdf (дата обращения: 14.09.2020)
41. Wang X., Wang C. Time series data cleaning: A survey // IEEE Access. 2020. Vol. 8. P. 1866–1881. DOI: 10.1109/ACCESS.2019.2962152.
42. Wu Y., Kirillov A., Massa F., et al. Detectron2. URL: <https://github.com/facebookresearch/detectron2> (дата обращения: 25.02.2020).
43. Yoon E.J., Kim B., Lee D.K. Multi-objective planning model for urban greening based on optimization algorithms // Urban Forestry & Urban Greening. 2019. Vol. 40. P. 183–194. DOI: 10.1016/j.ufug.2019.01.004.
44. Yun K., Kwon Y., Oh S., et al. Vision-based garbage dumping action detection for real-world surveillance platform // ETRI Journal. 2019. Vol. 41, no. 4. P. 494–505. DOI: 10.4218/etrij.2018-0520.
45. Zaki M.J. Scalable algorithms for association mining // IEEE Transactions on Knowledge and Data Engineering. 2000. Vol. 12, no. 3. P. 372–390.
46. Zanella A., Bui N., Castellani A., et al. Internet of things for smart cities // IEEE Internet of Things Journal. 2014. Vol. 1, no. 1. P. 22–32. DOI: 10.1109/JIOT.2014.2306328.
47. Zupan J. Introduction to artificial neural network methods: What they are and how to use them // Acta Chimica Slovenica. 1994. Vol. 41, no. 3. P. 327–352.

Иванов Сергей Александрович, к.ф.-м.н., доцент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Никольская Ксения Юрьевна, ст. преподаватель, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Радченко Глеб Игоревич, к.ф.-м.н., доцент, директор Высшей школы электроники и компьютерных наук, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация).

Соколинский Леонид Борисович, д.ф.-м.н., профессор, проректор по информатизации, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Цымблер Михаил Леонидович, д.ф.-м.н., доцент, начальник отдела интеллектуального анализа данных и виртуализации Лаборатории суперкомпьютерного моделирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DIGITAL TWIN OF A CITY: CONCEPT OVERVIEW

© 2020 S.A. Ivanov, K.Yu. Nikolskaya, G.I. Radchenko,
L.B. Sokolinsky, M.L. Zymbler

South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)

E-mail: saivanov@susu.ru, nikolskaiaki@susu.ru, gleb.radchenko@susu.ru,

leonid.sokolinsky@susu.ru, mzym@susu.ru

Received: 15.09.2020

The urban economy is a complex multi-vector system. Creating a single digital twin of such a system is now a difficult task to solve. This article presents the concept of a digital twin of a city. The authors propose an evolutionary approach to this problem, according to which digital twins of individual elements of the urban environment are consistently built on a single hardware and software platform. These digital twins are linked in a single cooperative system that allows one digital twin to use data produced by other digital twins. The article gives the definition and architecture of such a system. It describes the classes of models that can be used to create digital twins. Special attention is paid to neural network models and models for data analysis. The information infrastructure of the digital twin of a city, including sensory networks, data cleansing mechanisms and nebulous calculations, is considered.

Keywords: digital twin, smart city, urban management, neural networks, data mining, sensors, fog computing.

FOR CITATION

Ivanov S.A., Nikolskaya K.Yu., Radchenko G.I., Sokolinsky L.B., Zymbler M.L. Digital Twin of a City: Concept Overview. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2020. Vol. 9, no. 4. P. 5–23. (in Russian) DOI: 10.14529/cmse200401.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

ЧИСЛЕННОЕ МОДЕЛИРОВАНИЕ НЕУСТОЙЧИВЫХ РЕЖИМОВ ОРЕГОНАТОРА

© 2020 Л.А. Прокудина, С.У. Турлакова
Южно-Уральский государственный университет
(454080 Челябинск, пр. им. В.И. Ленина, д. 76)
E-mail: prokudinala@susu.ru, turlakovas@su.ru
Поступила в редакцию: 09.01.2020

Представлена система дифференциальных уравнений в частных производных, являющаяся математической моделью химически активной системы (орегонатора) с диффузионным типом связи между компонентами. Актуальность исследования систем с диффузией связана с проблемой происхождения и формирования пространственных структур в химических, биологических, экологических системах. Записаны уравнения для стационарного состояния системы. Осуществлен переход от исходной системы дифференциальных уравнений к системе дифференциальных уравнений в возмущениях. Разработаны вычислительные алгоритмы для расчета параметров модели орегонатора. Проведены численные исследования представленной модели в пакете MATLAB. Рассчитаны стационарные состояния орегонатора для различных значений стехиометрического коэффициента, отвечающие физическому смыслу процесса. Стехиометрический коэффициент является бифуркационным параметром системы и каждому его значению соответствует единственное положительное стационарное решение. Выведено дисперсионное уравнение. Критерием неустойчивости являются положительные значения как скорости роста, так и частоты возмущений в орегонаторе. Осуществлено численное моделирование устойчивости стационарного состояния по отношению к возмущениям. Выявлены два типа неустойчивости в орегонаторе: смена устойчивости и колебательная неустойчивость. Результаты вычислительных экспериментов показали, что диффузия компонентов порождает более неустойчивые моды с волновыми числами, отличными от нуля. Это свидетельствует о дополнительной диффузионной неустойчивости, являющейся механизмом образования пространственных структур.

Ключевые слова: орегонатор, стационарные решения, неустойчивые режимы, диффузионная неустойчивость.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Прокудина Л.А., Турлакова С.У. Численное моделирование неустойчивых режимов орегонатора // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 24–32. DOI: 10.14529/cmse200402.

Введение

Колебательные реакции Белоусова—Жаботинского являются базовыми моделями для исследования процессов самоорганизации, включая образование неоднородных по пространству распределений концентраций реагирующих веществ в химических, биохимических, экологических системах и инженерных приложениях [1–6]. Наиболее известной моделью является орегонатор [8], который представляет собой систему трех обыкновенных дифференциальных уравнений для концентраций компонентов. В данной статье представлена математическая модель орегонатора как система трех дифференциальных уравнений в частных производных с учетом диффузии компонентов.

Механизм реакции модели Белоусова—Жаботинского обоснован учеными Р. Филдом, Е. Кёрсом, Р. Нойесом [9–12] и получил название ФКН-механизма. Разработка хорошо обоснованного механизма реакции Белоусова—Жаботинского обеспечила основу как для теоретических, так и для экспериментальных исследований.

Работы в изучении реакции Белоусова—Жаботинского направлены на уточнение кинетических констант модели орегонатор [5, 6], численное моделирование ее стационарных состояний и колебательных режимов [7, 8, 13, 14].

Целью работы является численное исследование ФКН-модели орегонатора, изучение влияния стехиометрического коэффициента, диффузии компонентов на неустойчивые режимы орегонатора. Для достижения этой цели решены следующие задачи:

- разработан алгоритм расчета параметров модели с учетом диффузии компонентов;
- по результатам экспериментальных данных авторов модели [11] рассчитаны стационарные состояния исследуемой системы;
- выведено дисперсионное уравнение, численно исследованы неустойчивые режимы орегонатора с учетом диффузии компонентов.

Статья организована следующим образом. В разделе 1 представлена математическая модель орегонатора с учетом диффузии компонентов, а также записана система уравнений для стационарного состояния в зависимости от различных значений стехиометрического коэффициента. В разделе 2 осуществлен вывод дисперсионного уравнения, разработаны вычислительные алгоритмы с целью численного исследования развития возмущений в исследуемой системе. В разделе 3 представлены проведенные в пакете MATLAB вычислительные эксперименты по нахождению неустойчивых параметров орегонатора. В заключении отражены результаты вычислительных экспериментов для неустойчивых режимов орегонатора.

1. Математическая модель орегонатора

Математическая модель орегонатора, описывающая изменение во времени концентрации промежуточных продуктов X , Y , Z с учетом диффузии компонентов, имеет вид [8, 13]:

$$\begin{aligned} \frac{\partial X}{\partial t} &= k_1 AY - k_2 XY + k_3 AX - 2k_4 X^2 + D_X \frac{\partial^2 X}{\partial \xi^2}, \\ \frac{\partial Y}{\partial t} &= -k_1 AY - k_2 XY + f k_5 Z + D_Y \frac{\partial^2 Y}{\partial \xi^2}, \\ \frac{\partial Z}{\partial t} &= k_3 AX - k_5 Z + D_Z \frac{\partial^2 Z}{\partial \xi^2}, \end{aligned} \quad (1)$$

где X — концентрация бромистой кислоты $HBrO_2$; Y — концентрация бромидида Br^{-1} ; Z — концентрация церия Ce^{4+} ; A — бромат-ион BrO_3^- ; параметр модели f — стехиометрический коэффициент; k_i — константы скоростей прямых реакций; D_X, D_Y, D_Z — коэффициенты диффузии компонентов.

Однородное стационарное состояние системы (1) определяется следующим образом:

$$\begin{aligned} k_1 AY_0 - k_2 X_0 Y_0 + k_3 AX_0 - 2k_4 X_0^2 &= 0, \\ -k_1 AY_0 - k_2 X_0 Y_0 + f k_5 Z_0 &= 0, \\ k_3 AX_0 - k_5 Z_0 &= 0. \end{aligned} \quad (2)$$

Стационарные состояния системы $X_0 > 0$, $Y_0 > 0$, $Z_0 > 0$ для различных значений стехиометрического коэффициента представлены в табл. 1.

Таблица 1

Стационарные состояния системы

f	$X_0 \cdot 10^{-8}$	$Y_0 \cdot 10^{-17}$	$Z_0 \cdot 10^{-13}$
0,1500	0,1811	0,0068	0,2282
0,2500	0,1969	0,0122	0,2481
0,3500	0,2126	0,0190	0,2679
0,4500	0,2284	0,0271	0,2878
0,5500	0,2441	0,0352	0,3076
0,6500	0,2599	0,0443	0,3274
0,7500	0,2756	0,0535	0,3473
0,8500	0,2914	0,0644	0,3671

Зависимости стационарных решений системы (2) от значений стехиометрического коэффициента представлены на рис. 1. Увеличение значения f приводит к увеличению значений концентраций веществ. Причем для переменных X и Z выражаются линейными зависимостями, а для переменной Y — нелинейными.

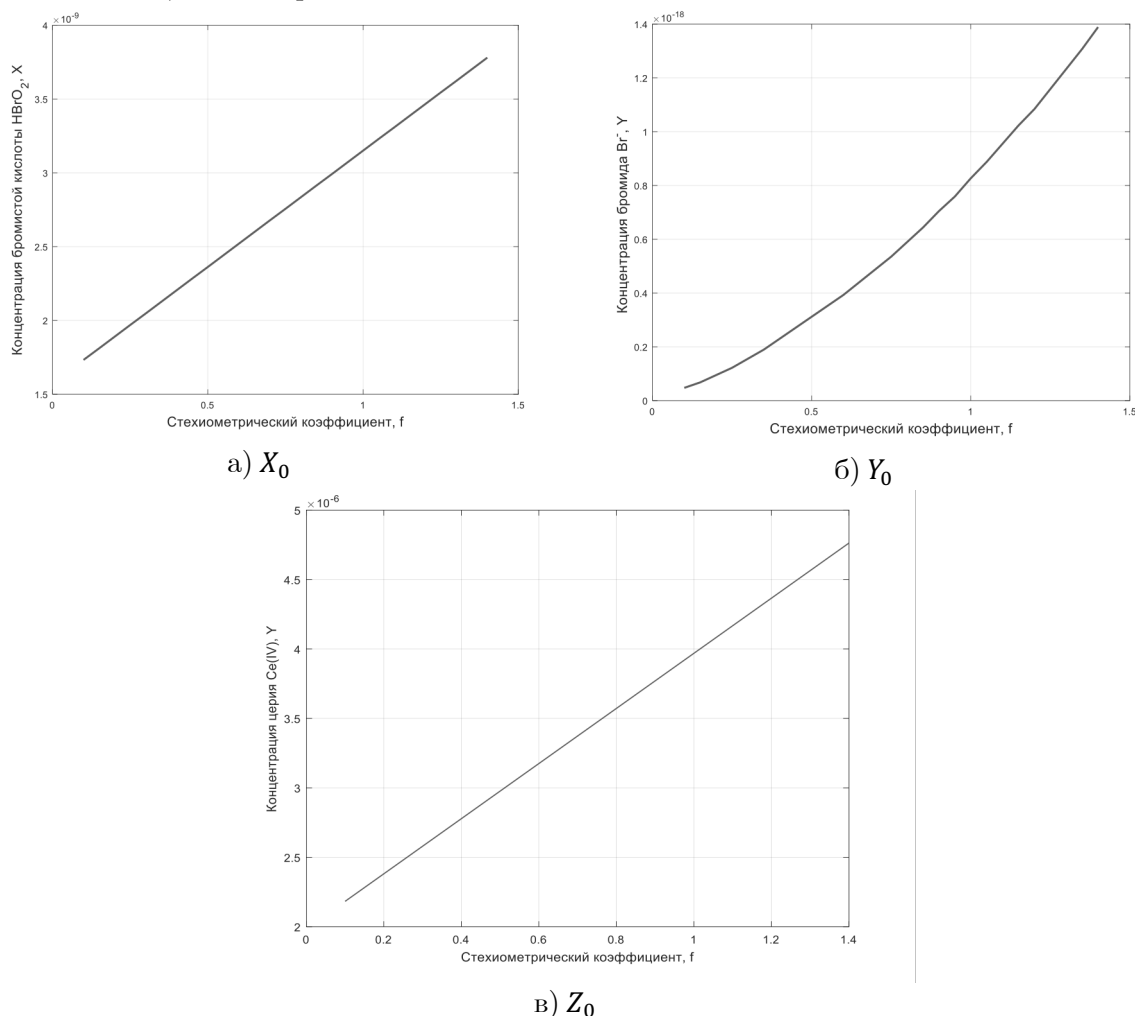


Рис. 1. Зависимость стационарных решений от стехиометрического коэффициента f

2. Моделирование развития возмущений в орегонаторе

Представим в системе (1) $X = X_0 + x$, $Y = Y_0 + y$, $Z = Z_0 + z$, где x , y , z — возмущения, и перейдем к системе дифференциальных уравнений в возмущениях с учетом диффузии компонентов [8, 13]:

$$\begin{aligned} \frac{\partial x}{\partial t} &= a_1x + a_2y + a_3xy + a_4x^2 + D_x \frac{\partial^2 x}{\partial \xi^2}, \\ \frac{\partial y}{\partial t} &= b_1y + b_2x + b_3z + b_4xy + D_y \frac{\partial^2 y}{\partial \xi^2}, \\ \frac{\partial z}{\partial t} &= c_1x + c_2z + D_z \frac{\partial^2 z}{\partial \xi^2}, \end{aligned} \tag{3}$$

где коэффициенты системы имеют вид:

$$\begin{aligned} a_1 &= -k_2Y_0 + k_3A - 4k_4X_0, & a_2 &= k_1A - k_2X_0, & a_3 &= -k_2, & a_4 &= -2k_4, \\ b_1 &= -k_1A - k_2X_0, & b_2 &= -k_2Y_0, & b_3 &= fk_5, & b_4 &= -k_2, \\ c_1 &= k_3A, & c_2 &= -k_5. \end{aligned}$$

Из системы (3) выведем дисперсионное уравнение, исследуя устойчивость стационарного состояния по отношению к возмущениям

$$x(\xi, t) = e^{k\xi - \omega t},$$

$$\begin{vmatrix} a_1 + i\omega - k^2 D_x & a_2 & 0 \\ b_2 & b_1 + i\omega - k^2 D_y & b_3 \\ c_1 & 0 & c_2 + i\omega - k^2 D_z \end{vmatrix} = 0. \quad (4)$$

Раскрыв определитель (4), получим дисперсионное уравнение

$$\omega^3 - ip_1\omega^2 + p_2\omega - ip_3 = 0, \quad (5)$$

где $\omega = \omega_r + i\omega_i$;

ω_r — частота возмущений; ω_i — скорость роста возмущений; k — волновое число;

$p_1 = a_1 + b_1 + c_2 - k^2(D_x + D_y + D_z)$;

$p_2 = (a_1 + b_1 - k^2(D_x + D_y))(c_2 - k^2 D_z) + (a_1 - k^2 D_x)(b_1 - k^2 D_y) - a_2 b_2$;

$p_3 = (a_1 - k^2 D_x)(b_1 - k^2 D_y)(c_2 - k^2 D_z) + a_2 b_3 c_1 - a_2 b_2 (c_2 - k^2 D_z)$.

Критерием неустойчивости исследуемой системы являются условия: $\omega_i > 0$ и хотя бы один корень уравнения (5) должен иметь положительную действительную часть. Разработан алгоритм вычисления корней дисперсионного уравнения (5).

Алгоритм включает в себя следующие шаги:

Шаг 1. Инициализация параметров системы.

Шаг 2. Создание таблицы стационарных решений.

Шаг 3. В цикле по волновому числу k расчет коэффициентов дисперсионного уравнения (5).

Шаг 4. Решение дисперсионного уравнения (5) и выбор корней.

3. Вычислительные эксперименты

В соответствии с алгоритмом в пакете MATLAB написана программа вычисления частоты и скорости роста возмущений для различных значений параметра f и коэффициентов диффузии модели (3). Проведена серия вычислительных экспериментов по выделению неустойчивых режимов в орегонаторе для значений коэффициентов диффузии $D_x = 0,008$; $D_y = 0,016$; $D_z = 0,00001$. Обнаружены различные типы неустойчивости орегонатора. На рис. 2 представлена зависимость частоты от волнового числа для параметра $f = 0,15$ для различных типов неустойчивости.

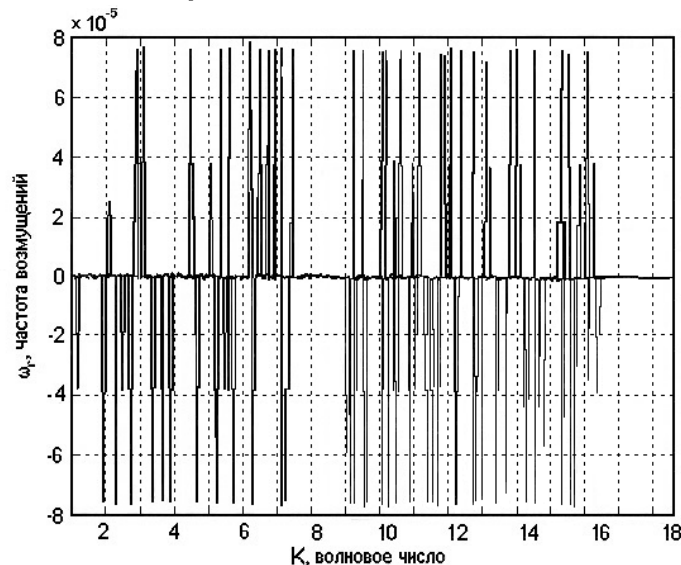


Рис. 2. Зависимость частоты возмущений от волнового числа

Из рис. 2 видно, что частота принимает положительные, отрицательные и нулевые значения для различных волновых чисел. Для $\omega_r < 0$ такой режим лишен физического смысла.

Другие два типа неустойчивости:

- 1) смена устойчивости ($\omega_i > 0, \omega_r = 0$);
- 2) колебательная неустойчивость ($\omega_i > 0, \omega_r > 0$).

Для режима колебательной неустойчивости значения частоты и скорости роста возмущений в орегонаторе представлены в табл. 2.

Таблица 2

Значения частоты и скорости роста возмущений для $f = 0,15$

Волновое число (K)	Частота возмущений (ω_r)	Скорость роста возмущений (ω_i)
1	$5,6702 \cdot 10^{-7}$	10000,3613
2	$2,7805 \cdot 10^{-8}$	10000,4096
3	$7,6371 \cdot 10^{-5}$	10000,4893
4	$8,1528 \cdot 10^{-7}$	10000,6014
5	$1,7974 \cdot 10^{-7}$	10000,7456
6	$1,2412 \cdot 10^{-7}$	10000,9215
7	$7,4676 \cdot 10^{-7}$	10001,1302

В табл. 3 приведены значения волновых чисел, соответствующих режиму смены устойчивости ($D_x = 0,008$; $D_y = 0,016$; $D_z = 0,00001$).

Таблица 3

Режим смены устойчивости для $f = 0,15$

№	K , волновое число
1	1,0018
2	1,0019
3	1,0026
4	1,0027
5	1,0034
6	1,0036
7	1,0042

На рис. 3 представлена графическая зависимость скорости роста возмущений от волнового числа для набора коэффициентов диффузии $D_x = 0,018$; $D_y = 0,006$; $D_z = 0,0001$, соответствующая режиму смены устойчивости. Для различных наборов коэффициентов диффузии величина скорости роста возмущений в орегонаторе меняется незначительно.

Из рис. 3 видно, что неустойчивые режимы орегонатора существуют для всех $k \geq 0$. Минимальное значение скорости роста возмущений соответствует $k = 0$. Режимы с $k > 0$, соответствующие учету диффузии компонентов, имеют большую величину ω_i . Такая неустойчивость называется диффузионной.

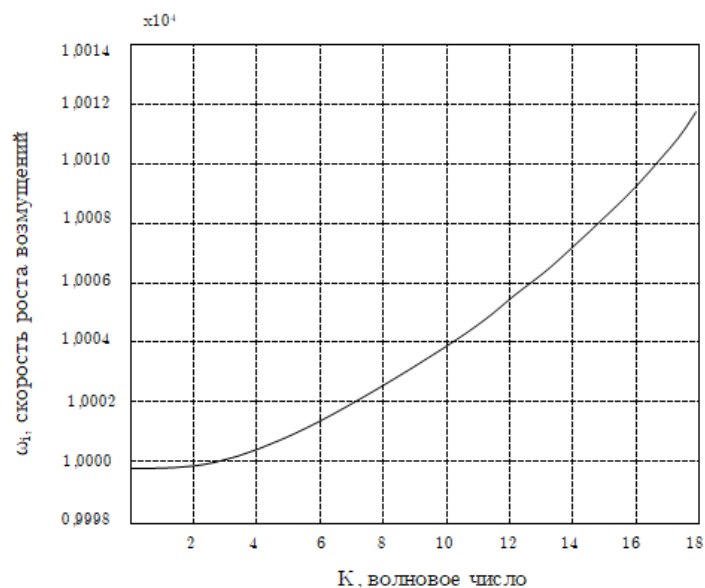


Рис. 3. Зависимость скорости роста возмущений от волнового числа

Заключение

В статье рассмотрена задача численного моделирования неустойчивых режимов орегонатора. Для орегонатора рассчитаны стационарные режимы при различных значениях стехиометрического коэффициента, каждому значению которого соответствуют стационарные состояния, имеющие физический смысл ($X_0 > 0$, $Y_0 > 0$, $Z_0 > 0$).

Вычислительные эксперименты, проведенные в рамках модели (3), показали, что стационарное состояние орегонатора неустойчиво.

Выведено дисперсионное уравнение, разработаны алгоритмы расчета частоты, скорости роста возмущений в орегонаторе для различных значений коэффициентов диффузии и стехиометрического коэффициента.

Проведена серия вычислительных экспериментов и выделены два неустойчивых режима орегонатора: смена устойчивости ($\omega_i > 0$, $\omega_r = 0$); колебательная неустойчивость ($\omega_i > 0$, $\omega_r > 0$).

Диффузия компонентов в исследуемой модели порождает моды с $k \neq 0$, которые более неустойчивы, чем моды с $k = 0$. Это позволяет говорить о дополнительной диффузионной неустойчивости, которая согласно работе Тьюринга [15], является механизмом образования пространственных структур.

Полученные результаты исследований могут использоваться для дальнейшего исследования образования пространственных структур в химических, биологических системах.

Работа выполнялась при финансовой поддержке Министерства образования и науки Российской Федерации прикладных научных исследований в рамках базовой части Государственного задания «Разработка, исследование и реализация алгоритмов обработки данных динамических измерений пространственно-распределенных объектов», техническое задание 8.9692.2017/8.9 от 17.02.2017.

Литература

1. Вольтер Б.В., Сальников И.Е. Устойчивость режимов работы химических реактивов. М.: Химия, 1981. 198 с.
2. Жаботинский А.М. Концентрационные колебания. М.: Наука, 1974. 179 с.

3. Жаботинский А.М. Периодические реакции окисления в жидкой фазе // Доклады Академии наук СССР. 1964. Т. 157. С. 392–395.
4. Жаботинский А.М. Периодический процесс окисления малоновой кислоты в растворе (исследование кинетики реакции Белоусова) // Биофизика. 1964. Т. 9. С. 306–311.
5. Икрамов Р.Д., Мустафина С.А. Численное исследование динамики расширенной модели реакции Белоусова—Жаботинского // Международный научно-исследовательский журнал. 2016. № 9(51). С. 124–128. DOI: 10.18454/IRJ.2016.51.161.
6. Икрамов Р.Д., Мустафина С.А. Алгоритм поиска констант скоростей колебательной реакции на примере реакции Белоусова—Жаботинского // Башкирский химический журнал. 2015. Т. 1, № 22. С. 87–91.
7. Прокудина Л.А. Моделирование неустойчивости орегонатора с диффузией // Нелинейный мир. 2014. Т. 12, № 1. С. 42–47.
8. Прокудина Л.А., Турлакова С.У. Математическое моделирование стационарного состояния и колебательных режимов орегонатора // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. 2018. Т. 7, № 1. С. 5–15. DOI: 10.14529/cmse180101.
9. Field R.J., Koros E., Noyes R. Oscillations in Chemical Systems. II. Thorough Analysis of Temporal Oscillation in the Bromate–Cerium–Malonic Acid System // Journal of the American Chemical Society. 1972. Vol. 94, no. 25. P. 8649–8664. DOI: 10.1021/ja00780a001.
10. Field R.J., Noyes R.M. Oscillations in Chemical Systems. IV. Limit Cycle Behavior in a Model of a Real Chemical Reaction // Journal of Chemical Physics. 1974. Vol. 60, no. 5. P. 1877–1884. DOI: 10.1063/1.1681288.
11. Edelson D., Field R.J., Noyes R.M. Mechanistic Details of the Belousov–Zhabotinskii Reaction // International Journal of Chemical Kinetics. 1975. Vol. 7. P. 417–432. DOI: 10.1021/j100381a039.
12. Koros E., Orban M. Uncatalysed oscillatory chemical reactions // Nature. 1978. Vol. 273. P. 371–372. DOI: 10.1038/273371b0.
13. Prokudina L.A., Kholpanov L.P. Nonlinear Development of Perturbations of Diffusion–Complicated Autocatalytic Reaction // Theoretical Foundations of Chemical Engineering. 2004. Vol. 38(6). P. 636–643. DOI: 10.1007/s11236-005-0037-0.
14. Prokudina L.A., Turlakova S.U. Mathematical Modeling of Oscillatory Regimes in Oregonator // II International Ural Conference on Measurements, UralCon 2017 (Chelyabinsk, Russia, 2017). 2017. P. 113–117.
15. Turing A.M. The Chemical Basis for Morphogenesis // Phil. Trans. Roy. Soc. London. 1952. B 237. P. 37–72.

Прокудина Людмила Александровна, д.ф.-м.н., профессор, кафедра вычислительной математики и высокопроизводительных вычислений, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Турлакова Светлана Ульмджиевна, к.ф.-м.н., доцент, кафедра вычислительной математики и высокопроизводительных вычислений, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

NUMERICAL SIMULATION OF UNSTABLE OREGONATOR REGIMES

© 2020 L.A. Prokudina, S.U. Turlakova

South Ural State University

(pr. Lenina 76, Chelyabinsk, 454080 Russia)

E-mail: prokudinala@susu.ru, turlakovasusu@susu.ru

Received: 09.01.2020

A system of partial differential equations is presented, which is a mathematical model of a chemically active system (oregonator) with a diffusive type of coupling between components. The relevance of the study of systems with diffusion is related to the problem of the origin and forming of spatial structures in chemical, biological, and ecological systems. Equations for the stationary state of the system are written. The transition from the original system of differential equations to the system of differential equations in perturbations is performed. Computational algorithms have been developed for calculating the parameters of the oregonator model. Numerical studies of the presented model are carried out in the MATLAB package. Stationary States of the oregonator are calculated for different values of the stoichiometric coefficient that correspond to the physical meaning of the process. The stoichiometric coefficient is a bifurcation parameter of the system, and each of its values corresponds to a single positive stationary solution. The dispersion equation is derived. The instability criterion is the positive values of both the growth rate and the frequency of disturbances in the oregonator. Numerical modeling of the stability of the stationary state with respect to perturbations is carried out. Two types of instability in the oregonator are identified: change of stability and oscillatory instability. The results of computational experiments have shown that the diffusion of components generates more unstable modes with wave numbers other than zero. This indicates additional diffusive instability, which is a mechanism for the formation of spatial structures.

Keywords: oregonator, stationary solutions, unstable modes, diffusion instability.

FOR CITATION

Prokudina L.A., Turlakova S.U. Numerical Simulation of Unstable Oregonator Regimes. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2019. Vol. 9, no. 4. P. 24–32. (in Russian) DOI: 10.14529/cmse200402.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Voltaire B.V., Salnikov I.E. Stability of Operating Modes of Chemical Reagents. M.: Chemistry, 1981. 198 p.
2. Zhabotinsky A.M. Concentration Auto-Oscillations. M.: Nauka, 1974. 179 p. (in Russian)
3. Zhabotinskiy A.M. Periodic Liquid Phase Reactions. Proc. Ac. Sci. USSR, 1964. Vol. 157. P. 392–395. (in Russian)
4. Zhabotinskiy A.M. Periodic Process of the Oxidation of Malonic Acid in Solution (Study of Kinetics of Belousov's Reaction). Biofizika. 1964. Vol. 9. P. 306–311. (in Russian)
5. Ikramov R.D., Mustafina S.A. A Numerical Study of the Dynamics of an Extended Reaction Model of Belousov–Zhabotinsky. International Research Journal. 2016. no. 9(51). P. 124–128. (in Russian) DOI: 10.18454/IRJ.2016.51.161.
6. Ikramov R.D., Mustafina S.A. Search Algorithm for the Rate Constants of an Vibrational Reaction Using the Example of the Belousov–Zhabotinsky Reaction. Bashkir Chemical Journal. 2015. Vol. 1, no. 22. P. 87–91. (in Russian)

7. Prokudina L.A. Mathematical Modeling of Oregonator with a Diffusion Type Communication. *Journal of Computational and Engineering Mathematics*. 2016. Vol. 3, no. 2. P. 48–56.
8. Prokudina L.A., Turlakova S.U. Mathematical Modelling of Stationary State and Oscillatory Regimes in Oregonator. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2018. Vol. 7, no. 1. P. 5–15. DOI: 10.14529/cmse180101.
9. Field R.J., Koros E., Noyes R. Oscillations in Chemical Systems. II. Thorough Analysis of Temporal Oscillation in the Bromate–Cerium–Malonic Acid System. *Journal of the American Chemical Society*. 1972. Vol. 94, no. 25. P. 8649–8664. DOI: 10.1021/ja00780a001.
10. Field R.J., Noyes R.M. Oscillations in Chemical Systems. IV. Limit Cycle Behavior in a Model of a Real Chemical Reaction. *Journal of Chemical Physics*. 1974. Vol. 60, no. 5. P. 1877–1884. DOI: 10.1063/1.1681288.
11. Edelson D., Field R.J., Noyes R.M. Mechanistic Details of the Belousov–Zhabotinskii Reaction. *International Journal of Chemical Kinetics*. 1975. Vol. 7. P. 417–432. DOI: 10.1021/j100381a039.
12. Koros E., Orban M. Uncatalysed oscillatory chemical reactions. *Nature*. 1978. Vol. 273. P. 371–372. DOI: 10.1038/273371b0.
13. Prokudina L.A., Kholpanov L.P. Nonlinear Development of Perturbations of Diffusion–Complicated Autocatalytic Reaction. *Theoretical Foundations of Chemical Engineering*. 2004. Vol. 38(6). P. 636–643. DOI: 10.1007/s11236-005-0037-0.
14. Prokudina L.A., Turlakova S.U. Mathematical Modeling of Oscillatory Regimes in Oregonator. II International Ural Conference on Measurements, UralCon 2017 (Chelyabinsk, Russia, 2017). 2017. P. 113–117.
15. Turing A.M. The Chemical Basis for Morphogenesis. *Phil. Trans. Roy. Soc. London*. 1952. B 237. P. 37–72.

