

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

ВЕСТНИК

ЮЖНО-УРАЛЬСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА

BULLETIN

OF THE SOUTH URAL
STATE UNIVERSITY

СЕРИЯ

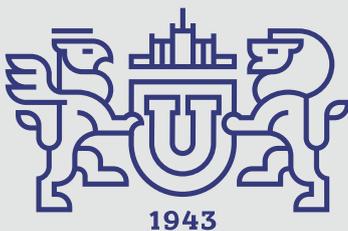
**ВЫЧИСЛИТЕЛЬНАЯ
МАТЕМАТИКА
И ИНФОРМАТИКА**

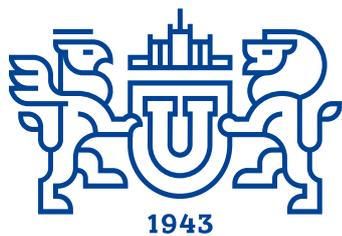
2024, том 13, № 4

SERIES

**COMPUTATIONAL
MATHEMATICS
AND SOFTWARE ENGINEERING**

2024, volume 13, no. 4





ВЕСТНИК

ЮЖНО-УРАЛЬСКОГО
ГОСУДАРСТВЕННОГО
УНИВЕРСИТЕТА

2024
Т. 13, № 4

ISSN 2305-9052

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*
М.Л. Цымблер, д.ф.-м.н., доц., *зам. гл. редактора*
Я.А. Краева, к.ф.-м.н., *отв. секретарь*
А.И. Гоглачев, *техн. редактор*

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
И.М. Куликов, д.ф.-м.н.
Д. Маллманн, PhD, профессор (Германия)
А.В. Панюков, д.ф.-м.н., профессор
Р. Продан, PhD, профессор (Австрия)
Г.И. Радченко, к.ф.-м.н., доцент (Австрия)
В.Н. Ушаков, д.ф.-м.н., чл.-кор. РАН
М.Ю. Хачай, д.ф.-м.н., чл.-кор. РАН
А. Черных, PhD, профессор (Мексика)
П. Шумяцкий, PhD, профессор (Бразилия)



BULLETIN

OF THE SOUTH URAL STATE UNIVERSITY **2024**
vol. 13, no. 4

SERIES

“COMPUTATIONAL
MATHEMATICS AND SOFTWARE
ENGINEERING”

ISSN 2305-9052

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)
M.L. Zymbler, South Ural State University (Chelyabinsk, Russia)
Ya.A. Kraeva, South Ural State University (Chelyabinsk, Russia)
A.I. Goglavchev, 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)
I.M. Kulikov, Institute of Computational Mathematics and Mathematical Geophysics, Siberian Branch of RAS (Novosibirsk, Russia)
D. Mallmann, Julich Supercomputing Centre (Germany)
A.V. Panyukov, South Ural State University (Chelyabinsk, Russia)
R. Prodan, Alpen-Adria-Universität Klagenfurt (Austria)
G.I. Radchenko, Silicon Austria Labs (Graz, Austria)
P. Shumyatsky, University of Brasilia (Brazil)
A. Tchernykh, CICESE Research Center (Mexico)
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

О ЧИСЛЕННОМ МЕТОДЕ ДЛЯ ЗАДАЧИ СТОКСА С ГРАНИЧНЫМИ УСЛОВИЯМИ НЕЙМАНА В НЕВЫПУКЛОЙ ОБЛАСТИ А.В. Рукавишников	5
PRECISE LOCALIZATION OF PDF417 CODE BASED ON FAST HOUGH TRANSFORM D.G. Mitrofanov, P.K. Zlobin, J.A. Shemiakina, P.V. Bezmaternykh	19
НЕЙРОСЕТЕВОЙ МЕТОД ОБНАРУЖЕНИЯ АНОМАЛИЙ В МНОГОМЕРНЫХ ПОТОКОВЫХ ВРЕМЕННЫХ РЯДАХ Я.А. Краева	35
ОБ ОДНОЙ ФУНКЦИИ ПОТЕРЬ ДЛЯ ОБУЧЕНИЯ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ ВОССТАНОВЛЕНИЯ ВРЕМЕННЫХ РЯДОВ А.А. Юртин	53

Contents

ON NUMERICAL METHOD FOR THE STOKES PROBLEM WITH NEUMANN BOUNDARY CONDITIONS IN NON-CONVEX DOMAIN A.V. Rukavishnikov	5
PRECISE LOCALIZATION OF PDF417 CODE BASED ON FAST HOUGH TRANSFORM D.G. Mitrofanov, P.K. Zlobin, J.A. Shemiakina, P.V. Bezmaternykh	19
DEEP LEARNING