О ЗАДАЧЕ ПО ВОССТАНОВЛЕНИЮ КОЭФФИЦИЕНТОВ-ФУНКЦИЙ ВРЕМЕНИ ИСТОЧНИКОВ СПЕЦИАЛЬНОГО ВИДА В ПАРАБОЛИЧЕСКОМ УРАВНЕНИИ

© 2020 А.Б. Рагимов

Институт систем управления НАН Азербайджана

(AZ1141 Азербайджан, Баку, ул. Б. Вахабзаде, д. 9)

E-mail: anar_r@yahoo.com

Поступила в редакцию: 05.03.2020

Исследуется обратная задача по идентификации коэффициентов, зависящих только от времени, при источнике специального вида в линейных параболических уравнениях с точечными условиями переопределения. К исследуемым обратным задачам, в частности, приводятся краевые задачи с нелокальными (интегральными) краевыми условиями. Специфика рассматриваемой в данной работе обратной задачи заключается в том, что восстанавливаемые коэффициенты находятся при свободном члене и они зависят только от временной переменной. Предлагается методика численного решения задачи с применением метода прямых, основанная на использовании специального вида представления решения. Методом прямых задача приводится к параметрически обратной задаче относительно обыкновенной системы дифференциальных уравнений. Для ее решения предложено представление этого решения в специальном виде. Построены вспомогательные краевые задачи, которые определяют решение исходной задачи. Наиболее существенным в данной работе является то, что предлагаемый подход к численному решению исследуемой обратной задачи по идентификации коэффициентов не требует (в отличие от ранее известных методов) построения каких-либо итерационных процедур. Приводятся результаты численных экспериментов в виде таблиц и графиков, полученных при решении тестовой задачи, и их анализ.

Ключевые слова: обратная задача, нелокальные условия, метод прямых, параболическое уравнение, параметрическая идентификация.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Рагимов А.Б. О задаче по восстановлению коэффициентов-функций времени источников специального вида в параболическом уравнении // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 33–47. DOI: 10.14529/cmse200403.

Введение

Обратным задачам по восстановлению правой части параболического уравнения посвящены работы многих авторов [5, 7, 11, 14, 17–23, 25, 26]. Наиболее общие случаи этих задач исследовались в монографиях А.И. Прилепко и др. [25] и М.И. Иванчова [22]. В [5] была изучена разрешимость обратных задач по нахождению неизвестного внешнего воздействия (правой части) специального вида. Исследования условий существования, единственности решения обратных задач для параболических уравнений проведены в работах [7, 11, 25].

В данной работе исследуется численное решение обратной задачи относительно параболического уравнения по восстановлению источника специального вида, в которой идентифицируемые коэффициенты зависят лишь от временной переменной. Для численного решения такой задачи можно было бы использовать различные подходы. Один из них заключается в сведении ее к задаче оптимального управления, численное решение которой требует использования итерационных методов оптимизации [18–20, 23]. Другой подход заключается в использовании фундаментального решения задачи и при-

ведении задачи к интегральному уравнению. В случае, когда участвующие в задаче функции, определяющие коэффициенты уравнения, не являются постоянными, использование такого подхода сталкивается с существенными трудностями [14, 17, 21, 26].

К исследуемым обратным задачам, в частности, приводит рассмотренный в работе важный класс краевых задач с нелокальными условиями [2–4, 6, 8–10, 16, 21]. Задание начальных и краевых условий в нелокальном (интегральном) виде обусловлено практической невозможностью производить замеры состояния объекта (процесса) в отдельных его точках или мгновенно во времени. Замеряемая информация, как правило, отражает состояние процесса в какой-либо окрестности точки замера или на интервале времени измерения, а иногда она определяет лишь среднее значение состояния в целом по объекту и/или по всему интервалу времени его функционирования. Подобные задачи возникают при изучении явлений, происходящих в плазме [10], процессов распространения тепла [4, 16, 21] и влагопереноса в капиллярно-пористых средах [3], математическом моделировании технологического процесса внешнего геттерирования при очищении кремниевых плат от примесей [8], в задачах математической биологии [6] и демографии [2].

Специфика рассматриваемой в данной работе обратной задачи заключается, во-первых, в том, что восстанавливаемые коэффициенты находятся при свободном члене, во-вторых, они зависят только от временной переменной. Эта специфика позволяет с использованием метода прямых свести решение исходной задачи к решению специально построенной краевой задачи относительно системы обыкновенных дифференциальных уравнений, а ее решение используется для формирования системы алгебраических уравнений [1, 15].

Статья организована следующим образом. В разделе 1 приводится постановка коэффициентно-обратной задачи относительно параболического уравнения. Раздел 2 посвящен описанию подхода к численному решению поставленной задачи, основанного на использовании метода прямых. В разделе 3 приводятся результаты численных экспериментов, полученные при решении тестовой задачи, и их анализ. В заключении резюмируются полученные результаты и намечаются направления дальнейших исследований.

1. Постановка задачи

Рассмотрим коэффициентно-обратную задачу относительно параболического уравнения [5, 7, 11, 12]:

$$\frac{\partial v(x, t)}{\partial t} = a(x, t) \frac{\partial^2 v(x, t)}{\partial x^2} + a_1(x, t) \frac{\partial v(x, t)}{\partial x} + a_2(x, t) v(x, t) + f(x, t) + F(x, t), \quad (1)$$

$$(x, t) \in \Omega = \{(x, t): 0 < x < l, 0 < t \leq T\},$$

где

$$F(x, t) = \sum_{s=1}^L C_s(x, t) B_s(t), \quad (2)$$

при следующих начально-краевых условиях и условиях переопределения:

$$v(x, 0) = \phi_0(x), \quad x \in [0, l], \quad (3)$$

$$v(0, t) = \psi_0(t), \quad v(l, t) = \psi_1(t), \quad t \in [0, T], \quad (4)$$

$$v(\bar{x}_s, t) = \psi_{2s}(t), \quad \bar{x}_s \in (0, l), \quad t \in [0, T], \quad s = 1, \dots, L. \quad (5)$$

Здесь заданными являются: $L > 0$ — целое число, определяющее количество идентифицируемых источников и условий переопределения; точки $\bar{x}_s \in (0, l), s = 1, \dots, L$; непрерывные по x и t функции $a(x, t) \geq \mu > 0$, $a_1(x, t)$, $a_2(x, t)$, $f(x, t)$, $\phi_0(x)$,

$\psi_0(t), \psi_1(t), \psi_{2s}(t), s = 1, \dots, L, \mu = \text{const} > 0; C_s(x, t), s = 1, \dots, L$ — линейно-независимые функции; $a_2(x, t) \leq 0, \phi_0(x) \in C^2([0, l]), \psi_{2s}(t) \in C^1([0, T]), |C_s(\bar{x}_k, t)| \geq \delta > 0, s, k = 1, \dots, L, t \in [0, T], \delta = \text{const} > 0;$ Заданные точки $\bar{x}_s, s = 1, \dots, L, \bar{x}_i \neq \bar{x}_j$ при $i \neq j, i, j = 1, \dots, L,$ являются точками наблюдения. Функции $\phi_0(x), \psi_0(t), \psi_1(t), \psi_{2s}(t)$ удовлетворяют условиям согласования:

$$\phi_0(0) = \psi_0(0), \phi_0(l) = \psi_1(0), \psi_{2s}(0) = \phi_0(\bar{x}_s), s = 1, \dots, L.$$

Задача (1)–(5) заключается в определении неизвестной непрерывной L -мерной вектор-функции $B(t) = (B_1(t), \dots, B_L(t))^T$ и соответствующего решения краевой задачи $v(x, t) \in C^{2,1}(\Omega) \cap C^{1,0}(\bar{\Omega}),$ удовлетворяющих условиям (1)–(5).

Отметим, что при сделанных выше предположениях обратная задача (1)–(5) имеет решение и оно единственно [7, 11, 12, 25].

Замечание. Известно, что число условий переопределения должно быть не меньше числа идентифицируемых функций, т.е. может превышать их. Этот случай наиболее часто встречается на практике.

К исследуемым обратным задачам, в частности, приводит важный класс краевых задач с нелокальными условиями [2–4, 6, 8–10, 16, 21]. Например, к частному случаю задачи (1)–(5) приводится следующая краевая задача с нелокальным (интегральным) краевым условием:

$$\frac{\partial u(x, t)}{\partial t} = a(t) \frac{\partial^2 u(x, t)}{\partial x^2} + a_1(t) \frac{\partial u(x, t)}{\partial x} + a_2(t)u(x, t) + \tilde{f}(x, t), (x, t) \in \Omega, \quad (6)$$

$$u(0, t) = \psi_2(t), \int_0^l e^{k\xi} u(\xi, t) d\xi = \psi_1(t), t \in [0, T], \quad (7)$$

$$u(x, 0) = \tilde{\phi}_0(x), x \in [0, l], \quad (8)$$

где постоянные $k,$ функции $\tilde{f}(x, t), \psi_0(t), \psi_1(t), \tilde{\phi}_0(x)$ — заданы. Задачи вида (6)–(8) исследовались, например, в работах [9, 21].

Для сведения задачи (6)–(8) к задаче (1)–(5) введем функцию

$$v(x, t) = \int_0^x e^{k\xi} u(\xi, t) d\xi. \quad (9)$$

Дифференцируя (9) по $x,$ получаем

$$u(x, t) = e^{-kx} \frac{\partial v(x, t)}{\partial x}. \quad (10)$$

Дифференцируя (10) один раз по t и дважды по x и подставляя это выражение в (6), после несложных преобразований имеем:

$$\begin{aligned} \frac{\partial^2 v(x, t)}{\partial t \partial x} &= a(t) \frac{\partial^3 v(x, t)}{\partial x^3} + (a_1(t) - 2ka(t)) \frac{\partial^2 v(x, t)}{\partial x^2} + \\ &+ (k^2 a(t) - ka_1(t) + a_2(t)) \frac{\partial v(x, t)}{\partial x} + e^{kx} \tilde{f}(x, t). \end{aligned}$$

Интегрируя обе части этого уравнения по $x,$ получаем уравнение с точностью до произвольной функции $B(t):$

$$\begin{aligned} \frac{\partial v(x, t)}{\partial t} &= a(t) \frac{\partial^2 v(x, t)}{\partial x^2} + (a_1(t) - 2ka(t)) \frac{\partial v(x, t)}{\partial x} + \\ &+ (k^2 a(t) - ka_1(t) + a_2(t))v(x, t) + f(x, t) + B(t). \end{aligned} \quad (11)$$

При этом использовано обозначение $f(x, t) = \int_0^x e^{k\xi} \tilde{f}(\xi, t) d\xi$.

Из (7)–(10) имеем следующие начальные и краевые условия:

$$v(0, t) = 0, \quad v(l, t) = \psi_1(t), \quad t \in [0, T]; \quad (12)$$

$$v(x, 0) = \phi_0(x), \quad x \in [0, l], \quad (13)$$

$$\frac{\partial v(0, t)}{\partial x} = \psi_2(t), \quad t \in [0, T], \quad (14)$$

где

$$\phi_0(x) = \int_0^x e^{k\xi} \tilde{\phi}_0(\xi) d\xi.$$

Ясно, что коэффициентно-обратная задача (11)–(14) является частным случаем задачи (1)–(5).

2. Решение задачи

Ниже предлагается подход к численному решению задачи (1)–(5), основанный на использовании метода прямых. Задача (1)–(5) сводится к системе обыкновенных дифференциальных уравнений с неизвестными параметрами.

В области Ω проведем прямые:

$$t_j = jh_t, \quad j = 0, 1, \dots, N, \quad h_t = T/N.$$

На этих прямых определим функции:

$$v_j(x) = v(x, t_j), \quad x \in [0, l], \quad j = 0, 1, \dots, N,$$

для которых, на основании (3)–(5), имеют место равенства:

$$v_0(x) = \phi_0(x), \quad x \in [0, l], \quad (15)$$

$$v_j(0) = \psi_0(t_j) = \psi_{0j}, \quad j = 0, \dots, N, \quad (16)$$

$$v_j(l) = \psi_1(t_j) = \psi_{1j}, \quad j = 0, \dots, N, \quad (17)$$

$$v_j(\bar{x}_s) = \psi_{2s}(t_j) = \psi_{2sj}, \quad \bar{x}_s \in (0, l), \quad s = 1, \dots, L, \quad j = 0, \dots, N. \quad (18)$$

На прямых $t = t_j$ аппроксимируем производные $\partial v(x, t)/\partial t$ с использованием разностной схемы:

$$\left. \frac{\partial v(x, t)}{\partial t} \right|_{t=t_j} = \frac{v_j(x) - v_{j-1}(x)}{h_t} + O(h_t), \quad j = 1, \dots, N. \quad (19)$$

Используя (19) в (1), получим N дифференциальных уравнений:

$$v''_j(x) + \tilde{a}_{1j}(x)v'_j(x) + \tilde{a}_{2j}(x)v_j(x) + \tilde{f}_j(x) + \sum_{s=1}^L \tilde{C}_{sj}(x)B_{sj} = 0, \quad (20)$$

$$j = 1, \dots, N, \quad x \in (0, l),$$

где использованы обозначения:

$$B_{sj} = B_s(t_j), \quad \tilde{f}_j(x) = \frac{v_{j-1}(x) + h_t f(x, t_j)}{a(x, t_j)h_t}, \quad \tilde{C}_{sj}(x) = \frac{C_s(x, t_j)}{a(x, t_j)},$$

$$\tilde{a}_{1j}(x) = \frac{a_1(x, t_j)}{a(x, t_j)}, \quad \tilde{a}_{2j}(x) = \frac{a_2(x, t_j)h_t - 1}{a(x, t_j)h_t}.$$

Сходимость при $h_t \rightarrow 0$ и погрешность метода прямых при аппроксимации производных по t в уравнении (1) (в данном случае при аппроксимации задачи (1)–(5) зада-

чей (20), (16)–(18) с погрешностью, оцениваемой как $O(h_t)$ исследована в работе [24]. Следовательно, при известных B_{sj} решение краевой задачи относительно системы (22) при $h_t \rightarrow 0$ сходится к решению исходной краевой задачи (1)–(4). Из существования и единственности решения исходной обратной задачи (1)–(5) можно показать существование и единственность решения обратной задачи (20), (16)–(18). Действительно, в случае отсутствия решения обратной задачи (20), (16)–(18) или ее неединственности аналогичные свойства имеет и исходная задача.

Уравнения системы (20) для каждого j решаются независимо и последовательно, начиная от $j = 1$ до N , а следовательно и компоненты вектора $B_s = (B_{s1}, \dots, B_{sN})^T$ определяются последовательно.

Теорема 1. Пусть функции $\alpha_j(x)$, $\beta_{sj}(x)$, $s = 1, \dots, L$ при $x \in [0, l]$ являются решением следующих краевых задач:

$$\alpha''_j(x) + \tilde{a}_{1j}(x)\alpha'_j(x) + \tilde{a}_{2j}(x)\alpha_j(x) + \tilde{f}_j(x) = 0, \quad (21)$$

$$\alpha_j(0) = \psi_{0j}, \quad \alpha_j(l) = \psi_{1j}, \quad (22)$$

$$\beta''_{sj}(x) + \tilde{a}_{1j}(x)\beta'_{sj}(x) + \tilde{a}_{2j}(x)\beta_{sj}(x) + \tilde{c}_{sj}(x) = 0, \quad (23)$$

$$\beta_{sj}(0) = 0, \quad \beta_{sj}(l) = 0. \quad (24)$$

Тогда для произвольных значений параметра B_{sj} функции

$$v_j(x) = \alpha_j(x) + \sum_{s=1}^L \beta_{sj}(x)B_{sj}, \quad x \in [0, l], \quad (25)$$

удовлетворяют системе дифференциальных уравнений (20) и условиям (16), (17).

Доказательство. С учетом первых условий в (22) и (24), для произвольных значений параметра B_{sj} из (25) получим, что

$$v_j(0) = \alpha_j(0) + \sum_{s=1}^L \beta_{sj}(0)B_{sj} = \psi_{0j}. \quad (26)$$

Аналогичным образом, учитывая вторые условия в (22) и (24), для произвольных значений параметра B_{sj} из (25) получим, что

$$v_j(l) = \alpha_j(l) + \sum_{s=1}^L \beta_{sj}(l)B_{sj} = \psi_{1j}. \quad (27)$$

Условия (26), (27) означают, что функции $v_j(x)$, определенные из (25), для произвольных значений B_{sj} , $s = 1, \dots, L$, $j = 1, \dots, N$ удовлетворяют условиям (16), (17).

Для проверки условия, что $v_j(x)$, $j = 1, \dots, N$ удовлетворяют системе дифференциальных уравнений (20), сначала продифференцируем обе части равенства (25), имеем

$$v'_j(x) = \alpha'_j(x) + \sum_{s=1}^L \beta'_{sj}(x)B_{sj}, \quad x \in [0, l], \quad (28)$$

$$v''_j(x) = \alpha''_j(x) + \sum_{s=1}^L \beta''_{sj}(x)B_{sj}, \quad x \in [0, l]. \quad (29)$$

Подставляя (25), (28), (29) в левую часть уравнения (20), получим

$$\begin{aligned}
 & v''_j(x) + \tilde{a}_{1j}(x)v'_j(x) + \tilde{a}_{2j}(x)v_j(x) + \tilde{f}_j(x) + \sum_{s=1}^L \tilde{C}_{sj}(x)B_{sj} = \\
 & = \alpha''_j(x) + \sum_{s=1}^L \beta''_{sj}(x)B_{sj} + \tilde{a}_{1j}(x)\alpha'_j(x) + \tilde{a}_{1j}(x) \sum_{s=1}^L \beta'_{sj}(x)B_{sj} + \\
 & + \tilde{a}_{2j}(x)\alpha_j(x) + \tilde{a}_{2j}(x) \sum_{s=1}^L \beta_{sj}(x)B_{sj} + \tilde{f}_j(x) + \sum_{s=1}^L \tilde{C}_{sj}(x)B_{sj}.
 \end{aligned} \tag{30}$$

После некоторых преобразований и группировки с учетом (21), (23) из (30), получим:

$$\begin{aligned}
 & v''_j(x) + \tilde{a}_{1j}(x)v'_j(x) + \tilde{a}_{2j}(x)v_j(x) + \tilde{f}_j(x) + \sum_{s=1}^L \tilde{C}_{sj}(x)B_{sj} = \\
 & = [\alpha''_j(x) + \tilde{a}_{1j}(x)\alpha'_j(x) + \tilde{a}_{2j}(x)\alpha_j(x) + \tilde{f}_j(x)] + \\
 & + \sum_{s=1}^L \left(\beta''_{sj}(x) + \tilde{a}_{1j}(x)\beta'_{sj}(x) + \tilde{a}_{2j}(x)\beta_{sj}(x) + \tilde{C}_{sj}(x) \right) B_{sj} = 0, \quad j = 1, \dots, N, \quad x \in (0, l).
 \end{aligned}$$

Следовательно, функции $v_j(x)$, определенные из (25), удовлетворяют системе дифференциальных уравнений (20). Отсюда следует справедливость утверждения теоремы.

Несложно доказывается следующая теорема.

Теорема 2. Для произвольных значений параметра B_{sj} представление (25) для решения дифференциального уравнения (20) с краевыми условиями (16), (17) единственно.

Доказательство. Предположим противное: пусть для какого-либо значения параметра B_{sj} задачи (20), (16), (17) имеются два решения $\bar{v}_j(x)$ и $\tilde{v}_j(x)$:

$$\bar{v}_j(x) = \bar{\alpha}_j(x) + \sum_{s=1}^L \bar{\beta}_{sj}(x)B_{sj}, \quad x \in [0, l], \tag{31}$$

$$\tilde{v}_j(x) = \tilde{\alpha}_j(x) + \sum_{s=1}^L \tilde{\beta}_{sj}(x)B_{sj}, \quad x \in [0, l]. \tag{32}$$

Ввиду линейности задачи (20), (16), (17), ясно, что разность $v_j(x) = \bar{v}_j(x) - \tilde{v}_j(x)$ будет решением следующего однородного уравнения

$$v''_j(x) + \tilde{a}_{1j}(x)v'_j(x) + \tilde{a}_{2j}(x)v_j(x) = 0, \quad j = 1, \dots, N, \quad x \in (0, l) \tag{33}$$

с однородными краевыми условиями

$$v_j(0) = 0, \quad v_j(l) = 0. \tag{34}$$

Ясно, что однородная задача (33), (34) имеет только тривиальное решение $v_j(x) \equiv 0, x \in [0, l]$. В противном случае нарушается условие единственности решения первоначальной неоднородной задачи (20), (16), (17) [13]. Следовательно, $\bar{v}_j(x) = \tilde{v}_j(x), x \in [0, l]$, а $\bar{\alpha}_j(x) = \tilde{\alpha}_j(x), \bar{\beta}_{sj}(x) = \tilde{\beta}_{sj}(x)$. Отсюда следует единственность представления (25). Теорема доказана.

Решив отдельно для каждого j две краевые задачи (21), (22) и (23), (24), далее используя условие (18) и представление (25), получим равенство

$$v_j(\bar{x}_s) = \alpha_j(\bar{x}_s) + \sum_{s=1}^L \beta_{sj}(\bar{x}_s)B_{sj} = \psi_{2s,j}, \quad s = 1, \dots, L, \tag{35}$$

представляющее собой алгебраическую систему уравнений, из которой можно определить идентифицируемый вектор $B_s, s = 1, \dots, L$. Учитывая, что L — число неизвестных

функций, участвующих в уравнении (1), как правило, в реальных задачах невелико, для решения алгебраической системы уравнений (35) можно использовать какие-либо известные методы, например, Гаусса или итеративные методы.

Разрешимость системы (35) зависит от разрешимости обратной задачи (20), (16)–(18) и обратно, если система (35) не имеет решения, то не имеет решения задача (20), (16)–(18), а следовательно, и исходная задача (1)–(5). Таким образом, свойства существования и единственности решения системы (35) и исходной обратной задачи (1)–(5) взаимосвязаны.

Из решения задачи (20), (16), (17), определяется функция $v_j(x)$, $x \in [0, l]$. После этого процедура (21)–(25), (35) повторяется на прямой $t = t_{j+1}$, на которой определяется $v_{j+1}(x)$.

Таким образом, для определения компонент вектора параметров B_s , $s = 1, \dots, L$ необходимо решить N раз краевую задачу относительно $(L + 1)$ независимых дифференциальных уравнений второго порядка. Вычисленный вектор $B_s = (B_s(t_1), \dots, B_s(t_N))^T$ в дальнейшем с применением методов интерполяции или аппроксимации может быть использован для получения аналитического вида функции $B_s(t)$.

3. Результаты численных экспериментов

Приведем результаты решения следующих задач параметрической идентификации. Для численного решения задач численным методом, предложенным в разд. 2, разработано программное обеспечение в среде Embarcadero RAD Studio Delphi XE8. Численные эксперименты проводились на персональном компьютере с 64-разрядным, двухъядерным процессором Intel Core i7-4510U, частотой 2,0 ГГц, с объемом оперативной памяти 8 Gb.

Пример 1. Рассмотрим задачу

$$\begin{aligned} \frac{\partial v(x, t)}{\partial t} &= a(x) \frac{\partial^2 v(x, t)}{\partial x^2} + \\ B(t)e^{2x} \left[\frac{1}{7}x - 4a(x)(x + 1) \right], & \quad (x, t) \in \Omega = \{(x, t): 0 < x < 1, 0 < t \leq 1\}, \\ v(x, 0) &= xe^{2x}, \quad x \in [0, 1], \\ v(0, t) &= 0, \quad v(1, t) = e^{\frac{t}{7}+2}, \quad \frac{\partial v(0, t)}{\partial x} = e^{\frac{t}{7}}, \quad t \in [0, 1], \end{aligned}$$

где $a(x) = \frac{\cos x}{e^x}$. Точными решениями данной задачи является функции:

$$B(t) = e^{\frac{t}{7}}, \quad v(x, t) = xe^{\frac{t}{7}+2x}.$$

Численные эксперименты проводились при разном числе N прямых $t = t_j$, $j = 1, \dots, N$. Для решения вспомогательных задач использовался метод Рунге—Кутты четвертого порядка при различных шагах h_x .

В табл. 1 приведены результаты решения задачи и времени выполнения программного обеспечения при числе прямых $N = 100, 200, 500$.

Были проведены расчеты при наличии случайных помех в функции $\frac{\partial v(0, t)}{\partial x}$, обусловленных погрешностями измерения состояния $v(0, t)$ на левом конце, которые определялись следующим образом:

$$\left(\frac{\partial v(0, t)}{\partial x} \right)^\sigma = e^{\frac{t}{7}}(1 + \sigma \text{rand}).$$

Здесь σ — величина, определяющая уровень погрешности в измерениях, rand — случайные числа, равномерно распределенные на интервале $[-1, 1]$, полученные с использованием функции MATLAB rand .

Таблица 1

Полученное и точные значения $B(t)$ для примера 1

t_j	$N = 100$		$N = 200$		$N = 500$		Точное $B(t_j)$
	Время: 0,61 сек.		Время: 0,96 сек.		Время: 2,20 сек.		
	$B(t_j)$	$ \Delta B(t_j) $	$B(t_j)$	$ \Delta B(t_j) $	$B(t_j)$	$ \Delta B(t_j) $	
0,10	1,01403	0,00036	1,01425	0,00014	1,01433	0,00006	1,01439
0,20	1,02856	0,00042	1,02883	0,00016	1,02892	0,00006	1,02898
0,30	1,04336	0,00043	1,04363	0,00016	1,04372	0,00007	1,04379
0,40	1,05837	0,00044	1,05864	0,00016	1,05874	0,00007	1,05881
0,50	1,07360	0,00045	1,07388	0,00017	1,07397	0,00007	1,07404
0,60	1,08904	0,00045	1,08933	0,00017	1,08943	0,00007	1,08949
0,70	1,10471	0,00046	1,10500	0,00017	1,10510	0,00007	1,10517
0,80	1,12061	0,00046	1,12090	0,00017	1,12100	0,00007	1,12107
0,90	1,13673	0,00047	1,13703	0,00018	1,13713	0,00007	1,13720

В табл. 2 приведены результаты решения примера 1 при $N = 500$, $h_x = 0,002$ для уровней погрешности, равных $\sigma = 1\%$, $\sigma = 3\%$ и $\sigma = 5\%$, а также без наличия шума, т.е. $\sigma = 0\%$. На рис. 1 даны графики точных значений коэффициента $B(t)$ (аналитическое решение) и полученных численным методом, предложенным в разд. 2 при различных уровнях шума σ .

Таблица 2

Значения коэффициента $B(t)$ для примера 1

j	t_j	Значение $B(t)$				
		Точное	Полученное			
			$\sigma = 0\%$	$\sigma = 1\%$	$\sigma = 3\%$	$\sigma = 5\%$
25	0,05	1,007168	1,007117	1,004635	0,999671	0,994708
50	0,10	1,014388	1,014331	1,011494	1,005821	1,000148
75	0,15	1,021660	1,021599	1,018857	1,013373	1,007889
100	0,20	1,028984	1,028920	1,026556	1,021827	1,017098
125	0,25	1,036360	1,036295	1,034525	1,030986	1,027446
150	0,30	1,043789	1,043724	1,042709	1,040681	1,038653
175	0,35	1,051271	1,051205	1,051040	1,050710	1,050379
200	0,40	1,058807	1,058741	1,059438	1,060833	1,062228
225	0,45	1,066397	1,066330	1,067822	1,070805	1,073787
250	0,50	1,074041	1,073974	1,076113	1,080392	1,084670
275	0,55	1,081741	1,081673	1,084251	1,089406	1,094561
300	0,60	1,089495	1,089427	1,092191	1,097718	1,103246
325	0,65	1,097305	1,097236	1,099916	1,105275	1,110634
350	0,70	1,105171	1,105102	1,107435	1,112100	1,116766
375	0,75	1,113093	1,113024	1,114782	1,118297	1,121813
400	0,80	1,121072	1,121002	1,122013	1,124035	1,126056
425	0,85	1,129109	1,129038	1,129203	1,129533	1,129862
450	0,90	1,137203	1,137132	1,136434	1,135040	1,133645
475	0,95	1,145355	1,145283	1,143792	1,140810	1,137827

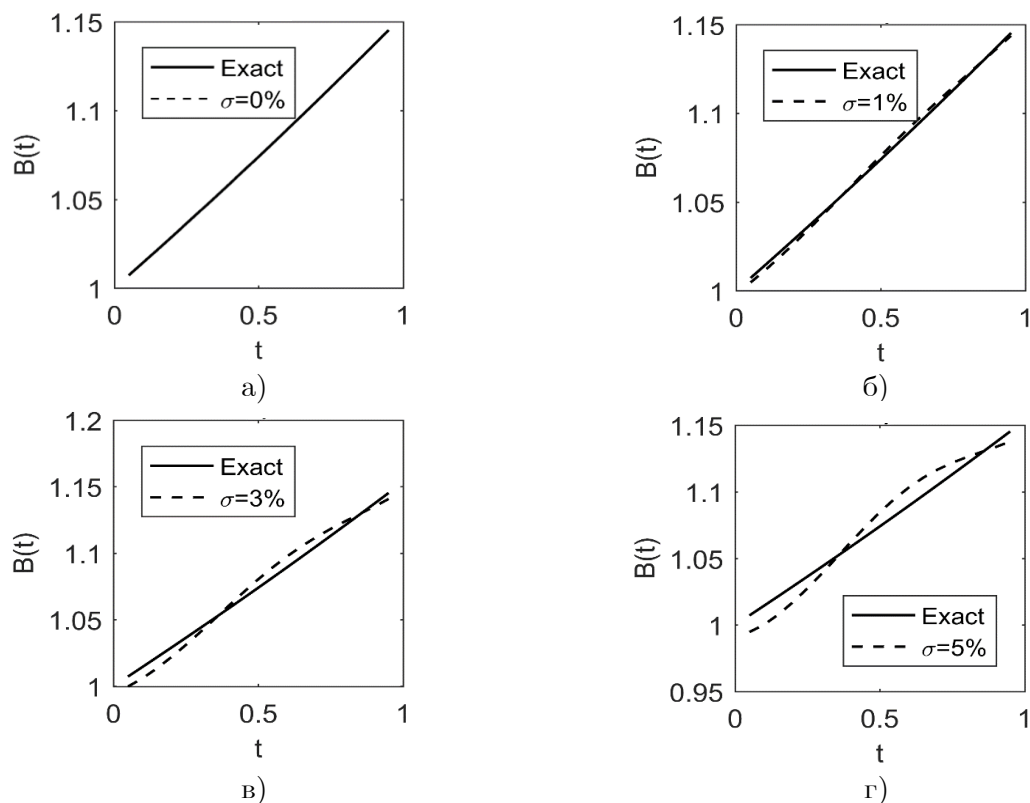


Рис. 1. Графики точного (Exact) и полученного численными методами коэффициента $B(t)$ при различных уровнях шума для примера 1

Численные эксперименты проводились при разном числе N прямых $t = t_j$, $j = 1, \dots, N$. Для решения вспомогательных задач использовался метод Рунге—Кутты четвертого порядка при различных шагах h_x .

В табл. 3 приведены результаты решения задачи и времени выполнения программного обеспечения при числе прямых $N = 100, 200, 500$.

Таблица 3

Полученное и точные значения $B(t)$ для примера 2

t_j	$N = 100$		$N = 200$		$N = 500$		Точное $B(t_j)$
	Время: 0,47 сек.		Время: 0,82 сек.		Время: 1,86 сек.		
	$B(t_j)$	$ \Delta B(t_j) $	$B(t_j)$	$ \Delta B(t_j) $	$B(t_j)$	$ \Delta B(t_j) $	
0,10	1,05042	0,00085	1,05079	0,00048	1,05093	0,00034	1,05127
0,20	1,10416	0,00102	1,10461	0,00056	1,10480	0,00037	1,10517
0,30	1,16071	0,00112	1,16122	0,00062	1,16144	0,00040	1,16183
0,40	1,22020	0,00120	1,22075	0,00066	1,22099	0,00042	1,22140
0,50	1,28275	0,00127	1,28333	0,00070	1,28359	0,00044	1,28403
0,60	1,34852	0,00134	1,34913	0,00073	1,34940	0,00046	1,34986
0,70	1,41765	0,00141	1,41830	0,00077	1,41858	0,00049	1,41907
0,80	1,49034	0,00149	1,49101	0,00081	1,49131	0,00051	1,49182
0,90	1,56675	0,00156	1,56746	0,00085	1,56777	0,00054	1,56831

Были проведены расчеты при наличии случайных помех в функции $\frac{\partial v(0,t)}{\partial x}$, обусловленных погрешностями измерения состояния $v(0,t)$ на левом конце, которые определялись следующим образом:

$$\left(\frac{\partial v(0,t)}{\partial x}\right)^\sigma = \sigma \text{rand},$$

здесь σ — величина, определяющая уровень погрешности в измерениях, rand — случайные числа, равномерно распределенные на интервале $[-1, 1]$, полученные с использованием функции MATLAB rand .

В табл. 4 приведены результаты решения примера 2 при $N = 500$, $h_x = 0,002$ для уровней погрешности равных $\sigma = 1\%$, $\sigma = 3\%$ и $\sigma = 5\%$, а также без наличия шума, т.е. $\sigma = 0\%$. На рис. 2 даны графики точных значений коэффициента $B(t)$ (аналитическое решение) и полученных численным методом, предложенным в разд. 2 при различных уровнях шума σ .

Результаты большого числа проведенных численных экспериментов по решению различных тестовых обратных задач по определению коэффициентов $B(t)$ показали следующее.

Точность решения обратных задач, как и следовало ожидать, существенно зависит от числа используемых прямых N в методе прямых для аппроксимации исходной краевой задачи.

Для задачи идентификации коэффициента $B(t)$ увеличение числа прямых существенно не сталкивается с вычислительными проблемами, т.к. задача (20), (16), (17) на каждой прямой $t = t_j$ решается независимо и последовательно при $j = 1, 2, \dots, N$. Решение этой задачи возможно практически с любой заданной точностью с использованием известных эффективных численных методов решения задач Коши.

Таблица 4

Значения коэффициента $B(t)$ для примера 2

j	t_j	Значение $B(t)$				
		Точное	Полученное			
			$\sigma = 0\%$	$\sigma = 1\%$	$\sigma = 3\%$	$\sigma = 5\%$
25	0,05	1,025315	1,024989	1,032816	1,048470	1,064124
50	0,10	1,051271	1,050927	1,062911	1,086879	1,110846
75	0,15	1,077884	1,077526	1,092253	1,121709	1,151165
100	0,20	1,105171	1,104799	1,120715	1,152548	1,184380
125	0,25	1,133148	1,132765	1,148240	1,179190	1,210140
150	0,30	1,161834	1,161439	1,174912	1,201858	1,228805
175	0,35	1,191246	1,190840	1,200963	1,221208	1,241454
200	0,40	1,221403	1,220985	1,226748	1,238273	1,249797
225	0,45	1,252323	1,251894	1,252720	1,254371	1,256022
250	0,50	1,284025	1,283586	1,279386	1,270986	1,262586
275	0,55	1,316531	1,316080	1,307260	1,289621	1,271982
300	0,60	1,349859	1,349396	1,336817	1,311659	1,286501
325	0,65	1,384031	1,383557	1,368447	1,338228	1,308009
350	0,70	1,419068	1,418581	1,402419	1,370094	1,337770
375	0,75	1,454991	1,454493	1,438859	1,407591	1,376323
400	0,80	1,491825	1,491314	1,477738	1,450586	1,423435
425	0,85	1,529590	1,529066	1,518877	1,498499	1,478121
450	0,90	1,568312	1,567775	1,561970	1,550359	1,538749
475	0,95	1,608014	1,607463	1,606610	1,604904	1,603197

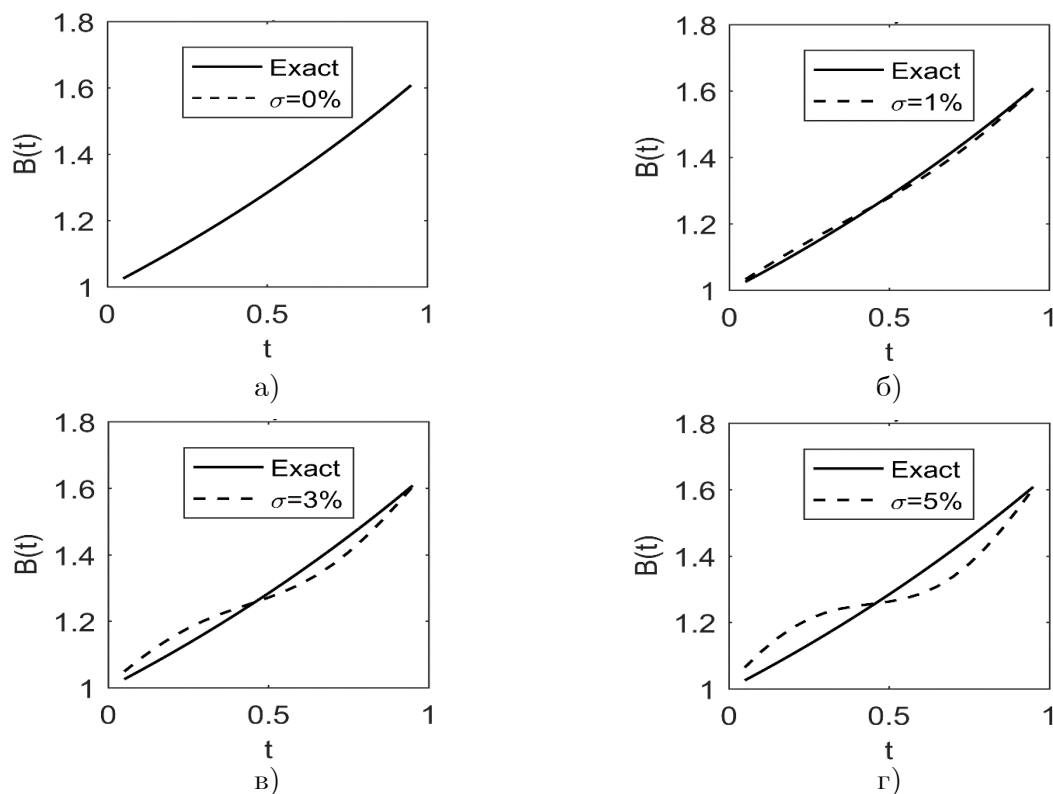


Рис. 2. Графики точного (Exact) и полученного численными методами коэффициента $B(t)$ при различных уровнях шума для примера 2

Заключение

В работе рассмотрена обратная задача по идентификации источника специального вида в линейных параболических уравнениях. Предложена методика численного решения задачи с применением метода прямых, основанная на использовании представления специального вида для решения полученной системы обыкновенных дифференциальных уравнений. Эффективность предложенного в работе численного метода решения задач параметрической идентификации для параболического уравнения обоснована тем, что он приводится к решению вспомогательных, хорошо исследованных задач, и не требует построения каких-либо итерационных процедур. К рассмотренным коэффициентно-обратным задачам сводятся часто встречающиеся на практике прямые задачи с нелокальными начальными или граничными условиями.

В качестве перспективных направлений исследований отметим применимость этого подхода к другим типам дифференциальных уравнений, например, гиперболическим уравнениям, и к другим видам локальных и нелокальных условий.

Автор выражает искреннюю благодарность проф. К.Р. Айда-заде за ценные советы и внимание к работе.

Литература

1. Айда-заде К.Р., Рагимов А.Б. Решение классов коэффициентно-обратных задач и задач с нелокальными условиями для параболических уравнений // Дифференциальные уравнения. 2015. Т. 51, № 1. С. 84–94. DOI: 10.1134/S0012266115010085.

2. Белавин В.А., Капица С.П., Курдюмов С.П. Математическая модель глобальных демографических процессов с учетом пространственного распределения // Журнал вычислительной математики и математической физики. 1999. Т. 38, № 6. С. 885–902.
3. Водахова В.А. Краевая задача с нелокальным условием А.М.Нахушева для одного псевдопараболического уравнения влагопереноса // Дифференциальные уравнения. 1982. Т. 18, № 2. С. 280–285.
4. Ионкин Н.И. Решение одной краевой задачи теории теплопроводности с неклассическим краевым условием // Дифференциальные уравнения. 1977. Т. 13, № 2. С. 294–304.
5. Кожанов А.И. Обратные задачи восстановления правой части специального вида в параболическом уравнении // Математические заметки СВФУ. 2016. Т. 23, № 4. С. 31–45.
6. Нахушев А.М. Уравнения математической биологии. М.: Высшая школа, 1995. 305 с.
7. Прилепко А.И., Соловьев В.В. Теоремы разрешимости и метод Ротэ в обратных задачах для уравнения параболического типа. I // Дифференциальные уравнения. 1987. Т. 23, № 10. С. 1791–1799.
8. Прилепко А.И., Ткаченко Д.С. Корректность обратной задачи об источнике для параболических систем // Дифференциальные уравнения. 2004. Т. 40, № 11. С. 1540–1547.
9. Пулькина Л.С. Об одном классе нелокальных задач и их связи с обратными задачами // Труды третьей всероссийской научной конференции «Дифференциальные уравнения и краевые задачи, Математическое моделирование и краевые задачи». Ч. 3. Самара: Изд-во СамГТУ, 2006. С. 190–192.
10. Самарский А.А. О некоторых проблемах современной теории дифференциальных уравнений // Дифференциальные уравнения. 1980. Т. 16, № 11. С. 1221–1228.
11. Соловьев В.В. Определение источника и коэффициентов в параболическом уравнении в многомерном случае // Дифференциальные уравнения. 1995. Т. 31, № 6. С. 1060–1069.
12. Соловьев В.В. Существование решения в «целом» обратной задачи определения источника в квазилинейном уравнении параболического типа // Дифференциальные уравнения. 1996. Т. 32, № 4. С. 536–544.
13. Тихонов А.Н., Васильева А.Б., Свешников А.Г. Дифференциальные уравнения. М.: Физматлит, 2005. 256 с.
14. Ahmadabadi M. Nili, Arab M., Maalek Ghaini F.M. The method of fundamental solutions for the inverse space-dependent heat source problem // Engineering Analysis with Boundary Elements. 2009. Vol. 33. P. 1231–1235. DOI: 10.1016/j.enganabound.2009.05.001.
15. Aida-zade K.R., Rahimov A.B. An approach to numerical solution of some inverse problems for parabolic equations // Inverse Problems in Science and Engineering. 2014. Vol. 22, no. 1. P. 96–111. DOI: 10.1080/17415977.2013.827184.
16. Bouziani A., Benouar N.-E. Probleme mixte avec conditions integrales pour une classe d'equations paraboliques // Comptes Rendus de l'Academie des Sciences. Paris, Serie 1. 1995. Vol. 321. P. 1177–1182.
17. Farcas A., Lesnic D. The boundary-element method for the determination of a heat source dependent on one variable // Journal of Engineering Mathematics. 2006. Vol. 54. P. 375–388. DOI: 10.1007/s10665-005-9023-0.
18. Hasanov A. Identification of spacewise and time dependent source terms in 1D heat conduction equation from temperature measurement at a final time // Int. J. of Heat and Mass Transfer. 2012. Vol. 55. P. 2069–2080. DOI: 10.1016/j.ijheatmasstransfer.2011.12.009.

19. Hasanov A. An inverse source problem with single Dirichlet type measured output data for a linear parabolic equation // Applied Mathematics Letters. 2011. Vol. 24. P. 1269–1273. DOI: 10.1016/j.aml.2011.02.023.
20. Hasanov A., Otelbaev M., Akpayev B. Inverse heat conduction problems with boundary and final time measured output data // Inverse Problems in Science and Engineering. 2011. Vol. 19, no. 7. P. 895–1006. DOI: 10.1080/17415977.2011.565931.
21. Ismailov M.I., Kanca F., Lesnic D. Determination of a time-dependent heat source under nonlocal boundary and integral overdetermination conditions // Applied Mathematics and Computation. 2011. Vol. 218. P. 4138–4146. DOI: 10.1016/j.amc.2011.09.044.
22. Ivanchov M.I. Inverse Problems for Equations of Parabolic Type. VNTL Publications, Lviv, Ukraine, 2003. 238 p.
23. Johansson T., Lesnic D. A variational method for identifying a spacewise-dependent heat source // IMA Journal of Applied Mathematics. 2007. Vol. 72, no. 6. P. 748–760. DOI: 10.1093/imamat/hxm024.
24. Rothe E. Zweidimensionale parabolische Randwertaufgaben als Grenzfall eindimensionaler Randwertaufgaben // Mathematische Annalen. 1930. Vol. 102, no. 1. P. 650–670.
25. Prilepko A.I., Orlovsky D.G., Vasin I.A. Methods for solving inverse problems in mathematical physics. M. Dekker, New York, 2000. 709 p.
26. Yan L., Fu C.L., Yang F.L. The method of fundamental solutions for the inverse heat source problem // Engineering Analysis with Boundary Elements. 2008. Vol. 32. P. 216–222. DOI: 10.1016/j.enganabound.2007.08.002.

Рагимов Анар Бейбала, к.ф.-м.н., доцент, ведущий научный сотрудник, Институт Систем Управления НАН Азербайджана (Баку, Азербайджан)

DOI: 10.14529/cmse200403

ON THE PROBLEM OF RESTORING TIME COEFFICIENT-FUNCTIONS OF SPECIAL TYPE SOURCE IN PARABOLIC EQUATION

© 2020 A.B. Rahimov

The Institute of Control Systems, Azerbaijan National Academy of Science

(B. Vahabzadeh 9, Baku, AZ1141 Azerbaijan)

E-mail: anar_r@yahoo.com

Received: 05.03.2020

An inverse problem of identifying coefficients depend only on time for a special form source in a linear parabolic equation with point overdetermined conditions. In particular, boundary value problems with nonlocal (integral) boundary conditions are reduced to such problems. The specificity of the problems is that the identifiable parameters depend only on a time variable and are factors of the coefficients of the right-hand side of the equation. A method for numerically solving the problem using the method of lines is proposed, based on using a special type representation of solution. By applying the method of lines, the problem is reduced to a parametric inverse problem with respect to an ordinary differential equations system. For its solution, a special type of representation of this solution is proposed. To solve this problem, auxiliary boundary value problems are constructed that determine a solution to the initial problem. The most important in this work is that the proposed approach to the numerical solution to the investigated inverse problem of identifying the coefficients does not require (in contrast to previously known methods) to construct any iterative procedure. The results of numerical experiments in the form of tables and graphs obtained by solving the test problems, and their analysis are provided.

Keywords: inverse problem, nonlocal conditions, method of lines, parabolic equation, parametric identification.

FOR CITATION

Rahimov A.B. On the Problem of Restoring Time Coefficient-Functions of Special Type Source in Parabolic Equation. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2020. Vol. 9, no. 4. P. 33–47. (in Russian) DOI: 10.14529/cmse200403.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Aida-zade K.R., Rahimov A.B. Solution to classes of inverse coefficient problems and problems with nonlocal conditions for parabolic equations. *Differential Equations*. 2015. Vol. 51, no. 1. P. 83–93. DOI: 10.1134/S0012266115010085.
2. Belavin V.A., Kapitza S.P., Kurdyumov S.P. A mathematical model of global demographic processes with regard to the spatial distribution. *Computational Mathematics and Mathematical Physics*. 1998. Vol. 38, no. 6. P. 849–865.
3. Vodahova V.A. A boundary value problem with A.M. Nakhushev's nonlocal condition for a pseudoparabolic equation of moisture transfer. *Differential Equations*. 1982. Vol. 18, no. 2. P. 280–285. (in Russian)
4. Ionkin N.I. The solution of a certain boundary value problem of the theory of heat conduction with a nonclassical boundary condition. *Differential Equations*. 1977. Vol. 13, no. 2. P. 294–304. (in Russian)
5. Kozhanov A.I. A Nonlinear Loaded Parabolic Equation and a Related Inverse Problem. *Mathematical Notes*. 2004. Vol. 76, no. 78. P. 784–795. DOI: 10.1023/B:MATN.0000049678.16540.a5.
6. Nakhushev A.M. *Equations of mathematical biology*. M.: Vysshaya Shkola, 1995. 305 p. (in Russian)
7. Prilepko A.I., Solov'ev V.V. Solvability theorems and the rothe method in inverse problems for an equation of parabolic type. I. *Differential Equations*. 1987. Vol. 23, no. 10. P. 1791–1799. (in Russian)
8. Prilepko A.I., Tkachenko D.S. Well-posedness of the inverse source problem for parabolic systems. *Differential Equations*. 2004. Vol. 40, no. 11. P. 1619–1626.
9. Pulkina L.S. On one class of nonlocal problems and their connection with inverse problems. *Proceedings of the Third All-Russia scientific conference “Mathematical modelling and boundary value problems”, Part 3: Differential equations and boundary value problems*. Samara: Publishing of the Samara State Technical University, 2006. P. 190–192. (in Russian)
10. Samarskii A.A. On some problems of the modern theory of differential equations. *Differential Equations*. 1980. Vol. 16, no. 11. P. 1221–1228. (in Russian)
11. Solov'ev V.V. Determination of a source and coefficients in a parabolic equation in the multidimensional case. *Differential Equations*. 1995. Vol. 31, no. 6. P. 1060–1069. (in Russian)
12. Solov'ev V.V. Existence of solution as a “whole” of inverse problem of determining the source in a quasilinear equation of parabolic type. *Differential Equations*. 1996. Vol. 32, no. 4. P. 536–544. (in Russian)
13. Tikhonov A.N., Vasilyeva A.B., Sveshnikov A.Q. *Differential equations*. M.: Fizmatlit, 2005. 256 p. (in Russian)

14. Ahmadabadi M. Nili, Arab M., Maalek Ghaini F.M. The method of fundamental solutions for the inverse space-dependent heat source problem. *Engineering Analysis with Boundary Elements*. 2009. Vol. 33. P. 1231–1235. DOI: 10.1016/j.enganabound.2009.05.001.
15. Aida-zade K.R., Rahimov A.B. An approach to numerical solution of some inverse problems for parabolic equations. *Inverse Problems in Science and Engineering*. 2014. Vol. 22, no. 1. P. 96–111. DOI: 10.1080/17415977.2013.827184.
16. Bouziani A., Benouar N.-E. Probleme mixte avec conditions integrales pour une classe d'equations paraboliques. *Comptes Rendus de l'Academiedes Sciences. Paris, Serie 1*. 1995. Vol. 321. P. 1177–1182.
17. Farcas A., Lesnic D. The boundary-element method for the determination of a heat source dependent on one variable. *Journal of Engineering Mathematics*. 2006. Vol. 54. P. 375–388. DOI: 10.1007/s10665-005-9023-0.
18. Hasanov A. Identification of spacewise and time dependent source terms in 1d heat conduction equation from temperature measurement at a final time. *Int. J. of Heat and Mass Transfer*. 2012. Vol. 55. P. 2069–2080. DOI: 10.1016/j.ijheatmasstransfer.2011.12.009.
19. Hasanov A. An inverse source problem with single dirichlet type measured output data for a linear parabolic equation. *Applied Mathematics Letters*. 2011. Vol. 24. P. 1269–1273. DOI: 10.1016/j.aml.2011.02.023.
20. Hasanov A., Otelbaev M., Akpayev B. Inverse heat conduction problems with boundary and final time measured output data. *Inverse Problems in Science and Engineering*. 2011. Vol. 19, no. 7. P. 895–1006. DOI: 10.1080/17415977.2011.565931.
21. Ismailov M.I., Kanca F., Lesnic D. Determination of a time-dependent heat source under nonlocal boundary and integral overdetermination conditions. *Applied Mathematics and Computation*. 2011. Vol. 218. P. 4138–4146. DOI: 10.1016/j.amc.2011.09.044.
22. Ivanchov M.I. *Inverse problems for equations of parabolic type*. VNTL Publications, Lviv, Ukraine, 2003. 238 p.
23. Johansson T., Lesnic D. A variational method for identifying a spacewise-dependent heat source. *IMA Journal of Applied Mathematics*. 2007. Vol. 72, no. 6. P. 748–760. DOI: 10.1093/imamat/hxm024.
24. Rothe E. Zweidimensionale parabolische randwertaufgaben als grenzfall eindimensionaler randwertaufgaben. *Mathematische Annalen*. 1930. Vol. 102, no. 1. P. 650–670.
25. Prilepko A.I., Orlovsky D.G., Vasin I.A. *Methods for solving inverse problems in mathematical physics*. M. Dekker, New York, 2000. 709 p.
26. Yan L., Fu C.L., Yang F.L. The method of fundamental solutions for the inverse heat source problem. *Engineering Analysis with Boundary Elements*. 2008. Vol. 32. P. 216–222. DOI: 10.1016/j.enganabound.2007.08.002.

ДВУМЕРНАЯ МОДЕЛЬ ФАСЕТОЧНОГО ЗРЕНИЯ

© 2020 Р.С. Федянина, Л.Б. Соколинский

Южно-Уральский государственный университет

(454080 Челябинск, пр. им. В.И. Ленина, д. 76)

E-mail: raisa.fedianina@susu.ru, leonid.sokolinsky@susu.ru

Поступила в редакцию: 26.10.2020

Одним из современных перспективных направлений исследований в области машинного зрения являются системы визуализации на основе искусственных фасеточных глаз, в основе которых лежат принципы организации зрения насекомых. Такие глаза имеют сложную структуру, основными элементами которой являются зрительные элементы, называемые омматидиями. В данной работе строится математическая модель бинокулярного фасеточного зрения в двумерном пространстве. Модель позволяет описывать задачи навигации робототехнического устройства и детектирования объектов на плоскости. В предложенной модели фасеточный глаз является правильным многоугольником с обзором 360° . Основными параметрами модели являются количество фасеток (омматидиев) в одном глазу, радиус описанной вокруг глаза окружности и расстояние между глазами. Модель предполагает, что области видимости омматидиев одного глаза не пересекаются. В качестве наблюдаемых объектов фигурируют окружности различных диаметров. Выводится формула вычисления номера омматидия, в область видимости которого попадает точка с заданными координатами. Доказывается утверждение о необходимом и достаточном условии попадания заданной окружности в область видимости омматидия с заданным номером. Предлагается алгоритм построения обучающей выборки для искусственной нейронной сети, работающей на основе модели двумерного фасеточного зрения.

Ключевые слова: система визуализации, математическая модель фасеточного зрения, нейронная сеть, генерация обучающей выборки.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Федянина Р.С., Соколинский Л.Б. Двумерная модель фасеточного зрения // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4 С. 48–66. DOI: 10.14529/cmse200404.

Введение

Одной из актуальных задач современной робототехники является создание полностью автономных робототехнических устройств с элементами искусственного интеллекта, способных выполнять определенные действия без участия оператора. Важным элементом подобных устройств является система машинного зрения. С ее помощью решаются задачи распознавания различных объектов и интерпретации сцен [1], определения расстояний до объектов, а так же задачи, связанные с ориентацией во внешнем пространстве. Например, задачи одновременной локализации и картографирования [2], задачи обнаружения и отслеживания движущихся объектов [3]. Обязательной частью системы машинного зрения является видеокамера или система видеокамер. В большинстве случаев в системах машинного зрения робототехнических устройств используются сложные видеокамеры с переменным фокусным расстоянием, под определенным углом наклона [4], работающие под управлением удаленного оператора. Задача полной автоматизации управления такими видеокамерами (позиционирование, фокусировка) является трудно решаемой. Поэтому при разработке полностью автономных робототехнических систем все больший интерес вызывают видеокамеры, созданные по образу и подобию фасеточных глаз насекомых, в которых нет движущихся частей, и которые не нужно фокусировать и позиционировать. Помимо этого, фасеточные глаза насекомых обладают целым рядом преимуществ. Они имеют большое угловое поле зрения (вплоть до

360°) [5] и низкую аберрацию [6]. Высокая контрастность и малая инерционность рецепторов фасеточных глаз обеспечивают способность различать быстро движущиеся объекты. Таким образом, скорость зрения некоторых насекомых достигает 250–300 кадров в секунду [7]. Фасеточные глаза обладают высокой чувствительностью. Они способны воспринимать свет в диапазоне от 300–700 нм, в том числе ультрафиолетовую часть спектра [8]. Также, фасеточные глаза характеризуются большой глубиной изображаемого пространства, чувствительностью к поляризации света, стереоскопичностью и способностью к цветовому восприятию [5, 7]. Несмотря на миниатюрные размеры, перечисленные выше особенности фасеточного зрения, позволяют насекомым с высокой эффективностью решать задачи навигации, идентификации цели, ее преследования и захвата. В [9] описывается способность краба-скрипача определять расстояние между объектами. В [10] исследуется способность стрекоз и хищных мух идентифицировать и преследовать быстро двигающуюся жертву. Проведенные эксперименты показывают, что время реакции стрекозы на изменение места положения жертвы составляет не более 25 мс.

На сегодняшний день, достигнуты определенные успехи в создании оптоэлектронных систем аналогов фасеточных глаз насекомых на основе микролинзовых матриц, размещенных на сферических поверхностях [11]. Размеры создаваемых устройств, не превышают нескольких мм, а их масса нескольких сотен миллиграмм [5]. Количество микролинз искусственного фасеточного глаза достигает 4400 шт. [12]. Все это определяет перспективность применения искусственных фасеточных глаз в автономных робототехнических устройствах. В соответствии с вышесказанным актуальной является задача разработки нейросетевых моделей на основе фасеточных глаз насекомых. Цель данной статьи — построить математическую модель двумерного фасеточного зрения (на плоскости) и создание на ее основе множества прецедентов для обучения искусственной нейронной сети, способной определять азимут и расстояние до наблюдаемого объекта.

Статья имеет следующую структуру. В разделе 1 дано описание строения фасеточных глаз насекомых, перечислены их основные типы. В разделе 2 приводится краткий обзор систем визуализации на основе искусственных фасеточных глаз. Раздел 3 посвящен описанию математической модели двумерного фасеточного зрения; приведены формулы для расчета номеров омматидиев, в область видимости которых попадает наблюдаемый объект. В разделе 4 описан алгоритм порождения прецедентов обучающей выборки для нейронных сетей, обрабатывающих сигналы, поступающие из системы двумерного фасеточного зрения. В заключении суммируются полученные результаты и намечаются направления дальнейших исследований.

1. Строение фасеточных глаз

Фасеточные глаза представляют собой основной парный орган зрения насекомых, ракообразных и некоторых других беспозвоночных. Они состоят из отдельных структурных единиц — омматидиев. Каждый омматидий является автономным приемником

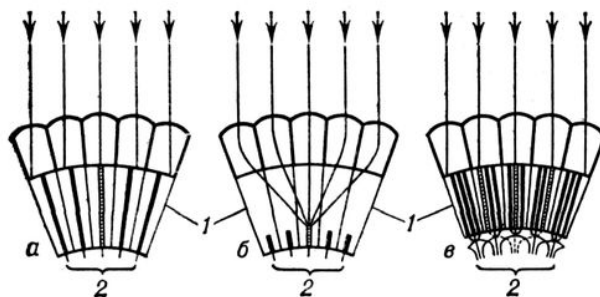


Рис. 1. Схема возникновения сетчаточного изображения в аппозиционном (а), оптико- (б) и нейросуперпозиционном (в) глазу. 1 – диоптрический аппарат омматидия; 2 – зрительные клетки с замкнутым (а и б) или открытым (в) рабдомом. Заштрихованы те рабдомеры, на которые попадают параллельно идущие лучи света (показаны стрелками) [13]

светового излучения. У разных насекомых число омматидиев различно, их число варьируется от нескольких единиц до 30 тысяч [7, 13]. Светопреломляющий аппарат омматидия состоит из роговичной линзы и кристаллического конуса. С внешней стороны роговица омматидия имеет вид выпуклой шестигранной фасетки. Каждый омматидий и соответствующая ему фасетка ведут себя как единый оптический блок, и чем больше таких единиц вписывается в данную область, тем выше разрешающая способность глаза. Светочувствительный аппарат омматидия состоит из ретинальных (зрительных) клеток с четко выраженными аксонами, связывающими их с оптическим центром мозга насекомого. Внутри каждой ретинальной клетки на стороне, обращенной внутрь омматидия, расположен рабдомер — особое образование из множества (до 75–100 тыс.) микроскопических трубочек-ворсинок, в мембране которых содержится зрительный пигмент. Совокупность рабдомеров всех зрительных клеток называется рабдом [13].

Различают три морфологических типа фасеточных глаз [13, 14]: аппозиционный, оптикосуперпозиционный и нейросуперпозиционный (см. рис. 1). Аппозиционные глаза приспособлены для работы при избытке света, поэтому они свойственны большинству насекомых активных в дневное время. Отличительной особенностью глаз данного типа является удлиненный рабдом и постоянная пигментная изоляция смежных омматидиев. Каждый омматидий передает только небольшую часть поля зрения, а конечное изображение всего поля обзора представляет собой совокупность смежных полей зрения всех омматидиев. В оптикосуперпозиционных глазах, при недостаточном освещении, гранулы пигмента могут «оседать» на дне омматидия, обеспечивая, таким образом, возможность проникновения в один и тот же омматидий лучей, прошедших через группу смежных фасеток. Оптикосуперпозиционные глаза встречаются, как правило, у насекомых, ведущих ночной образ жизни. Третий морфологический тип фасеточных глаз — нейросуперпозиционные глаза. У данного типа глаз отсутствует способность к суперпозиции лучей на сетчатке, как у оптикосуперпозиционных глаз, но имеет место суперпозиция сигналов, исходящих от зрительных клеток разных омматидиев, что обеспечивает им повышенную светосилу по сравнению с аппозиционными глазами. Такой тип глаза встречается у мух. Для построения автономных систем машинного зрения наибольший интерес представляет аппозиционный фасеточный глаз в силу его простоты. Везде далее рассматриваются фасеточные глаза именно такого типа.

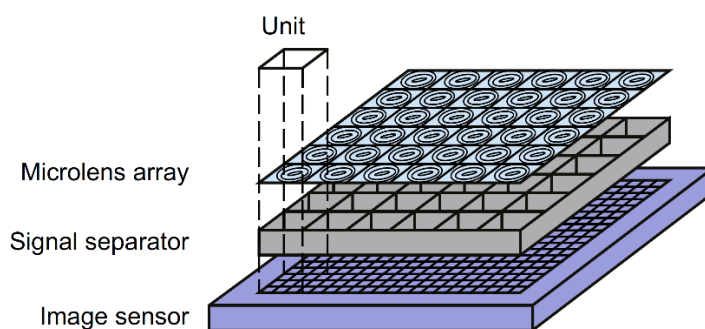


Рис. 2. Структура системы визуализации ТОМВО [16]

2. Искусственные фасеточные глаза

В данном разделе дается обзор современных систем визуализации на основе искусственных фасеточных глаз. В зависимости от типа используемого датчика изображения и формы матрицы микролинз, выделяются следующие три основных класса таких устройств [15]:

- 1) устройства на основе плоского датчика изображения и плоской матрицы микролинз;
- 2) устройства на основе плоского датчика изображения и изогнутой матрицы микролинз;
- 3) устройства на основе изогнутого датчика изображения и изогнутой матрицы микролинз.

Рассмотрим каждый класс более подробно.

Устройства на основе плоского датчика изображения и плоской матрицы микролинз состоят из трех компонент: матрица микролинз (microlens array), промежуточный слой (signal separator) и датчик изображения (image sensor), представленных на рис. 2, иллюстрирующем структуру системы визуализации ТОМВО [16]. Особенностью системы ТОМВО является то, что одна микролинза собирает оптические сигналы к блоку светочувствительных ячеек на датчике изображения. Соседние блоки разделены разделителем сигналов (signal separator) для предотвращения оптических перекрестных помех. Количество блоков формирования изображения системы, их размеры, а также тип и количество светочувствительных элементов может варьироваться, в зависимости от нужд пользователя. В качестве недостатка системы ТОМВО можно отметить то, что эта система позволяет получить четкое изображение только на определенном расстоянии до цели. Это объясняется плоской регулярной структурой матрицы микролинз. Такая матрица также характеризуется узким полем зрения. На основе архитектуры системы визуализации ТОМВО было построено несколько более совершенных систем с улучшенными характеристиками [17–20].

Появление следующего класса устройств на основе плоского датчика изображения и изогнутой матрицы микролинз обусловлено развитием технологий изготовления оптики. Типичным примером системы визуализации этого класса является система SCECam [12]. Изогнутый характер матрицы микролинз позволяет увеличить поле зрения системы визуализации. Однако при этом возникает проблема сопряжения изогнутой матрицы микролинз с плоским датчиком изображения. Для решения этой проблемы было предложено несколько подходов. В работе [12] предложена промежуточная

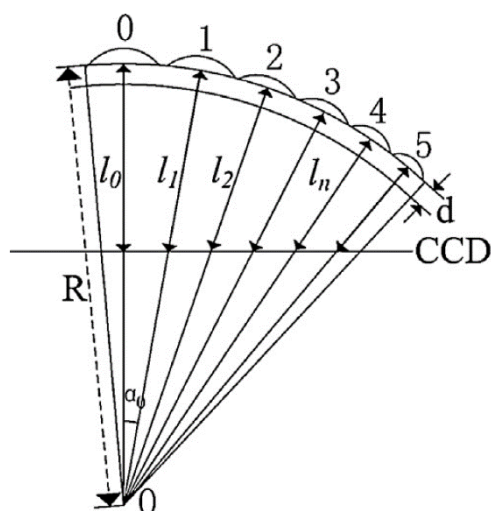


Рис. 3. Сопряжение изогнутой матрицы микролинз переменного диаметра с плоским датчиком изображения [21]

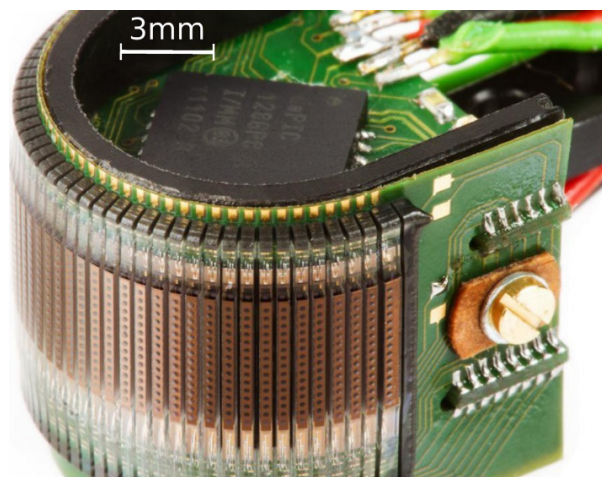


Рис. 4. Искусственный фасеточный глаз CurvACE [22]

оптическая система, отображающая сигналы, поступающие от изогнутой матрицы микролинз на плоский датчик. В [21] предложен подход базирующийся на использовании матрицы, состоящей из микролинз различных размеров. При данном подходе, по мере удаления микролинзы от центра матрицы ее диаметр уменьшается, что приводит к уменьшению фокусного расстояния (см. рис. 3). В этом случае отпадает необходимость использования промежуточной оптической системы.

Следующий этап в развитии систем визуализации на основе искусственных фасеточных глаз связан с созданием *устройств на основе изогнутого датчика изображения*. Представителем данного класса является система CurvACE [22], которая состоит из трех технологических слоев: матрица микролинз, матрица фотодетекторов и тонкая изогнутая печатная плата (рис. 4). Матрицы микролинз и фотодетекторов формируются из микрокомпонентов, количество которых соответствует количеству омматидиев. Данная конструкция характеризуется небольшими размерами, низким энергопотреблением, широким полем обзора и высокой чувствительностью.

Обзор показывает, что прогресс, достигнутый в разработке искусственных фасеточных глаз, позволяет уже сегодня создавать компактные устройства машинного зрения, применимые на практике.

3. Модель двумерного фасеточного зрения

С неформальной точки зрения плоскостная система искусственного фасеточного зрения включает в себя два одинаковых фасеточных глаза, неподвижно закрепленных на жестком горизонтальном стержне, к середине которого перпендикулярно крепится другой вертикальный стержень, задающий направление «вперед» (см. рис. 4). Центр горизонтального стержня называется центром системы фасеточного зрения. Горизонтальный и вертикальный стержни вместе с центром системы фасеточного зрения определяют оси декартовой системы координат, используемой для описания модели фасеточного зрения.

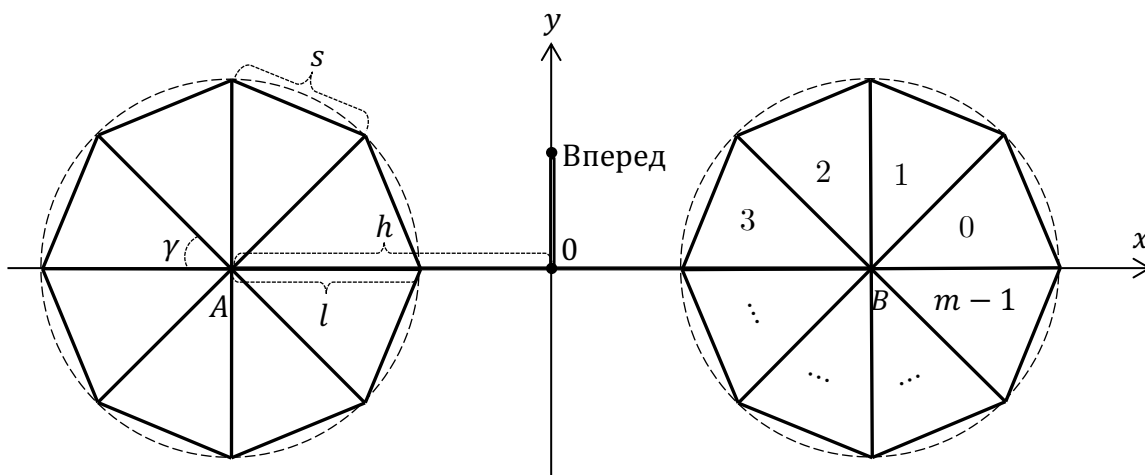


Рис. 5. Система плоскостного фасеточного зрения

В декартовой системе координат на плоскости фасеточные глаза представляют собой правильные многоугольники, вписанные в окружности радиуса $l > 0$, с центрами в точках $A = (-h, 0)$ и $B = (h, 0)$ соответственно, где h — некоторое положительное число, являющееся параметром модели. Предполагается, что

$$h > l. \quad (1)$$

Каждый многоугольник имеет m граней, где $m > 3$ — целое положительное число, также являющееся параметром модели. Таким образом, многоугольник делится на m конгруэнтных треугольников, каждый из которых является моделью омматидия. Внутренний угол омматидия вычисляется по формуле

$$\gamma = \frac{2\pi}{m}. \quad (2)$$

Внешняя грань омматидия, соответственно, имеет длину

$$s = 2l \sin\left(\frac{\pi}{m}\right). \quad (3)$$

Будем предполагать, что направление «вперед» соответствует положительному направлению оси ординат. Будем также предполагать, что ось абсцисс совпадает с границей между двумя омматидиями.

Построим математическую модель проекции внешнего мира на растр зрительных клеток. Рассмотрим сначала простейший случай, когда видимый объект представляет собой точку с координатами (x, y) в контексте рис. 5. Введем полярную систему координат относительно центра правого глаза и углом, откладываемым от оси абсцисс. Определим функции $\Phi_A: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ и $\Phi_B: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ следующим образом:

$$\Phi_A(x, y) = \begin{cases} \arcsin\left(\frac{y}{r}\right) & \left| y \geq 0 \wedge x + h \geq 0; \right. \\ \pi - \arcsin\left(\frac{y}{r}\right) & \left| y \geq 0 \wedge x + h < 0; \right. \\ \pi + \arcsin\left(\frac{|y|}{r}\right) & \left| y < 0 \wedge x + h < 0; \right. \\ 2\pi - \arcsin\left(\frac{|y|}{r}\right) & \left| y < 0 \wedge x + h \geq 0; \right. \end{cases} \quad (4)$$

$$\Phi_B(x, y) = \begin{cases} \arcsin\left(\frac{y}{r}\right) & \left| y \geq 0 \wedge x - h \geq 0; \right. \\ \pi - \arcsin\left(\frac{y}{r}\right) & \left| y \geq 0 \wedge x - h < 0; \right. \\ \pi + \arcsin\left(\frac{|y|}{r}\right) & \left| y < 0 \wedge x - h < 0; \right. \\ 2\pi - \arcsin\left(\frac{|y|}{r}\right) & \left| y < 0 \wedge x - h \geq 0. \right. \end{cases} \quad (5)$$

Тогда полярные координаты (r, φ) точки с координатами (x, y) для левого (с центром в точке A) и правого глаза (с центром в точке B) могут быть вычислены соответственно по следующим формулам:

$$r_A = \sqrt{(x+h)^2 + y^2}; \quad (6)$$

$$r_B = \sqrt{(x-h)^2 + y^2}; \quad (7)$$

$$\varphi_A = \Phi_A(x, y); \quad (8)$$

$$\varphi_B = \Phi_B(x, y). \quad (9)$$

Далее будем рассматривать модель только одного глаза, используя полярную систему координат относительно его центра с углом, откладываемым от оси абсцисс. Пронумеруем все омматидии числами от 0 до $m-1$ против часовой стрелки, как это показано на рис. 5 для глаза с центром в точке B . Областью видимости омматидия будем называть угол с вершиной в центре фасеточного глаза, границы которого совпадают с радиальными границами омматидия (см. рис. 6). Будем предполагать, что любая точка на границе между двумя омматидиями будет попадать в область видимости только омматидия с большим номером. Таким образом, мы предполагаем, что области видимости омматидиев одного глаза не пересекаются. Следующее утверждение позволяет определить омматидий, который «видит» точку с заданными координатами.

Утверждение 1. Пусть задана плоскостная модель фасеточного глаза с радиусом $l \in \mathbb{R}$ и количеством омматидиев $m \in \mathbb{N}$. Пусть омматидии пронумерованы числами от 0 до $m-1$ против часовой стрелки, как это показано на рис. 6. Введем полярные координаты относительно центра фасеточного глаза с углом, откладываемым от правой

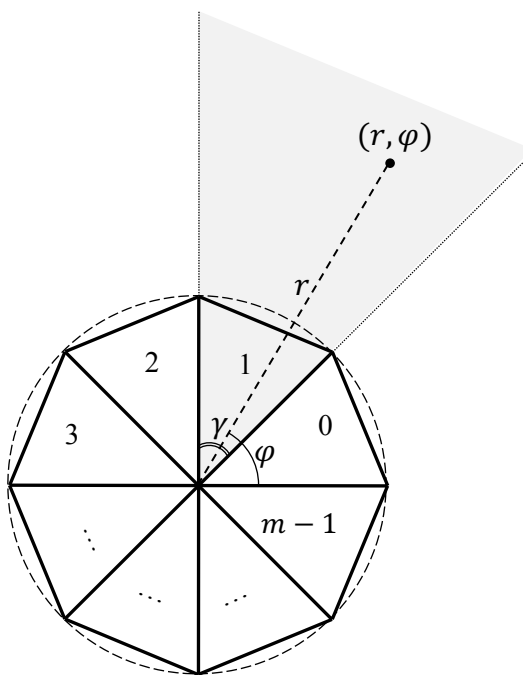


Рис. 6. Область видимости омматидия 1

(по часовой стрелке) грани нулевого омматидия. Тогда, номер омматидия $k \in \{0, 1, \dots, m-1\}$, в область видимости которого попадает точка с полярными координатами (r, φ) , определяется по формуле

$$k = \left\lfloor \frac{m}{2\pi} \varphi \right\rfloor. \tag{10}$$

Доказательство. Пусть k — номер омматидия, в область видимости которого попадает точка с полярными координатами (r, φ) . Тогда, с учетом формулы (2), угол φ , определяющий область видимости k -го омматидия, в полярных координатах задается системой неравенств

$$\begin{cases} \varphi \geq \frac{2\pi}{m} k; \\ \varphi < \frac{2\pi}{m} (k+1). \end{cases} \tag{11}$$

Перепишем эту систему в виде

$$\begin{cases} k \leq \frac{m}{2\pi} \varphi; \\ k > \frac{m}{2\pi} \varphi - 1. \end{cases} \tag{12}$$

Тогда, по определению наибольшего целого числа, не превышающего $\frac{m}{2\pi} \varphi$, из (12) по-

лучаем $k = \left\lfloor \frac{m}{2\pi} \varphi \right\rfloor$. Утверждение доказано.

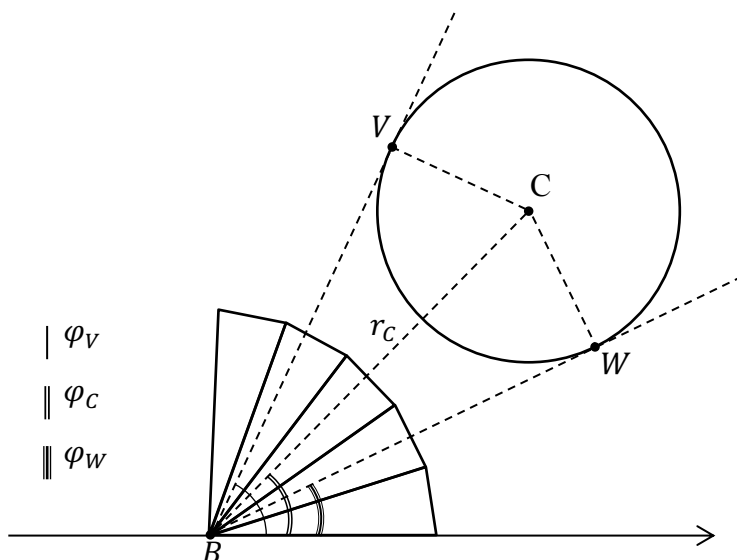


Рис. 7. Видимость объекта

Теперь рассмотрим в качестве видимого объекта плоскостной системы фасеточного зрения окружность радиуса g и центром в точке $C(x, y)$ в контексте рис. 5. Введем полярную систему координат относительно центра правого глаза и оси абсцисс. Тогда полярные координаты (r, φ) точки с координатами (x, y) для правого и левого глаз могут быть вычислены по формуле (8). Расстояние от центра соответствующего глаза до объекта $r - g$. Следующее утверждение позволяет определить омматидии, которые «видят» окружность с заданными радиусом и координатами центра.

Утверждение 2. Пусть выполняются условия утверждения 1. Тогда справедливо следующее утверждение: окружность с центром в точке $C = (r_c, \varphi_c)$ и радиусом g попадает в область видимости омматидия с номером k в том, и только в том случае, когда

$$\left| \frac{m}{2\pi} \left(\varphi_c - \arcsin \left(\frac{g}{r_c} \right) \right) \right| \leq k \leq \left| \frac{m}{2\pi} \left(\varphi_c + \arcsin \left(\frac{g}{r_c} \right) \right) \right|. \quad (13)$$

Доказательство. Проведем касательные к окружности, проходящие через центр фасеточного глаза B (совпадает с центром полярной системы координат). Пусть указанные касательные касаются окружности в точках $V = (r_V, \varphi_V)$ и $W = (r_W, \varphi_W)$, как это показано на рис. 7. Тогда

$$\varphi_V = \varphi_c + \arcsin \frac{g}{r_c}, \quad (14)$$

$$\varphi_W = \varphi_c - \arcsin \frac{g}{r_c}. \quad (15)$$

Отсюда утверждению 1 имеем

$$\left| \frac{m}{2\pi} \left(\varphi_c - \arccos \left(\frac{g}{r_c} \right) \right) \right| \leq k \leq \left| \frac{m}{2\pi} \left(\varphi_c + \arccos \left(\frac{g}{r_c} \right) \right) \right|,$$

что и требовалось доказать. Утверждение доказано.

4. Алгоритм порождения прецедентов

В данном разделе описывается алгоритм построения обучающей выборки для искусственной нейронной сети, работающей на основе модели двумерного фасеточного зрения. В качестве наблюдаемых объектов фигурируют круги различного радиуса, расположенные перед фасеточными глазами на различных расстояниях и под различными углами. Для генерации прецедентов используются углы от 0 до π , откладываемые против часовой стрелки между положительной полuosью (O, X) и вектором, соединяющим центр системы фасеточного зрения (рис. 5) с центром наблюдаемого круга.

Алгоритм построения обучающей выборки использует следующие параметры модели фасеточного зрения:

- m — количество омматидиев в каждом глазу;
- h — половина расстояния между фасеточными глазами;
- l — радиус описанной вокруг фасеточного глаза окружности.

Для генерации образцов алгоритм построения обучающей выборки использует следующие параметры:

- n — количество генерируемых образцов;
- g_{\min} — минимальный радиус наблюдаемого круга;
- g_{\max} — максимальный радиус наблюдаемого круга;
- r_{\min} — минимальное расстояние от центра системы фасеточного зрения до центра наблюдаемого круга;
- r_{\max} — максимальное расстояние от центра системы фасеточного зрения до центра наблюдаемого круга;
- φ_{\min} — минимальный азимут наблюдаемого круга;
- φ_{\max} — максимальный азимут наблюдаемого круга.

При генерации образцов алгоритм отбирает прецеденты, удовлетворяющие следующим условиям:

- (1) круг видим каждым глазом полностью (не перекрывается другим глазом);
- (2) круг не пересекается с областью обоих глаз;
- (3) никакая часть наблюдаемого круга не попадает в отрицательную область оси ординат;
- (4) круг видим для обоих глаз (количество омматидиев, в область видимости которых попадает образец, для каждого глаза больше единицы).

Свойство (4) проверяется непосредственно. Необходимое и достаточное условие для выполнения свойств (1)–(3) определяется следующим утверждением.

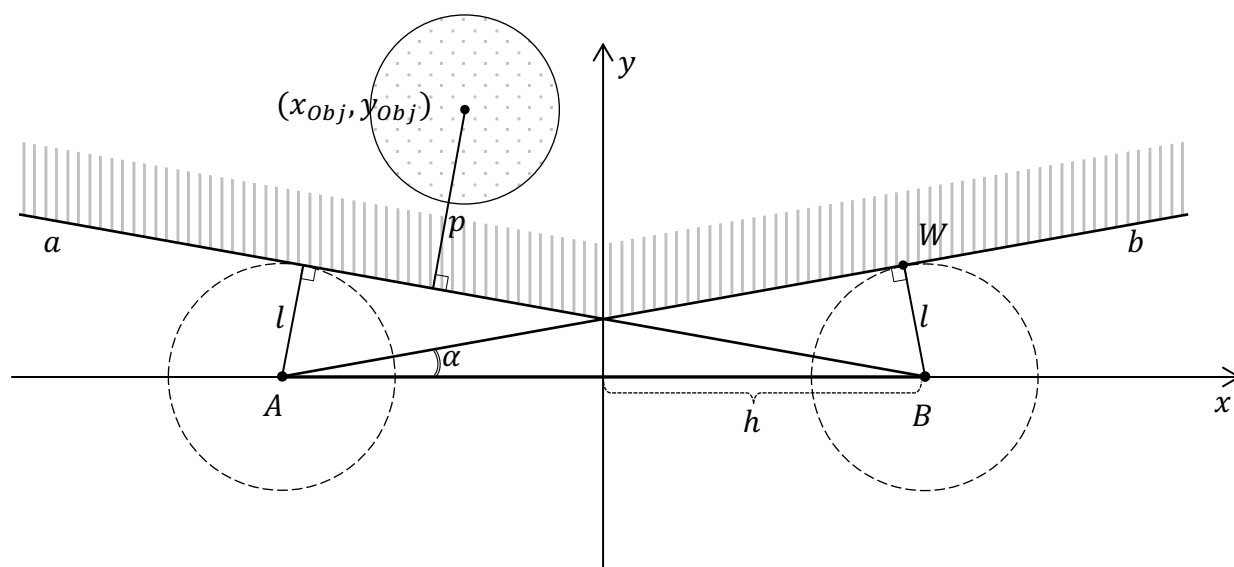


Рис. 8. Объект в плоскостной системе фасеточного зрения

Утверждение 3. Пусть задана система фасеточного зрения в декартовой системе координат (см. рис. 5) со следующими параметрами:

h — половина расстояния между фасеточными глазами;

l — радиус описанной окружности фасеточного глаза.

Для наблюдаемого круга S с центром в точке (x_{Obj}, y_{Obj}) и радиусом g_{Obj} условия (1)–(3) выполняются тогда и только тогда, когда точка (x_{Obj}, y_{Obj}) является допустимой для следующей системы неравенств:

$$\begin{cases} \sqrt{4h^2 - l^2} y_{Obj} + lx_{Obj} > lh; \\ \sqrt{4h^2 - l^2} y_{Obj} - lx_{Obj} > lh; \\ \left| -lx_{Obj} - \sqrt{4h^2 - l^2} y_{Obj} + lh \right| > 2hg_{Obj}; \\ \left| lx_{Obj} - \sqrt{4h^2 - l^2} y_{Obj} + lh \right| > 2hg_{Obj}. \end{cases} \quad (16)$$

Доказательство. Сначала докажем необходимость первых двух неравенств в системе (16). Доказательство выполним с помощью геометрических построений. Проведем касательную a к описанной окружности левого глаза через центр правого глаза $(h, 0)$ так, чтобы она пересекла положительную полуось ординат, и проведем касательную b к описанной окружности правого глаза через центр левого глаза $(-h, 0)$ так, чтобы она также пересекла положительную полуось ординат (см. рис. 8). Очевидно, что свойства (1)–(3) выполняются в том и только в том случае, когда видимый круг целиком находится выше обеих касательных в области, обозначенной вертикальной штриховкой.

Пусть прямая b касается описанной окружности правого глаза в точке W . Обозначим через α угол, образованный лучами $[A, W]$ и $[A, B]$. Тогда уравнения прямых a и b могут быть записаны в виде

$$y = -\operatorname{tg}(\alpha) \cdot (x - h) \quad (17)$$

и

$$y = \operatorname{tg}(\alpha) \cdot (x + h) \quad (18)$$

соответственно. Имеем

$$\operatorname{tg}(\alpha) = \frac{l}{\|W - A\|}. \quad (19)$$

Учитывая, что $\|W - A\| = \sqrt{4h^2 - l^2}$, из (19) получаем

$$\operatorname{tg}(\alpha) = \frac{l}{\sqrt{4h^2 - l^2}}. \quad (20)$$

Учитывая (1), видим, что подкоренное выражение в (20) принимает значение больше нуля. Подставив правую часть уравнения (20) в (17) и (18) получаем следующие уравнения прямых a и b :

$$y = -\frac{l(x-h)}{\sqrt{4h^2 - l^2}} \quad (21)$$

и

$$y = \frac{l(x+h)}{\sqrt{4h^2 - l^2}}. \quad (22)$$

Перепишем их в виде

$$\sqrt{4h^2 - l^2} y + lx = lh \quad (23)$$

и

$$\sqrt{4h^2 - l^2} y - lx = lh. \quad (24)$$

Таким образом, для того, чтобы точка лежала в заштрихованной области, не касаясь прямых a и b , необходимо, чтобы она удовлетворяла следующей системе неравенств:

$$\begin{cases} \sqrt{4h^2 - l^2} y + lx > lh; \\ \sqrt{4h^2 - l^2} y - lx > lh. \end{cases} \quad (25)$$

Правильность расстановки знаков неравенств легко проверяется путем подстановки координат точки $(0,0)$, не входящей в заштрихованную область. Очевидно, что центр (x_{Obj}, y_{Obj}) круга должен удовлетворять (25). Отсюда получаем первые два неравенства в (16).

Теперь докажем необходимость третьего и четвертого неравенств в системе (16). Расстояние p от центра (x_{Obj}, y_{Obj}) наблюдаемого круга до прямой a вычисляется по формуле

$$p = \frac{|-lx_{Obj} - \sqrt{4h^2 - l^2} y_{Obj} + lh|}{2h}. \quad (26)$$

Для того, чтобы наблюдаемый круг S радиуса g_{Obj} с центром в точке (x_{Obj}, y_{Obj}) не имел общих точек с прямой a , необходимо, чтобы выполнялось условие $p > g_{Obj}$, то есть

$$|-lx_{Obj} - \sqrt{4h^2 - l^2} y_{Obj} + lh| > 2hg_{Obj}. \quad (27)$$

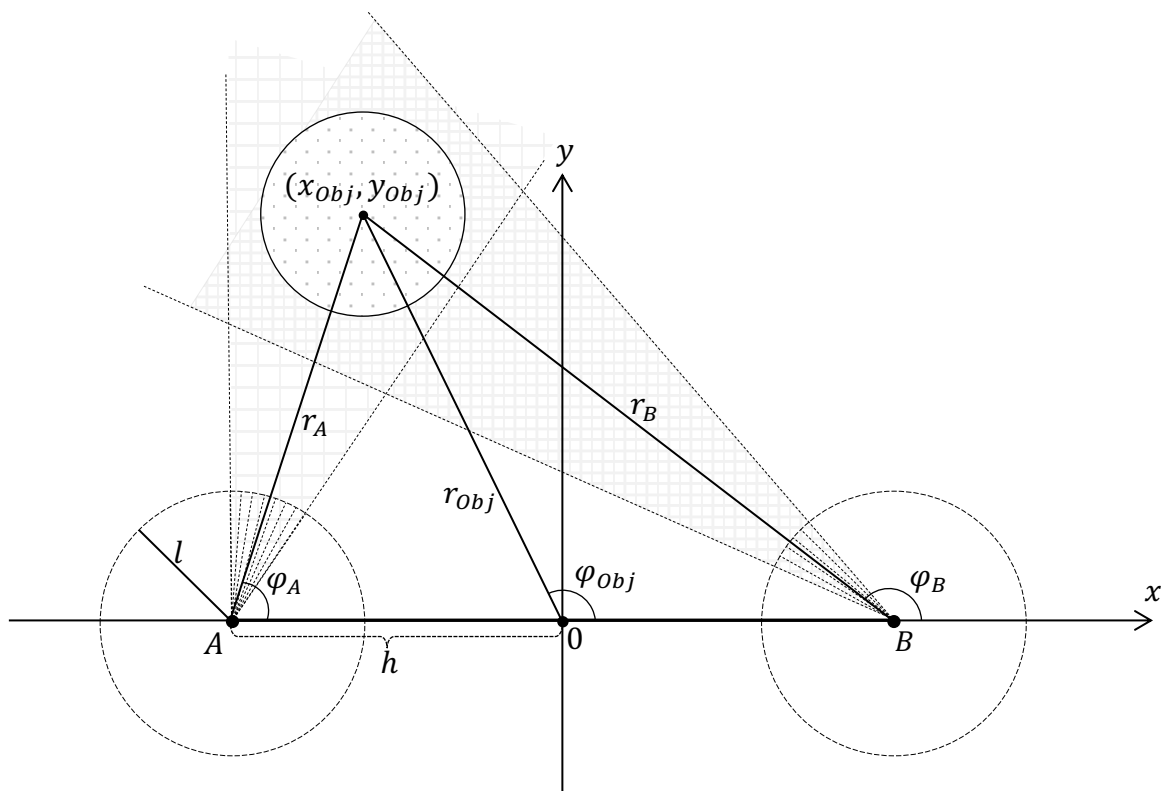


Рис. 9. Определение положения наблюдаемого объекта в плоскостной системе фасеточного зрения

Аналогичным образом получаем необходимое условие для того, чтобы наблюдаемый круг S радиуса g_{Obj} с центром в точке (x_{Obj}, y_{Obj}) не имел общих точек с прямой b :

$$\left| lx_{Obj} - \sqrt{4h^2 - l^2} y_{Obj} + lh \right| > 2hg_{Obj}. \quad (28)$$

Таким образом, получаем все неравенства системы (16) как необходимые условия выполнения свойств (1)–(3). Достаточность этих условий очевидна. Утверждение доказано.

Процедура генерации обучающей выборки представлена в виде алгоритма 1. Каждый прецедент генерируемой выборки представляет собой четверку (β_A, β_B, d, a) . Параметры β_A и β_B представляют собой битовые шкалы, кодирующие визуальные сигналы, поступающие на омматидии левого и правого глаз соответственно. Указанные битовые шкалы подаются на вход нейронной сети. Длина каждой шкалы равна m (количеству омматидиев в одном глазу). Каждый бит описывает состояние соответствующего омматидия фасеточного глаза: единица, если в поле зрения омматидия попадает какая-либо часть наблюдаемого круга; ноль в противном случае. Параметры d и a определяют истинное положение наблюдаемого круга в соответствии со следующими формулами:

$$d = \tanh \left(\lambda \frac{r_{Obj}}{h} \right); \quad (29)$$

$$a = \frac{1}{1 + e^{-\gamma\left(\frac{\pi}{2} - \varphi_{Obj}\right)}}. \quad (30)$$

Здесь $\lambda, \gamma \in (0; 1]$ — параметры алгоритма. Значения r_{Obj} и φ_{Obj} могут быть восстановлены из d и a следующим образом:

$$r_{Obj} = \frac{h}{\lambda} \operatorname{artanh}(d); \quad (31)$$

$$\varphi_{Obj} = \frac{\pi}{2} + \frac{1}{\gamma} \ln \frac{1-a}{a}. \quad (32)$$

Рассмотрим более подробно шаги алгоритма. На каждой итерации алгоритма генерируется один прецедент обучающей выборки. На шаге 1 вводятся параметры алгоритма:

- m — количество омматидиев в каждом глазу;
- h — половина расстояния между фасеточными глазами;
- l — радиус описанной вокруг фасеточного глаза окружности;
- n — количество генерируемых образцов;
- g_{\min} — минимальный радиус наблюдаемого круга;
- g_{\max} — максимальный радиус наблюдаемого круга;
- r_{\min} — минимальное расстояние от центра системы фасеточного зрения до центра наблюдаемого круга;
- r_{\max} — максимальное расстояние от центра системы фасеточного зрения до центра наблюдаемого круга;
- φ_{\min} — минимальный азимут наблюдаемого круга;
- φ_{\max} — максимальный азимут наблюдаемого круга.

На шаге 2 счетчику прецедентов i присваивается значение ноль. На шаге 3 множество прецедентов \mathcal{M} первоначально определяется как пустое. Шаг 4 начинает итерационный цикл. На этом шаге, используя генератор случайных чисел, формируется тройка чисел $(r_{Obj}, \varphi_{Obj}, g_{Obj})$, определяющая наблюдаемый круг с радиусом g_{Obj} , центр которого расположен на расстоянии r_{Obj} от центра системы фасеточного зрения и азимутом φ_{Obj} (см. рис. 9). На шаге 5 вычисляются декартовы координаты (x_{Obj}, y_{Obj}) центра наблюдаемого круга. Далее, на шаге 6, в соответствии с утверждением 3 проверяется допустимость сгенерированного наблюдаемого круга. При этом используется функция *valid*, проверяющая, что центр круга (x_{Obj}, y_{Obj}) с радиусом g_{Obj} удовлетворяет системе неравенств (16). Если сгенерированный наблюдаемый круг не является допустимым, происходит переход на шаг 4 для повторной генерации прецедента. На шагах 7, 8 по формулам (6)–(9) вычисляются полярные координаты (r_A, φ_A) и (r_B, φ_B) центра наблюдаемого круга относительно левого и правого глаз системы фасеточного зрения соответственно.

Алгоритм 1. Генерация обучающей выборки \mathcal{M}

```

1: input  $h, l, m, n, r_{min}, r_{max}, \varphi_{min}, \varphi_{max}, g_{min}, g_{max}$ 
2:  $i := 0$ 
3:  $\mathcal{M} := \emptyset$ 
4:  $r_{Obj} := rnd(r_{min}, r_{max}); \varphi_{Obj} := rnd(\varphi_{min}, \varphi_{max}); g_{Obj} := rnd(g_{min}, g_{max})$ 
5:  $x_{Obj} := r_{Obj} \cos(\varphi_{Obj}); y_{Obj} := r_{Obj} \sin(\varphi_{Obj})$ 
6: if not  $valid(x_{Obj}, y_{Obj}, g_{Obj})$  goto 4
7:  $r_A := \sqrt{(x_{Obj} + h)^2 + y_{Obj}^2}; \varphi_A := \Phi_A(x_{Obj}, y_{Obj})$ 
8:  $r_B := \sqrt{(x_{Obj} - h)^2 + y_{Obj}^2}; \varphi_B := \Phi_B(x_{Obj}, y_{Obj})$ 
9:  $L_A := \left\lfloor \frac{m}{2\pi} \left( \varphi_A - \arcsin\left(\frac{g_{Obj}}{r_A}\right) \right) \right\rfloor; R_A := \left\lceil \frac{m}{2\pi} \left( \varphi_A + \arcsin\left(\frac{g_{Obj}}{r_A}\right) \right) \right\rceil$ 
10: if  $L_A = R_A$  goto 4
11: for  $j \in [L_A, R_A]$  do  $\beta_A[j] := 1$ 
12: for  $j \notin [L_A, R_A]$  do  $\beta_A[j] := 0$ 
13:  $L_B := \left\lfloor \frac{m}{2\pi} \left( \varphi_B - \arcsin\left(\frac{g_{Obj}}{r_B}\right) \right) \right\rfloor; R_B := \left\lceil \frac{m}{2\pi} \left( \varphi_B + \arcsin\left(\frac{g_{Obj}}{r_B}\right) \right) \right\rceil$ 
14: for  $j \in [L_B, R_B]$  do  $\beta_B[j] := 1$ 
15: for  $j \notin [L_B, R_B]$  do  $\beta_B[j] := 0$ 
16:  $d = \tanh\left(\lambda \frac{r_{Obj}}{h}\right)$ 
17:  $a = \frac{1}{1 + e^{-\gamma\left(\frac{\pi}{2} - \varphi_{Obj}\right)}}$ 
18:  $\mathcal{M} := \mathcal{M} \cup \{(\beta_A, \beta_B, d, a)\}$ 
19:  $i := i + 1$ 
20: if  $i \leq n$  goto 4
21: stop

```

На шаге 9, в соответствии с утверждением 2, определяются первый и последний номера омматидиев левого глаза, в область видимости которых попадает наблюдаемый круг. На шаге 10 проверяется, является ли наблюдаемый круг допустимым в соответствии с условием (4). На шагах 11 и 12 формируется битовая шкала β_A для левого глаза. На шагах 13–15 формируется битовая шкала β_B для правого глаза. На шагах 16 и 17 вычисляются параметры d и a , определяющие истинное положение наблюдаемого круга. На шаге 18, полученный прецедент вида (β_A, β_B, d, a) помещается в множество прецедентов обучающей выборки. На шаге 19 счетчик прецедентов увеличивается на единицу. На шаге 20 оценивается количество сгенерированных образцов: если количество прецедентов в обучающей выборке меньше n , итерационный процесс продолжается с шага 4, в противном случае итерационный процесс останавливается.

Заключение

В статье рассмотрены три типа фасеточных глаз насекомых: аппозиционные, оптико-суперпозиционные и нейросуперпозиционные. В качестве биологического прототипа для построения математической модели был выбран аппозиционный фасеточный глаз, поскольку, в силу своих анатомических особенностей, данный тип является наиболее простым с точки зрения технической реализации. В работе приведена классификация и краткий обзор искусственных фасеточных глаз, которые, в настоящее время, могут успешно применяться для создания автономных робототехнических устройств различного назначения. В рамках данного исследования построена математическая модель бинокулярного фасеточного зрения в двумерном пространстве. Предложенная модель позволяет формулировать задачи навигации и детектирования объекта для систем визуализации фасеточного типа на плоскости. В рамках предложенной модели выведены формулы для расчета координат наблюдаемого объекта, формула вычисления номера омматидия, в область видимости которого попадает окружность с заданными координатами или ее часть. Доказано утверждение о необходимом и достаточном условии попадания заданной окружности в область видимости омматидия с заданным номером. На основе модели двумерного фасеточного зрения сформулирован алгоритм порождения прецедентов обучающей выборки для искусственной нейронной сети.

В рамках дальнейших исследований мы планируем:

- 1) на основе представленной модели создать искусственную нейронную сеть, способную на плоскости определять азимут и расстояние до наблюдаемого объекта;
- 2) обобщить предложенную модель на трехмерный случай.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-07-00092-а и Министерства образования и науки РФ (государственное задание FENU-2020-0022).

Литература

1. Pérez L., Rodríguez Í., Rodríguez N., et al. Robot guidance using machine vision techniques in industrial environments: A comparative review // *Sensors*. 2016. Vol. 16, no. 3. P. 335. DOI: 10.3390/s16030335.
2. Azim A., Aycard O. Detection, classification and tracking of moving objects in a 3D environment // *IEEE Intelligent Vehicles Symposium, Proceedings*. 2012. P. 802–807. DOI: 10.1109/IVS.2012.6232303.
3. Llamazares Á., Molinos E.J., Ocaña M. Detection and Tracking of Moving Obstacles (DATMO): A Review // *Robotica*. 2020. Vol. 38, no. 5. P. 761–774. DOI: 10.1017/S0263574719001024.
4. Sitte J., Winzer P. Methodic design of robot vision systems // *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, ICMA 2007*. 2007. P. 1758–1763. DOI: 10.1109/ICMA.2007.4303816.
5. Соломатин В. Фасеточное зрение: перспективы в оптико-электронных системах // *Фототоника*. 2009. № 1. P. 22–26.
6. *Encyclopedia of Insects*. Academic Press, 2003, 1295 p.
7. Захваткин Ю.А. Курс общей энтомологии. М.: Колос, 2001. 376 с.

8. Тыщенко В.П. Физиология насекомых: учеб. пособие для студентов ун-тов, обучающихся по спец. «Биология». 1986. 303 с.
9. Hemmi J.M., Zeil J. Robust judgement of inter-object distance by an arthropod // *Nature*. 2003. Vol. 421, no. 6919. P. 160–163. DOI: 10.1038/nature01247.
10. Olberg R., Seaman R., Coats M., et al. Eye movements and target fixation during dragonfly prey-interception flights // *J. Comp. Physiol. A Neuroethol. Sensory, Neural, Behav. Physiol.* 2007. Vol. 193, no. 7. P. 685–693. DOI: 10.1007/s00359-007-0223-0.
11. Qu P., Chen F., Liu H., et al. A simple route to fabricate artificial compound eye structures // *Opt. Express*. 2012. Vol. 20, no. 5. P. 5775. DOI: 10.1364/oe.20.005775.
12. Shi C., Wang Y., Liu C., et al. SCECam: a spherical compound eye camera for fast location and recognition of objects at a large field of view // *Opt. Express*. 2017. Vol. 25, no. 26. P. 32333–32345. DOI: 10.1364/OE.25.032333.
13. Мазохин-Поршняков Г.А., Елизаров Ю.А., Жантиев Р.Д., и др. Руководство по физиологии органов чувств насекомых. М.: Изд-во МГУ, 1983. 262 с.
14. Borst A. Drosophila's View on Insect Vision // *Curr. Biol.* 2009. Vol. 19, no. 1. P. R36–R47. DOI: 10.1016/j.cub.2008.11.001.
15. Cheng Y., Cao J., Zhang Y., et al. Review of state-of-the-art artificial compound eye imaging systems // *Bioinspiration and Biomimetics*. 2019. Vol. 14, no. 3. DOI: 10.1088/1748-3190/aaffb5.
16. Tanida J., Mima H., Kagawa K., et al. Application of a compound imaging system to odontotherapy // *Opt. Rev.* 2015. Vol. 22, no. 2. P. 322–328. DOI: 10.1007/s10043-015-0052-2.
17. Li L., Yi A.Y. Design and fabrication of a freeform microlens array for a compact large-field-of-view compound-eye camera // *Appl. Opt.* 2012. Vol. 51, no. 12. P. 1843–1852. DOI: 10.1364/AO.51.001843.
18. Deng H., Gao X., Ma M., et al. Catadioptric planar compound eye with large field of view // *Opt. Express*. 2018. Vol. 26, no. 10. P. 12455. DOI: 10.1364/oe.26.012455.
19. Duparré J., Wippermann F., Dannberg P., et al. Artificial compound eye zoom camera // *Bioinspir Biomim.* 2008. Vol. 3, no. 4. DOI: 10.1088/1748-3182/3/4/046008.
20. Horisaki R., Kagawa K., Nakao Y., et al. Irregular lens arrangement design to improve imaging performance of compound-eye imaging systems // *Appl. Phys. Express*. 2010. Vol. 3, no. 2. P. 022501. DOI: 10.1143/APEX.3.022501.
21. Li F., Chen S., Luo H., et al. Curved micro lens array for bionic compound eye // *Optik*. 2013. Vol. 124, no. 12. P. 1346–1349. DOI: 10.1016/j.ijleo.2012.03.063.
22. Viollet S., Godiot S., Leitel R., et al. Hardware architecture and cutting-edge assembly process of a tiny curved compound eye // *Sensors*. 2014. Vol. 14, no. 11. P. 21702–21721. DOI: 10.3390/s141121702.

Федянина Раиса Сулеймановна, ст. преподаватель, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Соколинский Леонид Борисович, д.ф.-м.н., профессор, проректор по информатизации, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

TWO-DIMENSIONAL MODEL OF FACET VISION

© 2020 R.S. Fedyanina, L.B. Sokolinsky

*South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)**E-mail: raisa.fedianina@susu.ru, leonid.sokolinsky@susu.ru*

Received: 26.10.2020

One of the modern promising areas of research in the field of machine vision is visualization systems based on artificial compound eyes, which are based on the principles of organizing insect vision. Such eyes have a complex structure, the main elements of which are visual elements called ommatidium. In this paper, we construct a mathematical model of binocular facet vision in two-dimensional space. This model allows you to describe the tasks of navigating a robotic device and detecting objects on a plane. In the proposed model, the faceted eye is a regular polygon with a 360° view. The main parameters of the model are the number of facets (ommatidia) in one eye, the radius of the circle described around the eye, and the distance between the eyes. The model assumes that the ommatidia fields of view of one eye do not intersect. Circles of various diameters appear as observed objects. We derive the formula for determining the number of ommatidium in which the point with the specified coordinates falls. We prove a theorem on the necessary and sufficient condition for a given circle to fall into the field of view of an ommatidium with a given number. An algorithm for constructing a training sample for an artificial neural network based on a two-dimensional facet vision model is proposed.

Keywords: visualization systems, mathematical model of facet vision, neural network, training sample generation.

FOR CITATION

Fedyanina R.S., Sokolinsky L.B. Two-Dimensional Model of Facet Vision. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2020. Vol. 9, no. 4. P. 48–66. (in Russian) DOI: 10.14529/cmse200404.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Pérez L., Rodríguez Í., Rodríguez N., et al. Robot guidance using machine vision techniques in industrial environments: A comparative review. *Sensors*. 2016. Vol. 16, no. 3. P. 335. DOI: 10.3390/s16030335.
2. Azim A., Aycard O. Detection, classification and tracking of moving objects in a 3D environment. *IEEE Intelligent Vehicles Symposium, Proceedings*. 2012. P. 802–807. DOI: 10.1109/IVS.2012.6232303.
3. Llamazares Á., Molinos E.J., Ocaña M. Detection and Tracking of Moving Obstacles (DATMO): A Review. *Robotica*. 2020. Vol. 38, no. 5. P. 761–774. DOI: 10.1017/S0263574719001024.
4. Sitte J., Winzer P. Methodic design of robot vision systems. *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, ICMA 2007*. 2007. P. 1758–1763. DOI: 10.1109/ICMA.2007.4303816.
5. Solomatin V. Facet vision: the prospects of opto-electronic systems. *Photonics*. 2009. Vol. 1. P. 22–26. (in Russian)
6. *Encyclopedia of Insects*. Academic Press, 2003. 1295 p.
7. Zakhvatkin Y.A. *Course of General entomology*. M.: Kolos, 2001. 376 p. (in Russian)

8. Tyshchenko V.P. Physiology of insect: studies manual for students of uni-tov, studying in spec. "Biology". 1986. 303 p. (in Russian)
9. Hemmi J.M., Zeil J. Robust judgement of inter-object distance by an arthropod. *Nature*. 2003. Vol. 421, no. 6919. P. 160–163. DOI: 10.1038/nature01247.
10. Olberg R., Seaman R., Coats M., et al. Eye movements and target fixation during dragonfly prey-interception flights. *J. Comp. Physiol. A Neuroethol. Sensory, Neural, Behav. Physiol.* 2007. Vol. 193, no. 7. P. 685–693. DOI: 10.1007/s00359-007-0223-0.
11. Qu P., Chen F., Liu H., et al. A simple route to fabricate artificial compound eye structures. *Opt. Express*. 2012. Vol. 20, no. 5. P. 5775. DOI: 10.1364/oe.20.005775.
12. Shi C., Wang Y., Liu C., et al. SCECam: a spherical compound eye camera for fast location and recognition of objects at a large field of view. *Opt. Express*. 2017. Vol. 25, no. 26. P. 32333–32345. DOI: 10.1364/OE.25.032333.
13. Mazokhin-Porshnyakov G.A., Yelizarov Yu.A., Zhantiev R.D. et al. Guide to the physiology of the insect sense organs. M.: MSU publishing House, 1983. 262 p. (in Russian)
14. Borst A. Drosophila's View on Insect Vision. *Curr. Biol.* 2009. Vol. 19, no. 1. P. R36–R47. DOI: 10.1016/j.cub.2008.11.001.
15. Cheng Y., Cao J., Zhang Y., et al. Review of state-of-the-art artificial compound eye imaging systems. *Bioinspiration and Biomimetics*. 2019. Vol. 14, no. 3. DOI: 10.1088/1748-3190/aaffb5.
16. Tanida J., Mima H., Kagawa K., et al. Application of a compound imaging system to odontotherapy. *Opt. Rev.* 2015. Vol. 22, no. 2. P. 322–328. DOI: 10.1007/s10043-015-0052-2.
17. Li L., Yi A.Y. Design and fabrication of a freeform microlens array for a compact large-field-of-view compound-eye camera. *Appl. Opt.* 2012. Vol. 51, no. 12. P. 1843–1852. DOI: 10.1364/AO.51.001843.
18. Deng H., Gao X., Ma M., et al. Catadioptric planar compound eye with large field of view. *Opt. Express*. 2018. Vol. 26, no. 10. P. 12455. DOI: 10.1364/oe.26.012455.
19. Duparré J., Wippermann F., Dannberg P., et al. Artificial compound eye zoom camera. *Bioinspir Biomim.* 2008. Vol. 3, no. 4. DOI: 10.1088/1748-3182/3/4/046008.
20. Horisaki R., Kagawa K., Nakao Y., et al. Irregular lens arrangement design to improve imaging performance of compound-eye imaging systems. *Appl. Phys. Express*. 2010. Vol. 3, no. 2. P. 022501. DOI: 10.1143/APEX.3.022501.
21. Li F., Chen S., Luo H., et al. Curved micro lens array for bionic compound eye. *Optik*. 2013. Vol. 124, no. 12. P. 1346–1349. DOI: 10.1016/j.ijleo.2012.03.063.
22. Viollet S., Godiot S., Leitel R., et al. Hardware architecture and cutting-edge assembly process of a tiny curved compound eye. *Sensors*. 2014. Vol. 14, no. 11. P. 21702–21721. DOI: 10.3390/s141121702.

ЧИСЛЕННОЕ ИССЛЕДОВАНИЕ ОСЕСИММЕТРИЧНЫХ СТРУЙНЫХ ТЕЧЕНИЙ НА ОСНОВЕ ТУРБУЛЕНТНОЙ МОДЕЛИ ν_t -92

© 2020 М.Э. Мадалиев

Институт механики и сейсмостойкости сооружений АН РУз

(100125 Ташкент, ул. Дурман йули, д. 51)

E-mail: madaliev.me2019@mail.ru

Поступила в редакцию: 06.04.2020

В работе проводится верификация турбулентной модели Секундова ν_t -92 для различных струй. На основе этой модели проведены численные исследования осесимметричной дозвуковой холодной, горячей и околосзвуковой струи. Для численной реализации уравнения гидродинамики записаны в переменных Мизеса и использована неявная абсолютно устойчивая схема второго порядка точности в поперечном и первого порядка в продольном направлениях. Для сравнения полученных численных результатов использованы опытные данные из базы данных NASA, где представлены не только данные экспериментов последних лет, но сравнительный анализ многих моделей турбулентности. Сравнения проведены по распространению осевой скорости потока, по профилю турбулентных напряжений и продольной скорости в различных сечениях. Показано, что модель количественно хорошо описывает основные параметры несжимаемой и сжимаемой турбулентных струй. Было выявлено, что модель Секундова ν_t -92 менее подвержена к так называемой «аномалии» круглой струи, суть которой заключается в том, что многие другие RANS модели дают очень сильное расширение для осесимметричных струй.

Ключевые слова: осредненные по Рейнольдсу уравнения Навье—Стокса, модель Секундова ν_t -92, прогонка, функция тока, турбулентное напряжение.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Мадалиев М.Э. Численное исследование осесимметричных струйных течений на основе турбулентной модели ν_t -92 // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 67–78. DOI: 10.14529/cmse200405.

Введение

Во многих случаях при движении жидкости и газа возникают так называемые поверхности тангенциального разрыва; течение жидкости в окрестности такой поверхности называется струей. В зависимости от относительного направления движения струи могут быть спутными или встречными. Тангенциальный разрыв терпят такие параметры, как скорость течения, температура, концентрация примеси, распределение же статического давления оказывается непрерывным. Как известно, на поверхности тангенциального разрыва в связи с ее неустойчивостью возникают вихри, беспорядочно движущиеся вдоль и поперек потока, вследствие этого между соседними струями происходит обмен конечными массами (молями), т.е. поперечный перенос количества движения, тепла и примесей. В результате на границе двух струй формируется область конечной толщины с непрерывным распределением скорости, температуры и концентрации примеси, эта область называется струйным турбулентным пограничным слоем. Наиболее изученным видом турбулентной струи является струя, распространяющаяся в покоящейся среде, такая струя называется затопленной. При равномерном поле скорости в начальном сечении затопленной струи границы ее пограничного слоя представляют собой расходящиеся поверхности, которые пересекаются у кромки сопла. С внешней стороны пограничный слой струи соприкасается с неподвижной жидкостью, причем под внешней границей понимают поверхность, во всех точках которой составляющая скоро-

сти по оси x равна нулю ($U = 0$). С внутренней стороны пограничный слой переходит в ядро постоянной скорости, поэтому на внутренней границе пограничного слоя скорость потока равна скорости истечения $U = U_0$ [1].

Рассматриваемая задача имеет большое значение для авиационной и ракетно-космической техники. Поэтому к исследованию различных струй, начиная с работ Прандтля за последние 100 лет посвящены множества работ [2, 3]. Несмотря на множества этих исследований, рассматриваемую задачу нельзя рассматривать как полностью завершенной, т.к. она связана до сих пор не решенной проблемой турбулентности. Несмотря на то, что до сегодняшнего дня разработано более 100 различных моделей турбулентности, универсальную модель турбулентности все еще не существует. Это означает, что эти модели если удовлетворительно описывают одних классов турбулентности, других могут описывать неверно даже качественно.

Струи могут быть различных типов, холодная дозвуковая, горячая дозвуковая, околозвуковая, сверхзвуковая струи. Определить модель турбулентности, которая описывает все эти струи с приемлемой точностью имеет большое практическое значение. Поэтому целью настоящей статьи является верификация однопараметрической модели турбулентности Секундова ν_t -92 для дозвуковых холодной и горячей, а также околозвуковой затопленных струй. Для определения точности используемой модели турбулентности численные результаты сравниваются с экспериментальными данными из базы данных NASA [4]. Данные из этой базы используются по той причине, что там представлены опытные результаты для ближнего и переходного участков струи, которые являются наиболее сложными тестами для турбулентных моделей. Потому, что в этих участках струя состоит из невозмущенной ламинарной и турбулентной зон. Кроме этого, в базе данных NASA доступны экспериментальные результаты для холодной и горячей, а также околозвуковой струй.

Статья организована следующим образом. Раздел 1 посвящен краткому обзору некоторых современных и эффективных моделей турбулентности. В разделе 2 дается основные моменты однопараметрической модели Секундова ν_t -92. В разделе 3 описаны методика численной реализации математической модели и начальные условия. В разделе 4 показаны сравнения численных результатов с экспериментальным данным. В заключении содержатся выводы и направления дальнейших исследований.

1. Обзор моделей турбулентности

Турбулентность является последней нерешенной проблемой классической физики. На сегодняшний день предложены множества различных теорий и подходов к турбулентности. Подходы к турбулентности можно разбить на три группы. В первую группу входят так называемые методы прямого моделирования турбулентности (DNS). В основе данного подхода лежит гипотеза о применимости уравнений Навье—Стокса для описания турбулентности. Поэтому в данном подходе задачи рассматриваются в трехмерной постановке и область течения разбивается на расчетные сетки размерами менее Колмагоровского масштаба. Далее для улавливания турбулентных пульсаций интегрирование по времени проводится очень мелкими шагами. Поэтому данный подход требует очень больших вычислительных ресурсов. По этой причине данным методом получены результаты лишь для простых турбулентных течений для небольшого числа Рейнольдса. Следующую группу подходов составляют методы больших вихрей (LES). Однако и этот подход требует суперкомпьютеров. Поэтому при сохранении современного темпа развития компьютеров первые два подхода могут быть применены для решения практических задач лишь в конце нынешнего столетия. Следующим, широко распространенным подходом

является так называемый подход Рейнольдса. Данный подход основывается на двух гипотезах: 1) скорость турбулентного потока состоит из осредненной и флуктуирующей скоростей; 2) уместность применения уравнения Навье—Стокса для описания турбулентности. Однако при данном подходе получается незамкнутая система уравнений Навье—Стокса осредненная по Рейнольдсу. Поэтому существуют различные модели турбулентности которые направлены на замыкания этой системы. Эти модели в литературе называются RANS моделями. В основе этих моделей лежат различные гипотезы и предположения, которые позволяют получить замкнутую систему уравнений. Одним из существенных предположений является то, что симметричные задачи можно рассматривать в двумерной постановке, несмотря на сугубо трехмерность турбулентных течений. Данное предположение существенно облегчает задачу. Поэтому для инженерных задач широко используются в основном модели RANS.

В базе данных NASA [4] приведены экспериментальные данные для различных задач по турбулентности. Кроме этого приводятся численные результаты различных полумпирических моделей RANS, а также их сравнительный анализ. Из этого анализа можно заключить, что наиболее высоко рейтинговой модели являются модель Спаларта и Аллмараса SA [5], и модель Ментера SST [6]. Однако анализ результатов этих моделей показывает, что эти модели для осесимметричной струи дают более сильное расширение струи, чем показывает опыт. Поэтому убывание осевой скорости от расстояния до сопла по этим моделям происходит быстрее чем по закону $1/x$ в автотомельной зоне. Данное явление присуще практически всем полумпирическим моделям RANS и называется «аномалией» круглой струи.

Недавно в публикации появилась работа [7], где предложен иной подход к проблеме турбулентности. В указанной работе получена математическая модель турбулентности на основе нового подхода, которая хорошо описывает струйные течения и не страдает «аномалией» круглой струи. Еще одним достоинством новой математической модели то, что она проста для решения инженерных задач. Однако вышеуказанная модель Секундова ν_t-92 уже широко используется практически во всех пакетах программ для расчета турбулентных течений. Поэтому в работе проводится проверка модели Секундова (ν_t-92) для задач турбулентной осесимметричной холодной, горячей и околосзвуковой струи.

2. Математическая модель ν_t-92 для турбулентной струи

В работе в качестве математической модели использована система уравнений Навье—Стокса, осредненных по Рейнольдсу (RANS). Для поиска рейнольдсовых напряжений используется гипотеза Буссинеска, которая связывает эти напряжения с деформацией осредненной скорости линейным образом. Тогда уравнения стационарного переноса импульса и температуры в цилиндрической системе координат имеют вид:

$$\begin{cases} \frac{\partial U}{\partial x} + \frac{\partial rV}{r\partial r} = 0, \\ \rho \left(U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial r} \right) = \frac{\partial}{\partial r} \left(r(\nu + \nu_t) \frac{\partial U}{\partial r} \right), \\ \rho \left(U \frac{\partial T}{\partial x} + V \frac{\partial T}{\partial r} \right) = \frac{\partial}{\partial r} \left(r \left(\frac{\nu}{Pr} + \frac{\nu_t}{Pr_t} \right) \frac{\partial T}{\partial r} \right). \end{cases} \quad (1)$$

Здесь ν , ν_t — молекулярная и турбулентная вязкости, Pr , Pr_t — молекулярное и турбулентное числа Прандтля. Начальные и граничные условия для системы уравнений (1) для затопленной струи ставятся стандартным образом и подробно изложены в [8].

Модель Секундова ν_t-92 . Эта модель относится к классу однопараметрических моделей турбулентности. Здесь появляется только одно дополнительное уравнение для рас-

чета кинематического коэффициента вихревой вязкости. Это низкорейнольдсовая модель, которая описывает всю область течения, включая пристеночные слои. Эта модель была усовершенствована за прошедшие годы, включая модель под названием v_t -90. Ссылка для модели с одним уравнением v_t -92 [9]:

$$\frac{\partial(\rho U_j v_t)}{\partial x_j} = \rho(P_v - D_v) + \frac{\partial}{\partial x_j} \left[\rho(v - C_0 v_t) \frac{\partial v_t}{\partial x_j} \right] + \frac{\partial}{\partial x_j} \left[\rho(-v + (C_1 - C_0) v_t) \frac{\partial v_t}{\partial x_j} \right], \quad (2)$$

где

$$P_v = \rho C_2 A_2 (v_t \Gamma_1 + A_1 v_t^{4/3} \Gamma_2^{2/3}) + \rho C_2 F_2 A_2 N_1 \sqrt{(v + v_t) \Gamma_1} + \rho C_3 v_t \left(\frac{\partial^2 v_t}{\partial x_j \partial x_j} + N_2 \right),$$

$$D_v = \rho C_5 v_t^2 \Gamma_1^2 / a^2 + \rho C_4 v_t \left(\frac{\partial \langle U_j \rangle}{\partial x_j} + \left| \frac{\partial \langle U_j \rangle}{\partial x_j} \right| \right) + \rho [C_6 v_t (N_1 d_w + v_{t,w}) + C_7 F_1 v v_t] / d^2,$$

здесь a — скорость звука, а угловые скобки $\langle \rangle$ представляют собой среднее по времени. Турбулентная вихревая вязкость: $\mu_t = \rho v_t$.

Другие термины, встречающиеся в приведенных выше уравнениях, определяются как:

$$F_1 = \frac{N_1 d_w + 0,4 C_8 v}{v_t + C_8 v + v_{t,w}}, F_2 = \frac{\chi^2 + 1,3\chi + 0,2}{\chi^2 - 1,3\chi + 1,0}, \chi = \frac{v_t}{7v}, \Gamma_1 = \sqrt{\frac{\partial U_i}{\partial x_j} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)},$$

$$\Gamma_2 = \sqrt{\sum_i \left(\frac{\partial^2 U_i}{\partial x_j \partial x_j} \right)^2} = \sqrt{\left(\frac{\partial^2 U_i}{\partial x_n \partial x_n} \right) + \left(\frac{\partial^2 U_i}{\partial x_m \partial x_m} \right)}, N_1 = \sqrt{\frac{\partial v_t}{\partial x_j} \frac{\partial v_t}{\partial x_j}}, N_2 = \sqrt{\frac{\partial N_1}{\partial x_j} \frac{\partial N_1}{\partial x_j}},$$

здесь d_w — расстояние до ближайшей стены, $d = d_w + 0,01 k_s$;

k_s является высотой шкалы шероховатости Никурадзе (0 для гладких стен).

Константы: $A_1 = -0,5$, $A_2 = 4,0$, $C_0 = 0,8$, $C_1 = 1,6$, $C_2 = 0,1$, $C_3 = 4,0$, $C_4 = 0,35$, $C_5 = 3,5$, $C_6 = 2,9$, $C_7 = 31,5$, $C_8 = 0,1$.

3. Метод решения

Для численной реализации систем уравнений (1) и (2) введем обобщенную функцию тока ψ , для которой справедливы соотношения:

$$U = \frac{\psi \partial \psi}{r \partial r}, V = -\frac{\psi \partial \psi}{r \partial x}.$$

Тогда уравнение неразрывности системы (1) будет удовлетворяться автоматически.

Сделаем замену переменных и перейдем от переменных (x, r) к переменным Мизеса [10] (ξ, ψ) , где $\xi = x$.

В новых переменных, производные определяются по формуле:

$$\frac{\partial}{\partial x} = \frac{\partial \xi}{\partial x} \frac{\partial}{\partial \xi} + \frac{\partial \psi}{\partial x} \frac{\partial}{\partial \psi},$$

$$\frac{\partial}{\partial r} = \frac{\partial \xi}{\partial r} \frac{\partial}{\partial \xi} + \frac{\partial \psi}{\partial r} \frac{\partial}{\partial \psi}.$$

В новых переменных система уравнений (1) приобретает вид

$$\left\{ \begin{array}{l} \frac{\partial U}{\partial \xi} = \frac{\partial}{\psi \partial \psi} \left((v + v_t) \frac{U \partial U}{\psi \partial \psi} \right), \\ \frac{\partial T}{\partial \xi} = \frac{\partial}{\psi \partial \psi} \left(\frac{U \partial T}{\psi \partial \psi} \left(\frac{v}{Pr} + \frac{v_t}{Pr_t} \right) \right), \\ \frac{\partial v_t}{\partial \xi} = (P_v - D_v) + \frac{\partial}{\psi \partial \psi} \left[(v + C_0 v_t) \frac{U \partial v_t}{\psi \partial \psi} \right] + \\ + \frac{\partial}{\psi \partial \psi} \left[(-v + (C_1 - C_0) v_t) \frac{U \partial v}{\psi \partial \psi} \right]. \end{array} \right. \quad (3)$$

Таким образом, новые переменные позволяют привести все уравнения системы к параболическому виду и данную систему можно записать в векторном виде

$$\frac{\partial \Phi}{\partial \xi} = \frac{\partial}{\psi \partial \psi} \left(q \frac{U \partial \Phi}{\psi \partial \psi} \right) + w. \quad (4)$$

В этом уравнении:

$$\Phi = \begin{bmatrix} U \\ T \\ v_t \end{bmatrix} \quad q = \begin{bmatrix} v + v_t \\ \frac{v}{Pr} + \frac{v_t}{Pr_t} \\ v + C_0 v_t \end{bmatrix} \quad w = \begin{bmatrix} 0 \\ 0 \\ (P_v - D_v) + \frac{\partial}{\psi \partial \psi} \left[(-v + (C_1 - C_0) v_t) \frac{U \partial v_t}{\psi \partial \psi} \right] \end{bmatrix}.$$

Для численной реализации уравнения (4) была использована неявная схема

$$\frac{\Phi_{i,j}^{n+1} - \Phi_{i,j}^n}{\Delta \xi} = \frac{0,5}{\psi_j \Delta \psi^2} \left[(q_{j+1} + q_j) (\Phi_{i,j+1}^{n+1} - \Phi_{i,j}^{n+1}) - (q_j + q_{j-1}) (\Phi_{i,j}^{n+1} - \Phi_{i,j-1}^{n+1}) \right] + w_j.$$

Это схема имеет точность второго порядка по направлению ψ , т.е. $O(\Delta \xi, \Delta \psi^2)$ [11].

Для численной реализации системы (3) уравнения приведены к безразмерному виду.

Для этого скорости и температура соотнесены к скорости и температуре на срезе сопла, а все длины к радиусу сопла. Начальные условия для этой системы были следующие: на срезе сопла при $0 < \psi < 1$ безразмерные скорости и температуры $\bar{U} = \bar{T} = 1$ и $\bar{U} = \bar{T} = 0$ для окружающей среды $\psi > 1$. Граничные условия ставились на достаточно большом удалении от оси, т.е. при $\psi \gg 1$ скорость и температура струи были равны скорости и температуры окружающей среды.

Численные решения получены для струй истекающие из сопла радиусом 1 дюйм (25,4 мм). Число Маха на выходе из сопла составляли приблизительно $M_{jet} = u_{jet}/a_{jet} = 0,51$ для дозвуковой струи, $M_{jet} = u_{jet}/a_{jet} = 0,376$ для горячей дозвуковой струи, $M_{jet} = u_{jet}/a_{jet} = 0,985$, для околосзвуковой струи. Система (3) решалась для $Re = 5600$. Хорошее соответствие с экспериментальными данными получаются, если в качестве начального условия принять $v_t = 3/Re$.

Использованная для расчетов неявная схема является абсолютно устойчивой и неизвестные на новом слое находились методом прогонки. Для интегрирования были использованы шаги $\Delta \xi = 0,001$, $\Delta \psi = 0,01$. Количество узлов в радиальном направлении было 600. Численные эксперименты показали, что уменьшение шагов интегрирования в два раза приводят к изменению результатов интегрирования не более чем на 0,1%.

4. Численные результаты

Приведем некоторые конкретные примеры, иллюстрирующие кратко описанные выше свойства v_t -92 модели. Все экспериментальные данные были взяты из базы данных [4].

На рис. 1 приведено сравнение результатов осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными для безразмерной осевой скорости в зависимости от расстояния до сопла.

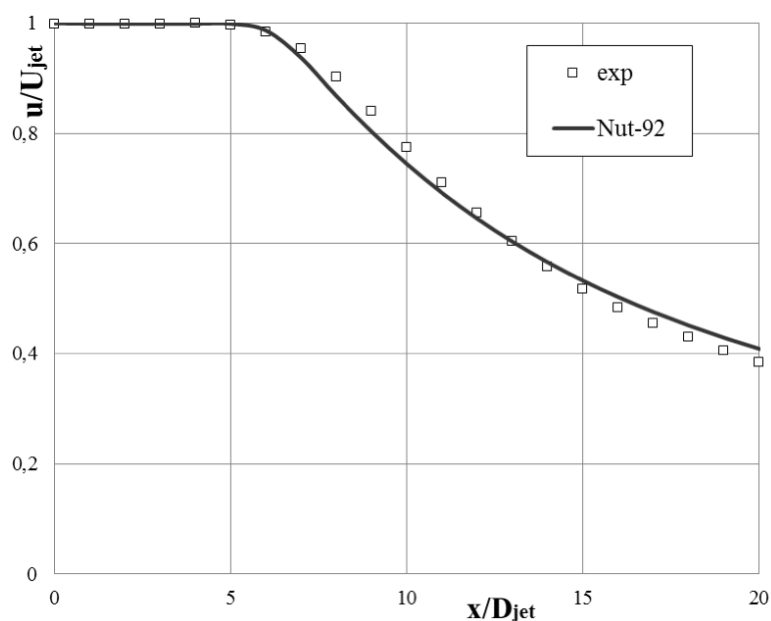


Рис. 1. Сравнение осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными для безразмерной осевой скорости от расстояния до сопла

На рис. 2 представлено сравнение результатов осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными для профилей турбулентного напряжения для различных сечений.

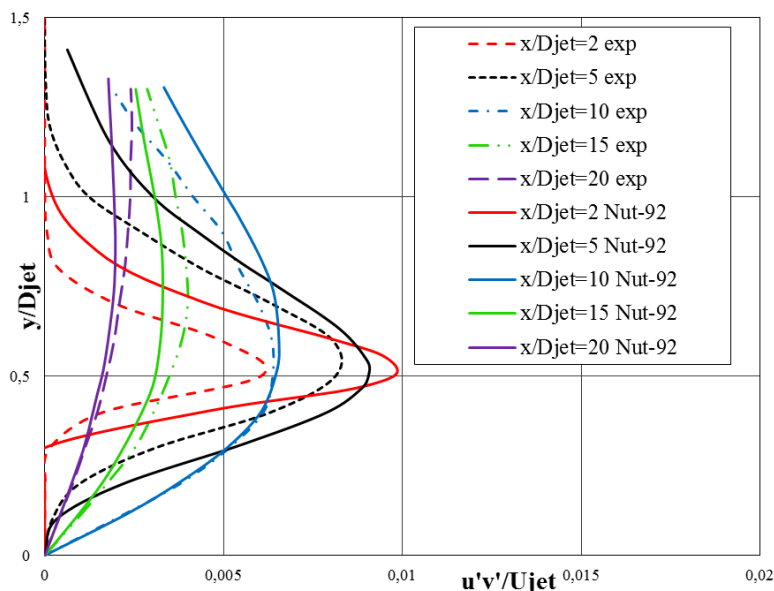


Рис. 2. Сравнение результатов осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными профили турбулентного напряжения для различных сечений в условиях эксперимента

На рис. 3 показано сравнение результатов осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными для профилей безразмерных продольных скоростей в различных расстояниях от сопла.

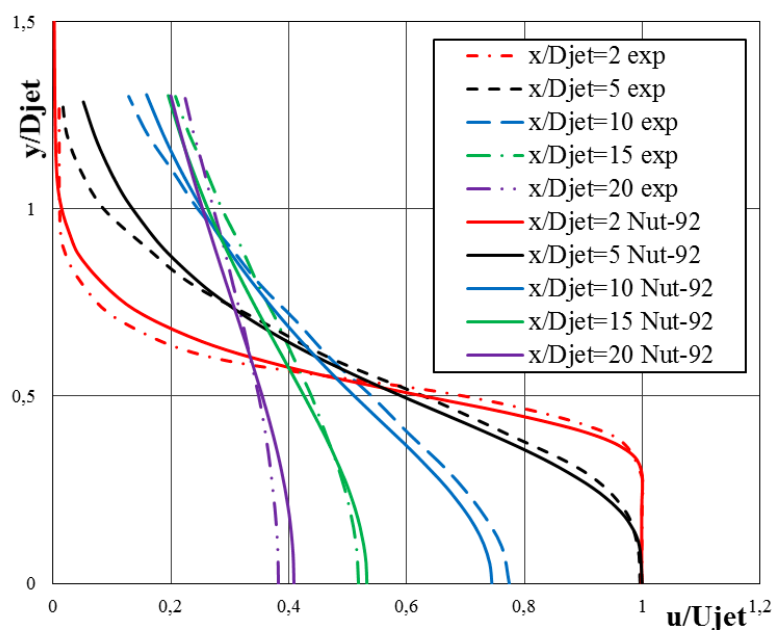


Рис. 3. Сравнение результатов осесимметричной дозвуковой холодной струи при использовании модели ν_t -92 с опытными данными профили продольных скоростей для различных сечений в условиях эксперимента

На рис. 4 выведено сравнение результатов осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными из безразмерной осевой скорости от расстояния до сопла.

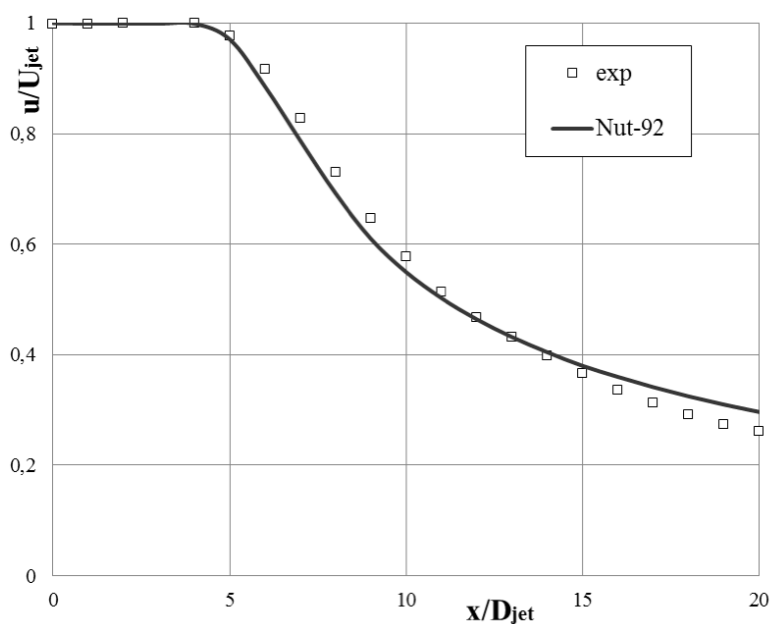


Рис. 4. Сравнение осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными для безразмерной осевой скорости от расстояния до сопла

На рис. 5 представлено сравнение результатов осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными для профилей турбулентного напряжения для различных сечений.

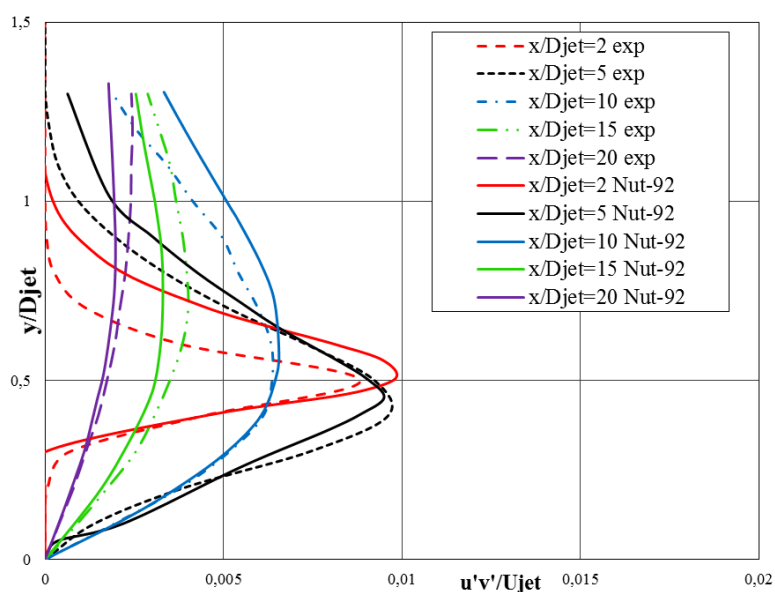


Рис. 5. Сравнение результатов осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными профили турбулентного напряжения для различных сечений в условиях эксперимента

На рис. 6 показано сравнение результатов осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными для профилей безразмерных продольных скоростей в различных расстояниях от сопла.

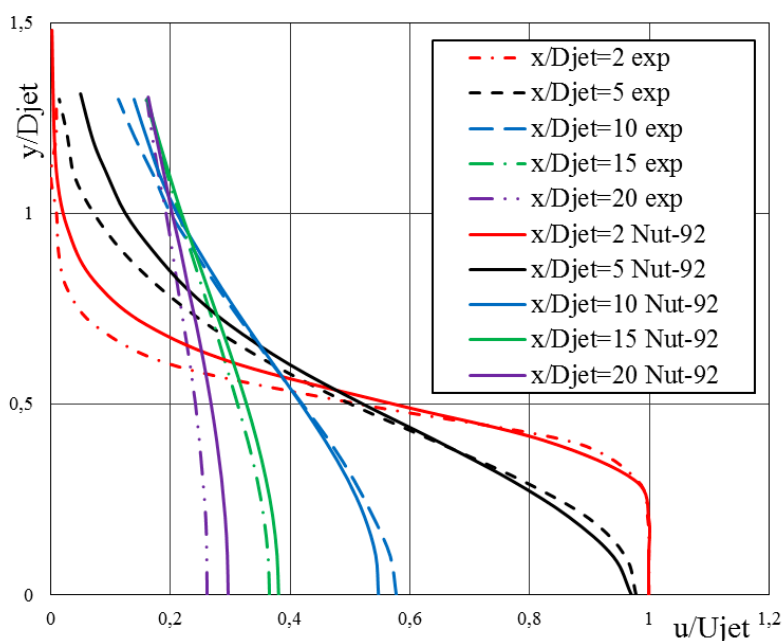


Рис. 6. Сравнение результатов осесимметричной дозвуковой горячей струи при использовании модели ν_t -92 с опытными данными профили продольных скоростей для различных сечений в условиях эксперимента

На рис. 7 представлено сравнение результатов осесимметричной околосзвуковой струи при использовании модели ν_t -92 с опытными данными для безразмерной осевой скорости от расстояния до сопла.

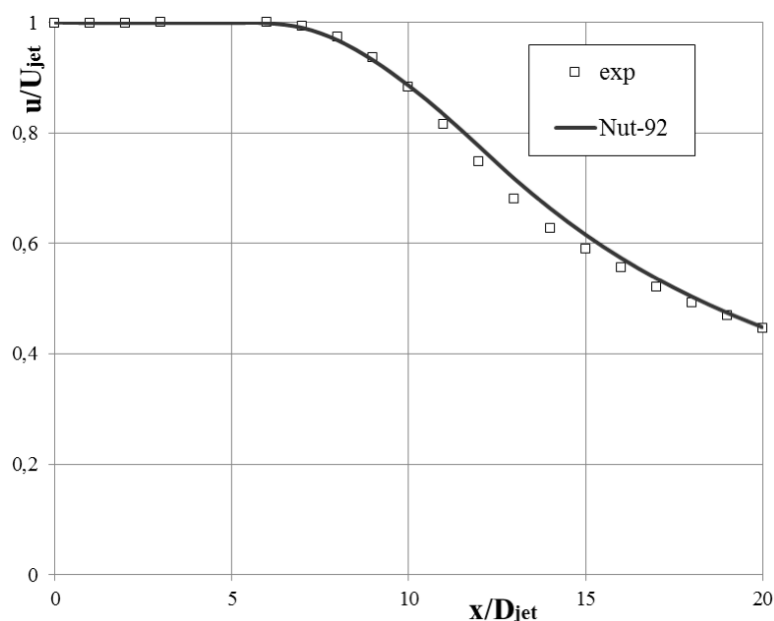


Рис. 7. Сравнение осесимметричной околосзвуковой струи при использовании модели ν_t -92 с опытными данными для безразмерной осевой скорости от расстояния до сопла

На рис. 8 представлено сравнение результатов осесимметричной околосзвуковой струи при использовании модели ν_t -92 с опытными данными для профилей турбулентного напряжения для различных сечений.

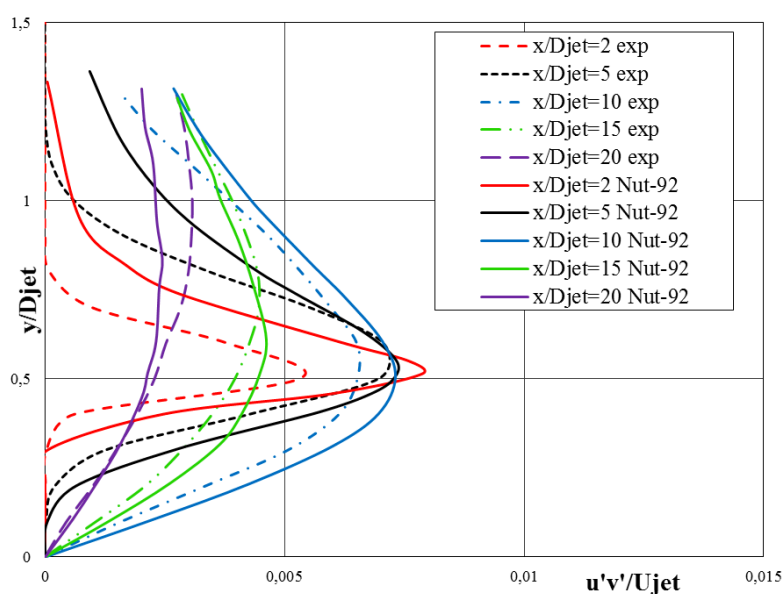


Рис. 8. Сравнение результатов осесимметричной околосзвуковой струи при использовании модели ν_t -92 с опытными данными профили турбулентного напряжения для различных сечений в условиях эксперимента

На рис. 9 представлено сравнение результатов осесимметричной околозвуковой струи при использовании модели ν_t -92 с опытными данными для профилей безразмерных продольных скоростей в различных расстояниях от сопла.

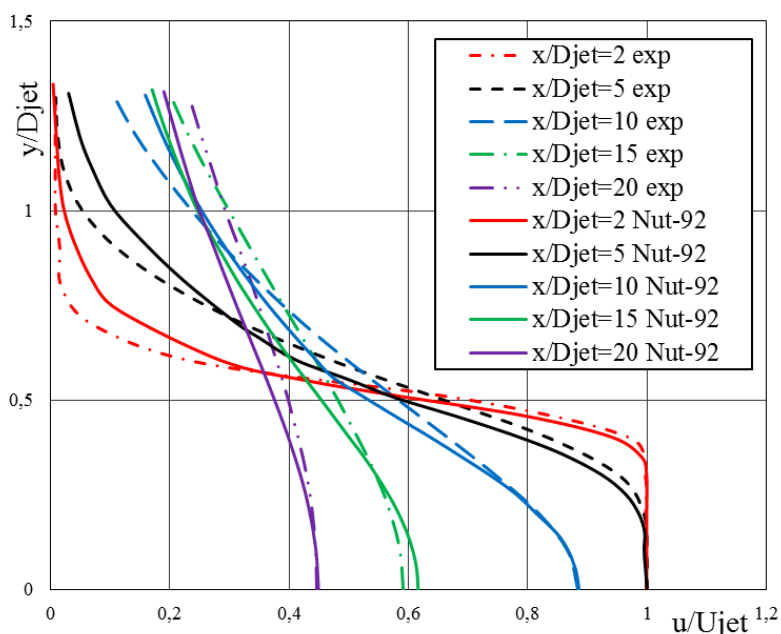


Рис. 9. Сравнение результатов осесимметричной околозвуковой струи при использовании модели ν_t -92 с опытными данными профили продольных скоростей для различных сечений в условиях эксперимента

Заключение

Проведено сравнительное тестирование модели ν_t -92 Секундова [9] для осесимметричных турбулентных дозвуковой холодной, горячей и околозвуковой струй.

Проведено сравнение результатов расчета с результатами экспериментов [4]. Получено, что эта модель турбулентности дает очень близкие численные результаты для струйных течений. В целом наблюдается удовлетворительное совпадение расчетных и экспериментальных данных.

По сравнению с другими моделями RANS модель Секундова ν_t -92 хорошо описывает расширение струй, в то время как многие другие модели турбулентности дают несколько большее расширение холодной, горячей и околозвуковой струй, чем наблюдается в эксперименте. Кроме того модель Секундова ν_t -92 показала хорошая сходимость и проста в численной реализации. Поэтому ее можно рекомендовать для исследования турбулентных струй в различных инженерных задачах.

Литература

1. Абрамович Г.Н. Теория турбулентных струй. М.: Наука, 1984. 716 с.
2. Кудимов Н.Ф., Сафронов А.В., Третьякова О.Н. Результаты экспериментальных исследований взаимодействия многоблочных сверхзвуковых турбулентных струй с преградой // Труды Московского авиационного института. 2013. № 69.
3. Кудимов Н.Ф., Сафронов А.В., Третьякова О.Н. Численное моделирование взаимодействия многоблочных сверхзвуковых турбулентных струй с преградой // Труды Московского авиационного института. 2013. № 70.

4. Cristopher R. Responsible NASA official. «Turbulence modeling Resource. NASA Langley Research Center». URL: <http://turbmodels.larc.nasa.gov> (дата обращения: 04.04.2019).
5. Spalart P.R., Allmaras S.R. A One-Equation Turbulence Model for Aerodynamic Flows // AIAA Journal. 1992. Vol. 439. DOI: 10.2514/6.1992-439.
6. Menter F.R. Two-equation eddy viscosity turbulence models for engineering applications // AIAA Journal. 1994. Vol. 32, no. 8. P. 1598–1605. DOI: 10.2514/3.12149.
7. Malikov Z. Mathematical Model of Turbulence Based on the Dynamics of Two Fluids // Applied Mathematic Modeling. 2020. Vol. 82. P. 409–436. DOI: 10.1016/j.apm.2020.01.047.
8. Bradshaw P., Ferriss D.H., Atwell N.P. Calculation of boundary layer development using the turbulent energy equation // Journal of Fluid Mechanics. 1967. Vol. 28, no. 3. P. 593–616. DOI: 10.1017/S0022112067002319.
9. Shur M., Strelets M., Zaikov L., et al. Comparative Numerical Testing of One- and Two-Equation Turbulence Models for Flows with Separation and Reattachment // AIAA. 1995. DOI: 10.2514/6.1995-863.
10. Mises R. von. Bernerkungen zur Hydrodinamik // Z. Angew. Math. Mech. 1927. Vol. 7. P. 425–431.
11. Сухинов А.И., Чистяков А.Е., Якобовский М.В. Точность численного решения уравнения диффузии-конвекции на основе разностных схем второго и четвертого порядков погрешности аппроксимации // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2016. Т. 5, № 1. С. 47–62. DOI: 10.14529/cmse160105.

Мадалиев Муродил Эркинжон угли, докторант, PhD, Институт механики и сейсмостойкости сооружений АН РУз (Ташкент, Республика Узбекистан)

DOI: 10.14529/cmse200405

NUMERICAL RESEARCH ν_t -92 TURBULENCE MODEL FOR AXISYMMETRIC JET FLOW

© 2020 M.E. Madaliev

*Institute of Mechanics and Earthquake Engineering,
Academy of Sciences of the Republic of Uzbekistan
(Durman yuli 51, Tashkent, 100125 Uzbekistan)*

E-mail: madaliev.me2019@mail.ru

Received: 06.04.2020

In this work, we verify the turbulent Secundov ν_t -92 model for various jets. Based on this model, numerical studies of an axisymmetric subsonic cold, hot, and transonic jet are carried out. For the numerical implementation of the equations of hydrodynamics are written in the Mises variables and an implicit absolutely stable scheme of the second order of accuracy in the transverse and first order in the longitudinal directions is used. To compare the obtained numerical results, we used experimental data from the NASA database, which presents not only experimental data of recent years, but a comparative analysis of many turbulence models. Comparisons were made on the propagation of the axial flow velocity, along the profile of turbulent stresses and longitudinal velocity in various sections. It is shown that the model quantitatively describes well the main parameters of incompressible and compressible turbulent jets. It is shown that the Secundov ν_t -92 model is less prone to the so-called “anomaly” of the circular jet, the essence of which is that many other RANS models give a very strong expansion for axisymmetric jets.

Keywords: Reynolds-averaged Navier–Stokes equations, Secundov ν_t -92 model, sweep, current function, turbulent voltage.

FOR CITATION

Madaliev M.E. Numerical Research v_t -92 Turbulence Model for Axisymmetric Jet Flow. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2020. Vol. 9, no. 4. P. 67–78. (in Russian) DOI: 10.14529/cmse200405.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Abramovich G.N. Theory of turbulent jets. M.: Nauka, 1984. 716 p. (in Russian)
2. Kudimov N.F., Safronov A.V., Tretyakova O.N. The results of experimental studies of the interaction of multiblock supersonic turbulent jets with an obstacle. *Trudy MAI*. 2013. no. 69. (in Russian)
3. Kudimov N.F., Safronov A.V., Tretyakova O.N. Numerical simulation of the interaction of multiblock supersonic turbulent jets with a barrier. *Trudy MAI*. 2013. no. 70. (in Russian)
4. Christopher R. Responsible NASA official. “Turbulence modeling Resource. NASA Langley Research Center”. URL: <http://turbmodels.larc.nasa.gov> (accessed: 04.04.2019).
5. Spalart P.R., Allmaras S.R. A One-Equation Turbulence Model for Aerodynamic Flows. *AIAA Journal*. 1992. Vol. 439. DOI: 10.2514/6.1992-439.
6. Menter F.R. Two-equation eddy viscosity turbulence models for engineering applications. *AIAA Journal*. 1994. Vol. 32, no. 8. P. 1598–1605. DOI: 10.2514/3.12149.
7. Malikov Z. Mathematical Model of Turbulence Based on the Dynamics of Two Fluids. *Applied Mathematic Modeling*. 2020. Vol. 82. P. 409–436. DOI: 10.1016/j.apm.2020.01.047.
8. Bradshaw P., Ferriss D.H., Atwell N.P. Calculation of boundary layer development using the turbulent energy equation. *Journal of Fluid Mechanics*. 1967. Vol. 28, no. 3. P. 593–616. DOI: 10.1017/S0022112067002319.
9. Shur M., Strelets M., Zaikov L., et al. Comparative Numerical Testing of One- and Two-Equation Turbulence Models for Flows with Separation and Reattachment. *AIAA*. 1995. DOI: 10.2514/6.1995-863.
10. Mises R. von. Bemerkungen zur Hydrodynamik. *Z. Angew. Math. Mech.* 1927. Vol. 7. P. 425–431.
11. Sukhinov A.I., Chistakov A.E., Iakobovskii M.V. Accuracy of the Numerical Solution of the Equations of Diffusion-Convection Using the Difference Schemes of Second and Fourth Order Approximation Error. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2016. Vol. 5, no. 1. P. 47–62. (in Russian) DOI: 10.14529/cmse160105.

ОБЗОР СОВРЕМЕННЫХ СИСТЕМ ОБРАБОТКИ ВРЕМЕННЫХ РЯДОВ

© 2020 Е.В. Иванова, М.Л. Цымблер

Южно-Уральский государственный университет

(454080 Челябинск, пр. им. В.И. Ленина, д. 76)

E-mail: elena.ivanova@susu.ru, mzym@susu.ru

Поступила в редакцию: 27.09.2020

Временной ряд представляет собой последовательность хронологически упорядоченных числовых значений, отражающих течение некоторого процесса или явления. В настоящее время одним из наиболее актуальных классов задач обработки временных рядов являются приложения Индустрии 4.0 и Интернета вещей. В данных приложениях типичной является задача обеспечения умного управления и предиктивного технического обслуживания сложных машин и механизмов, которые оснащаются различными сенсорами. Такие сенсоры имеют высокую дискретность снятия показаний и за сравнительно короткое время производят временные ряды длиной от десятков миллионов до миллиардов элементов. Получаемые с сенсоров данные накапливаются и подвергаются интеллектуальному анализу для принятия стратегически важных решений. Обработка временных рядов требует специфического системного программного обеспечения, отличного от имеющихся реляционных СУБД и NoSQL-систем. Системы обработки временных рядов должны обеспечивать, с одной стороны, эффективные операции добавления новых атомарных значений, поступающих в потоковом режиме, а с другой стороны, эффективные операции интеллектуального анализа, в рамках которых временной ряд рассматривается как единое целое. В статье рассмотрены особенности обработки временных рядов в сравнении с данными реляционной и нереляционной природы, и даны формальные определения основных задач интеллектуального анализа временных рядов. Представлен обзор основных возможностей трех наиболее популярных современных систем обработки временных рядов: InfluxDB, OpenTSDB, TimescaleDB.

Ключевые слова: обработка и анализ временных рядов, NoSQL, реляционная СУБД, InfluxDB, OpenTSDB, TimescaleDB.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Иванова Е.В., Цымблер М.Л. Обзор современных систем обработки временных рядов // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 79–97. DOI: 10.14529/cmse200406.

Введение

Временной ряд (time series) представляет собой последовательность хронологически упорядоченных числовых значений, отражающих течение некоторого процесса или явления [16, 17]. Временные ряды возникают в широком спектре предметных областей: мониторинг показателей функциональной диагностики организма человека [21] (временные ряды ЭКГ и ЭЭГ пациента), моделирование климата [65] (временные ряды температуры воздуха, силы ветра в некоторой локации), финансовое прогнозирование [22] (временные ряды курсов акций и валют), геномная инженерия [35] (цепочки ДНК как временные ряды) и др.

В настоящее время приложения Индустрии 4.0 [10] и Интернета вещей [32] представляют собой один из наиболее актуальных классов задач обработки временных рядов. В данных приложениях типичной является задача обеспечения умного управления и предиктивного технического обслуживания сложных машин и механизмов, которые оснащаются различными сенсорами. Такие сенсоры, как правило, имеют высокую дискретность снятия показаний (например, десятки раз в секунду) и за сравнительно короткое время производят

ют временные ряды длиной от десятков миллионов до миллиардов элементов. Получаемые с сенсоров данные временных рядов накапливаются и подвергаются интеллектуальному анализу, который позволяет выявить знания (скрытые тренды, аномалии и др.), необходимые для принятия стратегически важных решений.

Описанный сценарий говорит в пользу того, что обработка данных временных рядов требует специфического системного программного обеспечения, функционал которого должен отличаться как от традиционных систем управления базами данных (СУБД) на основе реляционной модели [18], так и от штатных решений на основе систем класса NoSQL [11]. Системы обработки временных рядов должны обеспечивать, с одной стороны, эффективные операции добавления новых атомарных значений, поступающих в потоковом режиме, а с другой стороны, эффективные операции интеллектуального анализа, в рамках которых временной ряд рассматривается как единое целое. Далее для краткого обозначения систем обработки временных рядов нами будет использоваться аббревиатура СУБД-ВР в соответствии с устоявшимися в англоязычной литературе терминами Time Series DBMS и Time Series Database (см. например, обзоры [5, 19, 41]).

По данным портала DB-Engines.com [12] на момент написания статьи насчитывается более тридцати коммерческих и свободных СУБД-ВР. При этом за последний год СУБД-ВР стабильно входят в тройку наиболее популярных категорий СУБД, где популярность системы представляет собой интегральный показатель частоты ее упоминания в социальных сетях, сервисе Google Trends, на сайтах с предложениями работы и др. Однако тщательный поиск отечественной и зарубежной научной литературы, предпринятый авторами, показывает, что, по-видимому, имеет место относительный недостаток обзорных статей по тематике СУБД-ВР. Например, с момента выхода обзора [41] в 2015 г. появилось большое количество новых систем, в обзоре [5] 2017 г. рассмотрены только свободные системы, а недавний обзор [19] 2020 г. рассматривает только немногочисленные системы с точки зрения поддержки ими краевых вычислений (Edge Computing) [7]. Целью настоящей статьи является попытка восполнить указанный пробел и дать обзор основных современных систем обработки данных временных рядов.

Статья организована следующим образом. В разделе 1 представлены особенности обработки временных рядов в сравнении с данными реляционной и нереляционной природы, и даны формальные определения основных задач обработки временных рядов. В разделе 2 рассмотрены наиболее популярные современные системы обработки временных рядов. Заключение резюмирует результаты исследования.

1. Особенности обработки данных временных рядов

В данном разделе рассматриваются концептуальные сходства и отличия СУБД-ВР от реляционных СУБД и NoSQL-систем (раздел 1.1) и основные задачи интеллектуального анализа данных временных рядов (раздел 1.2).

1.1. Сравнение СУБД-ВР с реляционными СУБД и NoSQL-системами

Реляционные СУБД предполагают два основных класса приложений: *обработка транзакций в реальном времени (OLTP, Online Transaction Processing)* [23] и *интерактивная аналитическая обработка данных (OLAP, Online Analytical Processing)* [48].

В приложениях OLTP под транзакцией понимают набор последовательных операций модификации информации в базе данных, которые рассматриваются как неделимая еди-

ница работы с данными и переводят базу данных из одного согласованного состояния в другое. Результатом транзакции СУБД является фиксация (успешное выполнение) или откат (неудачное выполнение) всех входящих в нее операций. Типичные сценарии использования временных рядов, рассмотренные выше, показывают, что СУБД-ВР не нуждаются в поддержке OLTP: данные накапливаются и могут архивироваться для экономии объема дисковой памяти, но операции модификации и удаления данных отсутствуют.

В этой связи СУБД-ВР имеют схожесть с реляционными СУБД, обслуживающими *хранилища данных (Data Warehouse)* [60]. Хранилище данных представляет собой предметно-ориентированную информационную базу данных, предназначенную для поддержки принятия решений в крупной организации. Данные в хранилище поступают из внешних источников, подвергаются очистке (исправление различного рода ошибок, заполнение пустых значений и др.) и интеграции (приведение к единым форматам и др.), но после добавления не корректируются и не удаляются.

Приложения OLAP строятся на основе хранилищ данных и предполагают подготовку суммарной (агрегированной) информации на основе многомерных данных. Агрегация может выполняться на основе дистрибутивных функций (минимум, максимум, сумма и др.), алгебраических функций (среднее, стандартное отклонение и др.) или целостных функций (медиана, мода и др.). Основной структурой в данном сценарии обработки является OLAP-куб (многомерный массив данных), создаваемый соединением таблиц хранилища в соответствии со схемой «звезда» или «снежинка». В центре схемы находится таблица фактов, содержащая сведения об объектах или событиях, совокупность которых подвергается анализу. Лучами в данной схеме являются таблицы измерений, связанных с таблицей фактов посредством внешнего ключа, которые содержат атрибуты событий, сохраненных в таблице фактов. Количество возможных вариантов агрегации информации определяется на основе числа смысловых уровней иерархии в каждом из измерений.

Однако в случае с СУБД-ВР обработка данных по сценарию OLAP востребована в урезанном варианте: в этом случае время является фактически единственным измерением, агрегация по которому не требует OLAP-куба, поскольку представляет собой примитивную операцию (например, нахождение минимального или максимального значения, стандартного отклонения временного ряда и др.).

Термин *NoSQL* используется для обозначения широкого класса систем обработки данных, не имеющих в своей основе реляционной модели и не использующих язык SQL: системы «ключ — значение», документоориентированные СУБД, графовые СУБД и др. [11]. NoSQL-системы отказываются от поддержки механизма транзакций (жертвуя свойствами атомарности и согласованности данных) для обеспечения масштабируемости системы и доступности данных при высоких нагрузках в распределенных системах обработки данных. В этой связи можно отметить, что СУБД-ВР, как и NoSQL-системы, не нуждаются в механизме транзакций и ориентированы на как можно более эффективное выполнение операций вставки новых данных.

Подобно реляционным СУБД и NoSQL-системам, СУБД-ВР должны обеспечивать пользователю (прикладному программисту) язык баз данных (аналог SQL), синтаксис которого предоставляет возможность формулировать запросы создания, манипулирования и выборки данных.

1.2. Основные задачи интеллектуального анализа временных рядов

Одной из наиболее важных особенностей СУБД-ВР является встроенная поддержка интеллектуального анализа данных временных рядов. К основным задачам относят [16, 17] выявление аномалий, обнаружение шаблонов (лейтмотивов), поиск по образцу, восстановление пропущенных значений и прогноз. Ниже приводится нотация и формализованные определения указанных задач.

Временной ряд представляет собой хронологически упорядоченную последовательность числовых значений: $T = (t_1, \dots, t_n)$, $t_i \in \mathbb{R}$, длина ряда обозначается как $|T| = n$.

Подпоследовательность $T_{i,m}$ временного ряда T представляет собой непрерывное подмножество T , состоящее из m элементов и начинающееся с позиции i : $T_{i,m} = (t_i, \dots, t_{i+m-1})$, $1 \leq i \leq n - m + 1$, $m \ll n$.

Пусть неотрицательная симметричная функция $\text{Dist} : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ используется в качестве *функции расстояния*.

Поиск аномалий предполагает обнаружение подпоследовательностей временного ряда, которые наиболее непохожи на все остальные подпоследовательности ряда. Концепция *диссонанса* (*discord*), предложенная в работе [30], уточняет и формализует понятие аномалии и в настоящее время признается научным сообществом как наиболее адекватный способ поиска аномалий во временном ряде [71]. Диссонанс определяется следующим образом.

Подпоследовательности $T_{i,m}$ и $T_{j,m}$ ряда T называются *непересекающимися*, если $|i - j| \geq m$. Подпоследовательность, которая является *непересекающейся* к данной подпоследовательности $T_{i,m}$, обозначается как $M_{T_{i,m}}$. Подпоследовательность $T_{i,m}$ является *диссонансом*, если

$$\forall T_{j,m}, M_{T_{i,m}} \in T \quad \min(\text{Dist}(T_{i,m}, M_{T_{i,m}})) > \min(\text{Dist}(T_{j,m}, M_{T_{j,m}})). \quad (1)$$

Иными словами, диссонанс представляет собой подпоследовательность ряда, имеющую максимальное расстояние до наиболее близкой к ней *непересекающейся* подпоследовательности. В работах [30] и [70] предложены алгоритмы поиска диссонансов в ряде, целиком размещенном в оперативной памяти, и для случая временного ряда, хранящегося на диске, соответственно. В настоящее время поиск аномалий во временных рядах является сферой интенсивных научных исследований (см., например, обзоры [8, 9]).

Поиск лейтмотивов (шаблонов) предполагает нахождение пар *непересекающихся* подпоследовательностей временного ряда, наиболее похожих друг на друга и формально определяется следующим образом [40]. Пара подпоследовательностей $\{T_{i,m}, T_{j,m}\}$ ряда T называется *лейтмотивом* (*motif*), если

$$\forall a, b, |a - b| \geq m, i, j, |i - j| \geq m \quad \text{Dist}(T_{i,m}, T_{j,m}) \leq \text{Dist}(T_{a,m}, T_{b,m}). \quad (2)$$

В работах [39, 58] были предложены алгоритмы поиска приближенного лейтмотива во временном ряде. В работе [40] предложен алгоритм МК, который находит точный лейтмотив во временном ряде. В настоящее время в области разработки методов и алгоритмов поиска лейтмотивов во временном ряде ведутся интенсивные исследования (см., например, обзоры [62, 63]).

Поиск по образцу предполагает нахождение во временном ряде подпоследовательности, форма которой наиболее похожа на заданный пользователем существенно более короткий временной ряд (поисковый запрос) и формально определяется следующим образом [49].

Пусть имеется временной ряд Q , $m = |Q| \ll n = |T|$, тогда подпоследовательность $T_{i,m}$ является *наиболее похожей* (*best match*) на поисковый запрос, если

$$\forall j, 1 \leq j \leq n - m + 1 \text{ Dist}(Q, T_{i,m}) \leq \text{Dist}(Q, T_{j,m}). \quad (3)$$

Данные выше определения диссонанса, лейтмотива и наиболее похожей подпоследовательности могут быть естественным образом расширены для нахождения $\text{top-}k$ соответствующих объектов временного ряда, где k — параметр поиска. Следует также отметить, что при реализации поиска диссонансов и лейтмотивов временного ряда в качестве метрики используется, как правило, евклидово расстояние или его производные. Однако для задачи поиска по образцу для временных рядов из большинства предметных областей мера схожести DTW (Dynamic Time Warping, динамическая трансформация времени) [6], которая имеет квадратичную временную сложность, считается наиболее адекватной [50]. Разработка методов и алгоритмов поиска похожих подпоследовательностей во временном ряде остается на сегодня областью интенсивных научных исследований [36, 49, 67].

В задаче *восстановления пропущенных значений* (*imputation of missing values*) предполагается, что временной ряд содержит один элемент или подпоследовательности элементов, имеющие пустое значение NULL, которые требуется заменить на синтетические правдоподобные значения. Пустые значения отражают типичную ситуацию отсутствия показаний сенсора вследствие аппаратного или программного сбоя либо человеческого фактора.

В задаче *прогноза* требуется сформировать одно или несколько синтетических значений временного ряда в будущем, основываясь на исторических данных этого временного ряда и/или временных рядов, семантически близких к нему. Задача прогноза может быть рассмотрена как задача восстановления будущих значений временного ряда.

Точность работы алгоритмов восстановления и прогноза (степень правдоподобия генерируемых ими синтетических значений) устанавливается с помощью тестового временного ряда, в котором часть реальных значений заменяется на NULL, и вычисляется одна из мер точности [25], например, среднеквадратическая ошибка (RMSE, Root Mean Square Error):

$$\text{RMSE}(T, \hat{T}) = \sqrt{\frac{1}{n} \sum_{i=1}^n (t_i - \hat{t}_i)^2}, \quad (4)$$

где T и \hat{T} — временные ряды длины n с реальными и восстановленными значениями соответственно.

В настоящее время тематике восстановления пропущенных значений и прогноза во временных рядах посвящается большое количество научных работ. Соответствующие методы и алгоритмы разрабатываются как на базе аппарата статистики и методов интеллектуального анализа данных [3, 31], так и на основе использования технологий нейронных сетей [20, 57, 72].

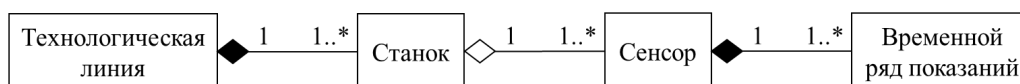
2. Основные классы СУБД-ВР и их основные представители

В работе [5] предложено разделять системы управления временными рядами на четыре следующих класса. В первый класс входят СУБД-ВР, в которых хранение временных рядов осуществляется с помощью сторонних реляционных СУБД или NoSQL-систем. Второй класс включает СУБД-ВР, самостоятельно выполняющие хранение временных рядов. В третий класс входят реляционные СУБД, обеспечивающие средства для хранения и обра-

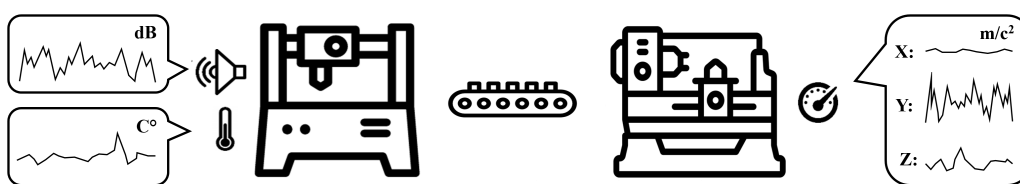
ботки временных рядов. Четвертый класс представляют коммерческие СУБД-ВР независимо от их базовой модели данных, использования сторонней СУБД либо NoSQL-системы для хранения временных рядов и др.

В настоящей статье предлагается разделить многообразие современных СУБД-ВР на два следующих класса: нативные и надстроечные. *Нативная СУБД-ВР* представляет собой самостоятельную проприетарную или свободную разработку с оригинальным языком запросов, машиной баз данных, системой хранения данных и др. Представителями данного класса являются системы InfluxDB [19, 27], Kdb+ [29], Prometheus [5, 47], Druid [68], LittleTable [51], FluteDB [34], PhilDB [37], EdgeDB [69], TSMMDB [33] и др.

Надстроечная СУБД-ВР реализуется на основе сторонней системы, обеспечивающей настройку машину баз данных и систему хранения данных. В зависимости от модели данных, используемой сторонней системой, мы можем различать СУБД-ВР, которые являются надстройками над NoSQL-системой либо над реляционной СУБД. В настоящее время имеется широкий спектр СУБД-ВР надстроек над различными NoSQL-системами. СУБД-ВР OpenTSDB [5, 43] и Gorilla [44] функционирует на основе системы HBase [38]. СУБД-ВР VTrDB [2] может быть развернута на основе распределенных файловых систем, например, HDFS [59], GlusterFS [55], CephFS [55] и др., либо на базе одной из следующих NoSQL-систем: MongoDB [26], Cassandra, HBase [38]. Система KairosDB [5, 28] работает на основе системы Cassandra [38]. Система tadb [13] разработана для функционирования в связке со встраиваемой СУБД Berkeley DB [54]. Базисом СУБД-ВР HeteroTSDB [64] является система Amazon DynamoDB [56]. СУБД-ВР Riak TS [19, 53] разработана как расширение системы Riak KV [52]. В подкласс СУБД-ВР, являющихся надстройками над реляционными СУБД, входят системы TimescaleDB [19, 61] (разработаны на основе СУБД PostgreSQL) и RecovDB [4] (разработана на основе СУБД MonetDB [26]).



а) Диаграмма классов



б) Иллюстративный пример

Рис. 1. Модельная предметная область

Далее представлен более детальный обзор возможностей трех систем: InfluxDB, OpenTSDB и TimescaleDB, — которые по данным портала DB-Engines.com [12] на сегодня являются наиболее популярными представителями перечисленных выше категорий СУБД-ВР. Для иллюстрации возможностей СУБД-ВР нами будет использоваться пример предметной области приложения Индустрии 4.0, представленный на рис. 1. Данный пример моделирует технологическую линию по производству изделий из металла, включающую в себя два станка. На станках установлены сенсоры для сбора данных, интеллектуальный анализ которых позволяет осуществлять предиктивное техническое обслуживание линии. На первом станке для контроля перегрева металла установлен сенсор температуры и для

улавливания волн, возникающих при изменениях в структуре металла (трещины, коррозия и др.) — сенсор акустической эмиссии. Каждый из указанных сенсоров выдает одно значение. На втором станке для контроля вибраций станка установлен виброакселерометр, который выдает три значения (виброускорение по осям X, Y, Z).

2.1. Нативная СУБД-ВР InfluxDB

InfluxDB [27] представляет собой свободную СУБД-ВР, написанную на языке программирования Go [14], распространяемую в виде исполняемого файла для основных операционных систем и аппаратных платформ. Для доступа к базе данных InfluxDB поддерживает интерфейсы командной строки и через протокол HTTP, а также клиентские библиотеки и плагины [46].

Организация хранения данных. В InfluxDB данные представляются в виде двумерной таблицы, называемой *измерением* (*measurement*). В измерении имеется столбец с *метками времени* (*timestamp*). Остальные столбцы измерения могут принадлежать одной из двух категорий: поле или тег. *Поле* (*field*) хранит данные временного ряда и состоит из *ключей* (*field keys*) и *значений* (*field values*). *Тег* (*tag*) представляет собой метаданные поля и состоит из *ключей тегов* (*tag key*) и *значений тегов* (*tag values*). Поля не индексируются, но для тегов могут быть созданы индексы. В InfluxDB отсутствует явная схема базы данных.

В InfluxDB поддерживаются понятия серии и точки. *Серия* (*series*) представляет собой набор данных, имеющих общие измерение, набор тегов и ключи полей. *Точка* (*point*) представляет собой элемент данных, состоящий из следующих компонентов: измерение, набор тегов, набор полей, метка времени. Точка однозначно идентифицируется по ее серии и метке времени.

```

1 CREATE DATABASE SensorDB
2 INSERT acoustic , machine=1 val=46
3 INSERT temperature , machine=1 val=21
4 INSERT accelerometer , machine=2 x=34.7 , y=5.0 , z=134.4
5 — Просмотр схемы созданной базы данных
6 USE SensorDB
7 SHOW SERIES
8 acoustic, machine=1
9 temperature, machine=1
10 accelerometer, machine=2

```

Рис. 2. Создание базы данных в InfluxDB

На рис. 2 представлен пример создания базы данных для модельной предметной области, описанной на рис. 1. В данном примере создается база данных, содержащая три измерения: *acoustic*, *temperature* и *accelerometer*, — которые используются для хранения данных соответствующих сенсоров. Каждое из этих измерений имеет тег *machine* для указания станка, на котором устанавливается сенсор. Поля измерений — *val* для измерений *acoustic* и *temperature*, *x*, *y*, *z* для измерения *accelerometer* — служат для идентификации значений, измеряемых соответствующим сенсором. Создание указанных измерений выполняется попутно с добавлением точек данных в базу данных с помощью команды *INSERT*, поскольку в InfluxDB отсутствует возможность явного задания схемы данных. При добавлении точек данных метки времени добавляются автоматически. После этого с помощью

команды доступа к словарию СУБД-ВР SHOW SERIES выводится список созданных серий.

Хранение данных на физическом уровне в InfluxDB основано на использовании древовидной структуры данных LSM (Log-structured merge-tree) [42], которая используется в реляционных СУБД и обеспечивает быстрый доступ к данным в случае сценария работы, предполагающего частые запросы на вставку данных. В InfluxDB также поддерживается автоматическое сжатие данных для минимизации объема хранимых данных.

Язык запросов. В InfluxDB поддерживается SQL-подобный язык запросов InfluxQL. На рис. 3 приведен пример запроса, вычисляющего минимальное значение точек данных температурного сенсора, установленного на первом станке.

```

1 SELECT MIN(val)
2 FROM "temperature"
3 WHERE "machine"='1'

```

Рис. 3. Запрос на выборку данных в InfluxQL

Из средств интеллектуального анализа временных рядов в языке InfluxQL обеспечивается прогнозирование значений ряда с помощью метода Холта–Винтерса [24, 66]. На рис. 4 приведен пример прогноза значений температурного сенсора.

```

1 — 1 шаг: настройка параметров.
2 — Получение данных сенсора для визуального определения параметров
3 — (промежуток между "пиками" и "впадинами" и интервал смещения).
4 SELECT "val"
5 FROM "SensorDB"
6 WHERE "sensor"='temperature' AND
7   time>='2020-08-22 22:12:00' AND time<='2020-08-28 03:00:00'
8 — 2 шаг: формирование линии тренда по настроенным параметрам.
9 SELECT FIRST("val")
10 FROM "SensorDB"
11 WHERE "sensor"='temperature' AND
12   time >= '2020-08-22 22:12:00' AND time <= '2020-08-28 03:00:00'
13 GROUP BY time(379m,348m)
14 — 3 шаг: прогноз.
15 — Прогноз 10 значений после 2020-08-28 03:00:00,
16 — по 4 точки в каждом интервале смещения.
17 SELECT HOLT_WINTERS_WITH_FIT(FIRST("val"), 10, 4)
18 FROM "SensorDB"
19 WHERE "sensor"='temperature' AND
20   time >= '2020-08-22 22:12:00' AND time <= '2020-08-28 03:00:00'
21 GROUP BY time(379m, 348m)

```

Рис. 4. Прогноз значения временного ряда в InfluxQL

InfluxDB поддерживает *непрерывные запросы (continuous query)* — запросы, которые запускаются автоматически с заданной периодичностью. На рис. 5 приведен пример непрерывного запроса, который запускается каждую минуту и находит минимальное значение показаний температурного сенсора за один час.

2.2. Надстроечная СУБД-ВР OpenTSDB на основе NoSQL-систем

OpenTSDB [43, 46] представляет собой свободную СУБД-ВР, написанную на языке программирования Java. OpenTSDB работает как надстройка над NoSQL-системами на основе

```

1 CREATE CONTINUOUS QUERY "cq_minimum" ON "SensorDB"
2 BEGIN
3   SELECT MIN("val") INTO "min_temperature"
4   FROM "SensorDB"
5   WHERE "sensor"='temperature'
6   GROUP BY time(1h)
7 END
8 SELECT * FROM "min_temperature"

```

Рис. 5. Непрерывный запрос в InfluxQL

семейства столбцов (column family store) HBase либо Cassandra [38]. Для доступа к базе данных OpenTSDB поддерживает интерфейсы командной строки и через протокол HTTP, а также клиентские библиотеки и плагины.

Организация хранения данных. Элемент временного ряда в OpenTSDB представляет собой набор, состоящий из вещественного значения, уникального идентификатора временного ряда (в терминах данной системы — метрика, *metric*), метки времени и непустого набора тегов. Тег представляет собой символьную строку для хранения метаданных.

OpenTSDB наследует способ организации данных от нижележащей системы класса NoSQL-системы. При этом NoSQL-система использует для хранения данных следующие системные таблицы с зарезервированными именами: *tsdb* для данных из временных рядов и *tsdb-uid tsdb-tree*, *tsdb-meta* для служебных данных. Запись таблицы *tsdb* представляет собой совокупность значения элемента ряда, метки времени и значения внешнего ключа, который ссылается на таблицу *tsdb-uid* и ассоциирует данную запись с определенным временным рядом. Таблица *tsdb-uid* хранит имена метрик и значения тегов временных рядов. Таблица *tsdb-tree* используется для задания и поддержки в OpenTSDB семантической иерархии хранимых временных рядов подобно файловой структуре в операционной системе. Таблица *tsdb-meta* позволяет хранить дополнительную информацию о временных рядах, задаваемую пользователем (например, текстовую аннотацию).

```

1 # Создание системных таблиц в HBase для работы OpenTSDB
2 env ./src/create_table.sh
3 # Создание таблиц в OpenTSDB и вставка в них данных
4 put acoustic 2020-08-22 22:12:00 46 machine=1
5 put temperature 2020-08-22 22:12:00 21 machine=1
6 put accelerometer.x 2020-08-22 22:14:00 34.7 machine=2
7 put accelerometer.y 2020-08-22 22:14:00 5.0 machine=2
8 put accelerometer.z 2020-08-22 22:14:00 134.4 machine=2

```

Рис. 6. Создание базы данных в OpenTSDB

На рис. 6 представлен пример создания базы данных для модельной предметной области, описанной на рис. 1, когда в качестве базиса OpenTSDB выступает система HBase. Сначала HBase с помощью стандартного скрипта создает системные таблицы для хранения данных. Далее создается база данных, содержащая пять метрик: *acoustic*, *temperature* и *accelerometer.x*, *accelerometer.y*, *accelerometer.z*, — которые используются для хранения данных соответствующих сенсоров. Каждая из этих метрик имеет тег *machine* для указания станка, на котором устанавливается сенсор. Создание указанных измерений выполняется попутно с добавлением точек данных в базу данных с помощью команды *put*, поскольку

в OpenTSDB отсутствует возможность явного задания схемы данных. После команды `put` указывается имя метрики, отметка времени, значение точки данных и теги.

Язык запросов. В OpenTSDB запросы к базе данных записываются с помощью языка JSON. Запрос описывает ориентированный ациклический графа (execution graph), узлы которого определяют источники данных и операции преобразования данных. В запросах поддерживаются операции вычисления арифметических и логических выражений, фильтры, группировка и др., а также статистические и аналитические функции: понижение частоты дискретизации ряда (downsampling), восстановление пропущенных значений (interpolation) и др.

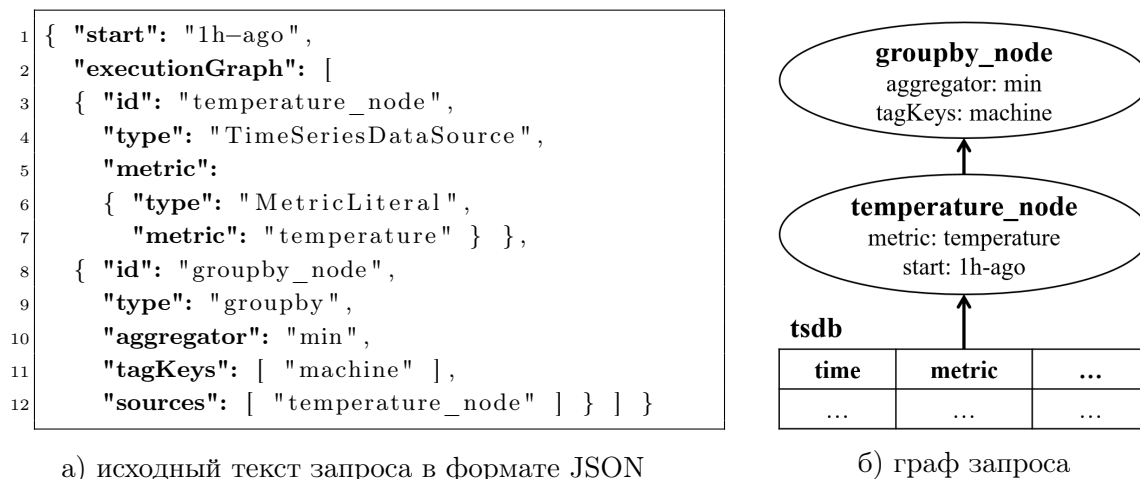


Рис. 7. Запрос на выборку данных в OpenTSDB

На рис. 7 приведен пример запроса, выполняющего операцию группировки с вычислением минимального значения по данным температурного сенсора, полученным за последний час. Граф запроса состоит из следующих узлов: узел запроса `temperature_node`, выполняющий чтение данных из метрики `temperature`, и узел преобразования данных `groupby_node`, выполняющий группировку данных по тегу `machine` и поиск минимального значения.

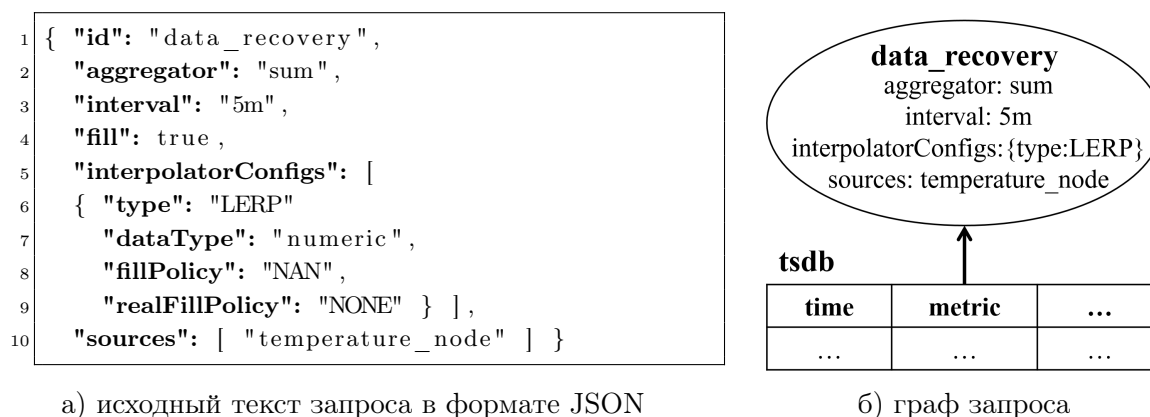


Рис. 8. Восстановление значений временного ряда в OpenTSDB

На рис. 8 приведен пример запроса, выполняющего суммирование данных из метрики `temperature_node`, сгруппированных по интервалу времени 5 мин. Для получившихся пустых групп производится автозаполнение суммой, вычисляемой методом линейной интерполяции (LERP, linear interpolation) [45]. В случае, когда реальных данных для интерполяции недостаточно, в качестве суммы выдается неопределенное значение “NaN”.

За исключением линейной интерполяции, OpenTSDB не поддерживает развитых средств интеллектуального анализа временных рядов, однако допускает расширения от сторонних разработчиков, обеспечивающие указанную функциональность (например, библиотека R2Time [1], реализованная на языке программирования R).

2.3. Надстроечная СУБД-ВР TimescaleDB на основе реляционной СУБД

TimescaleDB [61] представляет собой СУБД-ВР с открытым исходным кодом, написанную на языке программирования C и распространяемую как расширение (extension) реляционной СУБД PostgreSQL. TimescaleDB работает в связке с экземпляром PostgreSQL и штатным образом поддерживает те же операции, что могут быть выполнены в PostgreSQL.

Организация хранения данных. В TimescaleDB данные временных рядов хранятся и обрабатываются в гипертаблицах. *Гипертаблица (hypertable)* задает именованный набор временных рядов и способ разбиения данных указанных рядов по физически хранимым реляционным таблицам. Сведения о разбиении используются для параллельной обработки указанных таблиц в PostgreSQL.

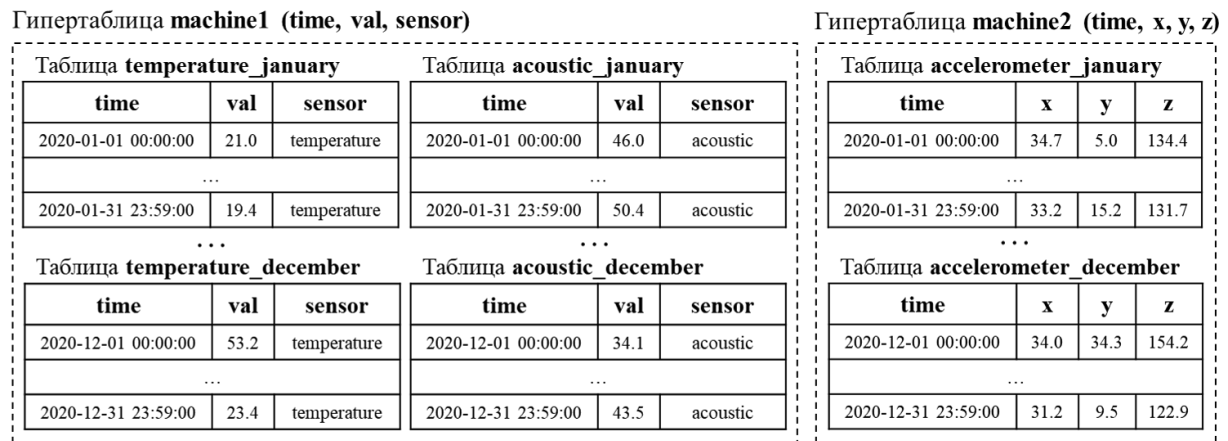


Рис. 9. Гипертаблицы в TimescaleDB

Рисунок 9 иллюстрирует концепцию гипертаблиц на примере данных для модельной предметной области, представленной на рис. 1. Для хранения временных рядов показаний сенсоров каждого из двух станков создаются две гипертаблицы. Гипертаблица *machine1* используется для хранения данных сенсоров, установленных на первом станке, и имеет следующие атрибуты: метка времени, показание сенсора и тип сенсора (сенсор акустической эмиссии либо температурный сенсор). Гипертаблица *machine2* определяет способ хранения показаний виброускорения по осям X, Y, Z с сенсора, установленного на втором станке. Обе гипертаблицы задают разбиение временного ряда на непересекающиеся подпоследовательности, соответствующие периодам в один месяц. Кроме того, гипертаблица *machine1* обеспечивает хранение показаний различных сенсоров в отдельных таблицах.

Рисунок 10 показывает пример создания базы данных в TimescaleDB с гипертаблицами, представленными на рис. 9. Сначала выполняется создание новой базы данных и ее расширение. Далее создаются таблицы, которые с помощью системной функции преобразуются в гипертаблицы с указанием способа разбиения. Вставка данных в гипертаблицу выполняется с помощью обычной SQL-команды INSERT.

```

1  — Создание базы данных в PostgreSQL и ее расширение с помощью TimescaleDB
2  CREATE DATABASE SensorsDB;
3  \connect SensorsDB
4  CREATE EXTENSION timescaledb;
5  — Создание таблиц для сенсорных данных и их преобразование в гипертаблицы
6  CREATE TABLE machine1 (
7      time      TIMESTAMP,
8      val       REAL,
9      sensor    TEXT      )
10 CREATE TABLE machine2 (
11     time      TIMESTAMP,
12     x, y, z   REAL      )
13 SELECT create_hypertable('machine1', 'time', 'sensor', 2,
14                          chunk_time_interval => INTERVAL '1 month');
15 SELECT create_hypertable('machine2', 'time',
16                          chunk_time_interval => INTERVAL '1 month');
17 — Вставка данных в гипертаблицы
18 INSERT INTO machine1 (time, val, sensor) VALUES (NOW(), 46.0, 'acoustic');
19 INSERT INTO machine1 (time, val, sensor) VALUES (NOW(), 21.0, 'temperature');
20 INSERT INTO machine2 (time, x, y, z) VALUES (NOW(), 34.7, 5.0, 134.4);

```

Рис. 10. Создание базы данных в TimescaleDB

Язык запросов. Выборка данных из гипертаблиц в TimescaleDB осуществляется с помощью запросов SQL (команда `SELECT` с широким набором стандартных возможностей языка: подзапросы, сортировка, группировка и др.). Кроме того, в TimescaleDB язык запросов дополнен функциями, позволяющими выполнять статистический анализ временных рядов: вычисление медианы, скользящего среднего и процентилей, построение гистограмм, группировка по заданному временному интервалу и др.

TimescaleDB поддерживает *непрерывные агрегаты* (*continuous aggregates*) — представления, которые в фоновом режиме автоматически вычисляют и материализуют результаты специфицированного запроса. Непрерывные агрегаты похожи на материализованные представления (*materialized view*) в PostgreSQL, но, в отличие от последних, непрерывные агрегаты не нужно обновлять вручную: представление будет обновляться автоматически по мере добавления или изменения данных. На рис. 11 приведен пример непрерывного агрегата, который вычисляет среднее значение показаний температурного сенсора и группирует эти значения по периодам, равным одному часу.

```

1  CREATE VIEW ca_minimum WITH (timescaledb.continuous) AS
2  SELECT
3      time_bucket(INTERVAL '1 hour', time) AS bucket, MIN(val)
4  FROM temperature
5  GROUP BY bucket;
6  SELECT * FROM ca_minimum;

```

Рис. 11. Непрерывный агрегат в TimescaleDB

TimescaleDB не предоставляет штатных функций интеллектуального анализа данных временных рядов, однако наследует от PostgreSQL возможность интеграции с библиотеками сторонних разработчиков, реализующими функции интеллектуального анализа данных внутри СУБД (например, Apache MADlib [15]), а также поддерживает возможность реализации пользовательских функций (UDF, *user-defined function*) для интеллектуального анализа временных рядов на языках программирования R и Python.

Заключение

В статье представлен обзор основных современных систем, обеспечивающих эффективное хранение и обработку данных временных рядов. Временной ряд представляет собой последовательность хронологически упорядоченных числовых значений, отражающих течение некоторого процесса или явления. Временные ряды возникают в широком спектре предметных областей: мониторинг показателей функциональной диагностики организма человека, моделирование климата, финансовое прогнозирование, геномная инженерия, умное управление и предиктивное техническое обслуживание сложных машин и механизмов в приложениях Индустрии 4.0 и Интернета вещей и др.

Рассмотрена специфика обработки данных временных рядов, требующая системного программного обеспечения, отличного от имеющихся реляционных СУБД и NoSQL-систем. Обработка временных рядов не предполагает наличие транзакций в реальном времени (сценарий OLTP): данные временных рядов накапливаются, операции модификации и удаления данных, как правило, отсутствуют. В этом смысле системы обработки временных рядов близки к реляционным СУБД для хранилищ данных, однако интерактивная аналитическая обработка данных (сценарий OLAP) востребована здесь в урезанном виде: время является фактически единственным измерением, агрегация по которому не требует OLAP-куба, поскольку представляет собой примитивную операцию (нахождение минимального или максимального значения, стандартного отклонения временного ряда и др.). Системы обработки временных рядов, как и NoSQL-системы, не нуждаются в механизме транзакций и ориентированы на как можно более эффективное выполнение операций вставки новых данных. Подобно реляционным СУБД и NoSQL-системам, системы обработки временных рядов должны обеспечивать язык баз данных (аналог SQL) для формулирования запросов на создание, манипулирование и выборку данных. Важной особенностью систем обработки временных рядов является необходимость поддержки эффективных операций интеллектуального анализа данных, в рамках которых временной ряд рассматривается как единое целое. Даны формальные определения основных задач интеллектуального анализа временных рядов: выявление аномалий, обнаружение шаблонов (лейтмотивов), поиск по образцу, восстановление пропущенных значений и прогноз.

Предложено деление современных систем обработки временных рядов на два класса: нативные и надстроечные. Нативная система представляет собой самостоятельную проприетарную или свободную разработку с оригинальным языком запросов, машиной баз данных, системой хранения данных и др. Надстроечная система реализуется на базе существующей системы класса NoSQL либо реляционной СУБД, обеспечивающей надстройку машину баз данных и систему хранения. Представлен обзор основных возможностей следующих наиболее популярных современных систем обработки временных рядов: нативная система InfluxDB, система OpenTSDB, являющаяся надстройкой над NoSQL-системой, и система TimescaleDB, являющаяся надстройкой над реляционной СУБД. Возможности указанных систем проиллюстрированы на примере модельной предметной области приложениях Индустрии 4.0. Следует отметить, что большинство современных систем обработки временных рядов предоставляют достаточно узкий спектр встроенных средств интеллектуального анализа данных. Как правило, указанные средства позволяют решать лишь задачи восстановления пропусков или/и прогноза значений временного ряда. В соответствии с этим актуальной задачей является разработка методов и подходов для расширения спектра средств интеллектуального анализа данных, выполняемого в рамках систем обработки временных рядов.

Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (грант № 20-07-00140) и Министерства науки и высшего образования РФ (государственное задание FENU-2020-0022).

Литература

1. Agrawal B., Chakravorty A., Rong C., Wlodarczyk T.W. R2Time: A framework to analyse Open TSDB time-series data in HBase // Proceedings of the 6th International Conference on Cloud Computing Technology and Science, CloudCom 2014 (Singapore, December, 15–18, 2014). IEEE, 2014. P. 970–975. DOI: 10.1109/CloudCom.2014.84.
2. Andersen M.P., Culler D.E. BTrDB: Optimizing storage system design for timeseries processing // Proceedings of the 14th USENIX Conference on File and Storage Technologies, FAST 2016 (Santa Clara, United States, February, 22–25, 2016). P. 39–52. URL: <https://www.usenix.org/system/files/conference/fast16/fast16-papers-andersen.pdf> (дата обращения: 30.07.2020).
3. Andiojaya A., Demirhan H. A bagging algorithm for the imputation of missing values in time series // Expert Syst. Appl. 2019. Vol. 129. P. 10–26. DOI: 10.1016/j.eswa.2019.03.044.
4. Arous I., Khayati M., Cudre-Mauroux P., et al. RecovDB: Accurate and efficient missing blocks recovery for large time series // Proceedings of the 35th International Conference on Data Engineering, ICDE 2019 (Macao, Macao, April, 8–11, 2019). IEEE Computer Society, 2019. P. 1976–1979. DOI: 10.1109/ICDE.2019.00218.
5. Bader A., Kopp O., Falkenthal M. Survey and comparison of open source time series databases // Proceedings of the Workshop on Business, Technologies and Web, BTW 2017 (Stuttgart, Germany, March, 6–7, 2017). Gesellschaft fur Informatik, 2017. P. 249–268. URL: <https://dl.gi.de/bitstream/handle/20.500.12116/922/paper31.pdf> (дата обращения: 16.07.2020).
6. Berndt D.J., Clifford J. Using Dynamic Time Warping to find patterns in time series // Knowledge Discovery in Databases: Papers from the 1994 AAAI Workshop (Seattle, Washington, USA, July 1994). 1994. P. 359–370.
7. Cao K., Liu Y., Meng G., Sun O. An overview on Edge Computing research // IEEE Access. 2020. Vol. 8. P. 85714–85728. DOI: 10.1109/ACCESS.2020.2991734.
8. Chandola V., Banerjee A., Kumar V. Anomaly detection for discrete sequences: A survey // IEEE Trans. Knowl. Data Eng. 2012. Vol. 24, no. 5. P. 823–839. DOI: 10.1109/TKDE.2010.235.
9. Cook A.A., Misirli G., Fan Z. Anomaly detection for IoT time-Series data: A Survey // IEEE Internet Things Journal. 2020. Vol. 7, no. 7. P. 6481–6494. DOI: 10.1109/JIOT.2019.2958185.
10. Da X.L., Duan L. Big data for cyber physical systems in Industry 4.0: a survey // Enterp. Inf. Syst. 2019. Vol. 13, no. 2. P. 148–169. DOI: 10.1080/17517575.2018.1442934.
11. Davoudian A., Chen L., Liu M. A survey on NoSQL stores // ACM Comput. Surv. 2018. Vol. 51, no. 2. P. 40:1–40:43. DOI: 10.1145/3158661.
12. DB-Engines Ranking of Time Series DBMS. URL: <https://db-engines.com/en/ranking/time+series+dbms> (дата обращения: 16.07.2020).

13. Deri L., Mainardi S., Fusco F. tsdb: A compressed database for time series // Proceedings of the 4th International Workshop on Traffic Monitoring and Analysis, TMA 2012 (Vienna, Austria, March, 12, 2012). P. 143–156. DOI: 10.1007/978-3-642-28534-9_16.
14. Donovan A.A.A., Kernighan B.W. The Go programming language. Addison-Wesley, 2015. 380 p. ISBN: 978-0134190440.
15. Hellerstein J.M., Re C., Schoppmann F., et al. The MADlib analytics library or MAD skills, the SQL // PVLDB. 2012. Vol. 5, no. 12. P. 1700–1711. DOI: 10.14778/2367502.2367510.
16. Esling P., Agon C. Time-series data mining // ACM Comput. Surv. 2012. Vol. 45, no. 1. P. 12:1–12:34. DOI: 10.1145/2379776.2379788.
17. Fu T.C. A review on time series data mining // Eng. Appl. of AI. 2011. Vol. 24, no. 1. P. 164–181. DOI: 10.1016/j.engappai.2010.09.007.
18. Garcia-Molina H., Ullman J.D., Widom J. Database systems – the complete book. Pearson, 2009. 1203 p.
19. Grzesik P., Mrozek D. Comparative analysis of time series databases in the context of Edge computing for low power sensor networks // Proceedings of the 20th International Conference on Computational Science, ICCS 2020 (Amsterdam, The Netherlands, June, 3–5, 2020). Part V. 2020. P. 371–383. DOI: 10.1007/978-3-030-50426-7_28.
20. Guo Z., Wan Y., Ye H. A data imputation method for multivariate time series based on generative adversarial network // Neurocomputing. 2019. Vol. 360. P. 185–197. DOI: 10.1016/j.neucom.2019.06.007.
21. Hamdi S., Chaabane N., Bedoui M.H. Intra and Inter Relationships between Biomedical Signals: A VAR Model Analysis // Proceedings of the International Conference on Information Visualisation, IV 2019 (Paris, France, July, 2–5, 2019). P. 411–416. DOI: 10.1109/IV.2019.00076.
22. Hanif M. Relationship between oil and stock markets: Evidence from Pakistan stock exchange // International Journal of Energy Economics and Policy. 2020. Vol. 10, no. 5. P. 150–157. DOI: 10.32479/ijeep.9653.
23. Harizopoulos S., Abadi D.J., Madden S., Stonebraker M. OLTP through the looking glass, and what we found there // Making Databases Work: the Pragmatic Wisdom of Michael Stonebraker / Ed. by Brodie M.L. ACM / Morgan & Claypool, 2019. P. 409–439. DOI: 10.1145/3226595.3226635.
24. Holt C.E. Forecasting seasonals and trends by exponentially weighted averages // International Journal of Forecasting. 2004. Vol. 20, no. 1. P. 5–10. DOI: 10.1016/j.ijforecast.2003.09.015.
25. Hyndman R.J., Koehler A.B. Another look at measures of forecast accuracy // International Journal of Forecasting. 2006. Vol. 22, no. 4. P. 679–688. DOI: 10.1016/j.ijforecast.2006.03.001.
26. Idreos S., Groffen F., Nes N., et al. MonetDB: Two decades of research in column-oriented database architectures // IEEE Data Engineering Bulletin. 2012. Vol. 35, no. 1. P. 40–45.
27. InfluxDB 1.8 Documentation. URL: <https://docs.influxdata.com/influxdb/v1.8/> (дата обращения: 27.09.2020).
28. KairosDB documentation. URL: <https://kairosdb.github.io/docs/build/html/> (дата обращения: 27.09.2020).

29. Kdb+ and q documentation. URL: <https://code.kx.com/> (дата обращения: 27.09.2020).
30. Keogh E., Lin J., Fu A. HOT SAX: efficiently finding the most unusual time series subsequence // Proceedings of the 5th IEEE International Conference on Data Mining, ICDM'05 (Houston, Texas, November, 27–30, 2005). 2005. P. 8. DOI: 10.1109/ICDM.2005.79.
31. Khayati M., Cudré-Mauroux P., Böhlen M.H. Scalable recovery of missing blocks in time series with high and low cross-correlations // Knowl. Inf. Syst. 2020. Vol. 62, no. 6. P. 2257–2280. DOI: 10.1007/s10115-019-01421-7.
32. Kumar S., Tiwari P., Zymbler M. Internet of Things is a revolutionary approach for future technology enhancement: a review // Journal of Big Data. 2019. Vol. 6. Article 111. DOI: 10.1186/s40537-019-0268-2.
33. Lan L., Shi R., Wang B., et al. A lightweight time series main-memory database for IoT real-time services // Proceedings of the 6th International Conference on Internet of Vehicles, Technologies and Services Toward Smart Cities, IOV 2019 (Kaohsiung, Taiwan, November, 18–21, 2019). P. 220–236. DOI: 10.1007/978-3-030-38651-1_19.
34. Li C., Li B., Bhuiyan M.Z.A., et al. FluteDB: An efficient and scalable in-memory time series database for sensor-cloud // J. Parallel Distributed Comput. 2018. Vol. 122. P. 95–108. DOI: 10.1016/j.jpdc.2018.07.021.
35. Lin T., Kaminski N., Bar-Joseph Z. Alignment and classification of time series gene expression in clinical studies // Bioinf. 2008. Vol. 24, no. 13. P. 147–155. DOI: 10.1093/bioinformatics/btn152.
36. Liu X.-Y., Ren C.-L. Fast subsequence matching under time warping in time-series databases // Proceedings of the International Conference on Machine Learning and Cybernetics, ICMLC 2013 (Tianjin, China, July, 14–17, 2013). P. 1584–1590. DOI: 10.1109/ICMLC.2013.6890855.
37. MacDonald A. PhilDB: the time series database with built-in change logging // PeerJ Comput. Sci. 2016. Vol. 2. P. e52. DOI: 10.7717/peerj-cs.52.
38. Matallah H., Belalem G., Bouamrane K. Evaluation of NoSQL databases: MongoDB, Cassandra, HBase, Redis, Couchbase, OrientDB // Int. J. Softw. Sci. Comput. Intell. 2020. Vol. 12, no. 4. P. 71–91. DOI: 10.4018/IJSSCI.2020100105.
39. Meng J., Yuan J., Hans M., Wu Y. Mining motifs from human motion // Proceedings of the Eurographics 2008 – Short Papers (Crete, Greece, April, 14–18, 2008). Eurographics Association, 2008. P. 71–74. DOI: 10.2312/egs.20081024.
40. Mueen A., Keogh E.J., Zhu Q., Cash S., Westover M.B. Exact Discovery of Time Series Motifs // Proceedings of the SIAM International Conference on Data Mining, SDM 2009 (Sparks, Nevada, USA, April, 30 – May, 2, 2009). SIAM, 2009. P. 473–484. DOI: 10.1137/1.9781611972795.41.
41. Namiot D. Time series databases // Selected Papers of the XVII International Conference on Data Analytics and Management in Data Intensive Domains, DAMDID/RCDL 2015 (Obninsk, Russia, October, 13–16, 2015). P. 132–137. URL: <http://ceur-ws.org/Vol-1536/paper20.pdf> (дата обращения: 16.07.2020).
42. O'Neil P., Cheng E., Gawlick D., O'Neil E. The log-structured merge-tree (LSM-tree) // Acta Informatica. 1996. Vol. 33. P. 351–385.

43. OpenTSDB 3.0 Documentation. URL: <http://opentsdb.net/docs/3x/build/html/> (дата обращения: 27.09.2020).
44. Pelkonen T., Franklin S., Cavallaro P., et al. Gorilla: A fast, scalable, in-memory time series database // Proc. VLDB Endow. 2015. Vol. 8, no. 12. P. 1816–1827. DOI: 10.14778/2824032.2824078.
45. Petersen D., Middleton D. Linear interpolation, extrapolation, and prediction of random space-time fields with a limited domain of measurement // IEEE Transactions on Information Theory. 1965. Vol. 11, no. 1. P. 18–30. DOI: 10.1109/TIT.1965.1053734.
46. Petre I., Boncea R., Radulescu C.Z., et al. A time-series database analysis based on a multi-attribute maturity model // Studies in Informatics and Control. 2019. Vol. 2, no. 2. P. 177–188. DOI: 10.24846/v28i2y201906.
47. Prometheus Documentation. URL: <https://prometheus.io/docs/> (дата обращения: 27.09.2020).
48. Queiroz-Sousa P.O., Salgado A.C. A review on OLAP technologies applied to information networks // ACM Trans. Knowl. Discov. Data. 2020. Vol. 14, no. 1. P. 8:1–8:25. DOI: 10.1145/3370912.
49. Rakthanmanon T., Campana B.J.L., Mueen A., et al. Searching and mining trillions of time series subsequences under Dynamic Time Warping // The 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD'12 (Beijing, China, August, 12–16, 2012). 2012. P. 262–270. DOI: 10.1145/2339530.2339576.
50. Ratanamahatana C.A., Keogh E.J. Three myths about Dynamic Time Warping data mining // Proceedings of the 2005 SIAM International Conference on Data Mining, SDM 2005 (Newport Beach, CA, USA, April, 21–23, 2005). 2005. P. 506–510. DOI: 10.1137/1.9781611972757.50.
51. Rhea S., Wang E., Wong E., et al. LittleTable: A time-series database and its uses // Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017 (Chicago, IL, USA, May, 14–19, 2017). P. 125–138. DOI: 10.1145/3035918.3056102.
52. Riak KV Documentation. URL: <https://docs.riak.com/riak/kv/> (дата обращения: 27.09.2020).
53. Riak TS Documentation. URL: <https://docs.riak.com/riak/ts/> (дата обращения: 27.09.2020).
54. Seltzer M.I. Berkeley DB: A retrospective // IEEE Data Eng. Bull. 2007. Vol. 30, no. 3. P. 21–28. URL: <http://sites.computer.org/debull/A07Sept/seltzer.pdf> (дата обращения: 30.07.2020).
55. Sim H., Khan A., Vazhkudai S.S., Lim S.-H., Butt A.R., Kim Y. An Integrated Indexing and Search Service for Distributed File Systems // IEEE Transactions on Parallel and Distributed Systems. 2020. Vol. 31, no. 10. P. 2375–2391. DOI: 10.1109/TPDS.2020.2990656.
56. Sivasubramanian S. Amazon dynamoDB: a seamlessly scalable non-relational database service // Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data (Scottsdale Arizona, USA, May, 2012). P. 729–730. DOI: 10.1145/2213836.2213945.

57. Shen Z., Zhang Y., Lu J., et al. A novel time series forecasting model with deep learning // *Neurocomputing*. 2020. Vol. 396. P. 302–313. DOI: 10.1016/j.neucom.2018.12.084.
58. Shieh J., Keogh E.J. *iSAX: Indexing and mining terabyte sized time series* // *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Las Vegas, Nevada, USA, August, 24–27, 2008)*. ACM, 2008. P. 623–631. DOI: 10.1145/1401890.1401966.
59. Shvachko K., Kuang H., Radia S., Chansler R. The Hadoop Distributed File System // *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, MSST 10 (May, 2010)*. P. 1–10. DOI: 10.1109/MSST.2010.5496972.
60. Song I.-Y. *Data Warehouse* // *Encyclopedia of Database Systems (2nd ed.)*. Ed. Liu L., Özsu M.T. Springer, 2018. DOI: 10.1007/978-1-4614-8265-9_882.
61. TimescaleDB Documentation. URL: <https://docs.timescale.com/> (дата обращения: 27.09.2020).
62. Torkamani S., Lohweg V. Survey on time series motif discovery // *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* 2017. Vol. 7, no. 2. DOI: 10.1002/widm.1199.
63. Truong C.D., Anh D.T. A survey on time series motif discovery // *Int. J. Bus. Intell. Data Min.* 2019. Vol. 15, no. 2. P. 204–227. DOI: 10.1504/IJBIDM.2019.101266.
64. Tsubouchi Y., Wakisaka A., Hamada K., et al. HeteroTSDB: An extensible time series database for automatically tiering on heterogeneous key-value stores // *Proceedings of the 43rd IEEE Annual Computer Software and Applications Conference, COMPSAC 2019 (Milwaukee, WI, USA, July, 15–19, 2019)*. Vol. 1. P. 264–269. DOI: 10.1109/COMPSAC.2019.00046.
65. Vibhute A., Halder S., Singh P., et al. Decadal variability of tropical Indian Ocean sea surface temperature and its impact on the Indian summer monsoon // *Theoretical and Applied Climatology*. 2020. Vol. 141, no. 1-2. P. 551–566. DOI: 10.1007/s00704-020-03216-1.
66. Winters P.R. Forecasting sales by exponentially weighted moving averages // *Management Science*. 1960. Vol. 6. P. 324–342. DOI: 10.1287/mnsc.6.3.324.
67. Wu J., Wang P., Pan N., et al. KV-Match: A subsequence matching approach supporting normalization and time warping // *Proceedings of the 35th IEEE International Conference on Data Engineering, ICDE 2019 (Macao, China, April, 8–11, 2019)*. P. 866–877. DOI: 10.1109/ICDE.2019.00082.
68. Yang F., Tschetter E., Léauté X., et al. Druid: a real-time analytical data store // *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data, SIGMOD '14 (New York, NY, US, June, 2014)*. P. 157–168. DOI: 10.1145/2588555.2595631.
69. Yang Y., Cao Q., Jiang H. EdgeDB: An efficient time-series database for Edge Computing // *IEEE Access*. 2019. Vol. 7. P. 142295–142307. DOI: 10.1109/ACCESS.2019.2943876.
70. Yankov D., Keogh E.J., Rebbapragada U. Disk aware discord discovery: Finding unusual time series in terabyte sized datasets // *Proceedings of the 7th IEEE International Conference on Data Mining, ICDM 2007 (Omaha, Nebraska, USA, October, 28–31, 2007)*. IEEE Computer Society, 2007. P. 381–390. DOI: 10.1109/ICDM.2007.61.

71. Yeh C.-C.M., Zhu Y., Ulanova L., et al. Time series joins, motifs, discords and shapelets: a unifying view that exploits the matrix profile // *Data Min. Knowl. Discov.* 2018. Vol. 32, no. 1. P. 83–123. DOI: 10.1007/s10618-017-0519-9.
72. Zhang Y.-F., Thorburn P.J., Xiang W., Fitch P. SSIM – A deep learning approach for recovering missing time series sensor data // *IEEE Internet Things Journal.* 2019. Vol. 6, no. 4. P. 6618–6628. DOI: 10.1109/JIOT.2019.2909038.

Иванова Елена Владимировна, к.ф.-м.н., кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

Цымблер Михаил Леонидович, д.ф.-м.н., доцент, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DOI: 10.14529/cmse200406

OVERVIEW OF MODERN TIME SERIES MANAGEMENT SYSTEMS

© 2020 E.I. Ivanova, M.L. Zymbler

South Ural State University (pr. Lenina 76, Chelyabinsk, 454080 Russia)

E-mail: elena.ivanova@susu.ru, mzym@susu.ru

Received: 27.09.2020

A time series is a sequence of chronologically ordered numerical values that reflect some process or phenomenon. Currently, one of the most topical applications related to time series processing are Industry 4.0 and Internet of Things. In these applications, the typical task is to provide intelligent control and predictive maintenance of complex machines and mechanisms that are equipped with various sensors. Such sensors have a high frequency, and in a relatively short time interval produce time series from tens of millions to billions of elements. The data obtained from the sensors is accumulated and mined to make strategic decisions. Time series processing requires specific system software that is different from the existing relational DBMS and NoSQL systems. Time series database systems should provide, on the one hand, efficient operations for adding new atomic values arriving in streaming mode, and on the other hand, efficient mining operations where time series is considered as a whole. The paper discusses the features of time series processing in comparison with data of a relational and non-relational nature, and gives formal definitions of the basic tasks of time series mining. The paper also presents an overview of three most popular modern time series database systems, namely InfluxDB, OpenTSDB, TimescaleDB.

Keywords: time series management and mining, NoSQL, relational DBMS, InfluxDB, OpenTSDB, TimescaleDB.

FOR CITATION

Ivanova E.V., Zymbler M.L. Overview of Modern Time Series Management Systems. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering.* 2020. Vol. 9, no. 4. P. 79–97. (in Russian) DOI: 10.14529/cmse200406.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 3.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

СВЕДЕНИЯ ОБ ИЗДАНИИ

Научный журнал «Вестник ЮУрГУ. Серия «Вычислительная математика и информатика» основан в 2012 году.

Учредитель — Федеральное государственное автономное образовательное учреждение высшего образования «Южно-Уральский государственный университет» (национальный исследовательский университет).

Главный редактор — Л.Б. Соколинский.

Свидетельство о регистрации ПИ ФС77-57377 выдано 24 марта 2014 г. Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций.

Журнал включен в Реферативный журнал и Базы данных ВИНИТИ; индексируется в библиографической базе данных РИНЦ. Журнал размещен в открытом доступе на Всероссийском математическом портале MathNet. Сведения о журнале ежегодно публикуются в международной справочной системе по периодическим и продолжающимся изданиям «Ulrich's Periodicals Directory».

Решением Президиума Высшей аттестационной комиссии Министерства образования и науки Российской Федерации журнал включен в «Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук» по научным специальностям и соответствующим им отраслям науки: 05.13.11 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей (физико-математические науки), 05.13.17 – Теоретические основы информатики (физико-математические науки).

Подписной индекс научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»: 10244, каталог «Пресса России». Периодичность выхода — 4 выпуска в год.

Адрес редакции, издателя: 454080, г. Челябинск, проспект Ленина, 76, Издательский центр ЮУрГУ, каб. 32.

ПРАВИЛА ДЛЯ АВТОРОВ

1. Правила подготовки рукописей и пример оформления статей можно загрузить с сайта серии <http://vestnikvmi.susu.ru>. **Статьи, оформленные без соблюдения правил, к рассмотрению не принимаются.**
2. Адрес редакционной коллегии научного журнала «Вестник ЮУрГУ», серия «Вычислительная математика и информатика»:
Россия 454080, г. Челябинск, пр. им. В.И. Ленина, 76, ЮУрГУ, кафедра СП,
ответственному секретарю Цымблеру М.Л.
3. Адрес электронной почты редакции: vestnikvmi@susu.ru
4. **Плата с авторов за публикацию рукописей не взимается, и гонорары авторам не выплачиваются.**

ВЕСТНИК
ЮЖНО-УРАЛЬСКОГО
ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
Серия
«ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАТИКА»
Том 9, № 4
2020

16+

Техн. редактор *А.В. Миних*

Издательский центр Южно-Уральского государственного университета

Подписано в печать 30.11.2020. Дата выхода в свет 11.12.2020. Формат 60×84 1/8. Печать цифровая.
Усл. печ. л. 11,62. Тираж 500 экз. Заказ 335/421. Цена свободная.

Отпечатано в типографии Издательского центра ЮУрГУ.
454080, г. Челябинск, проспект Ленина, 76.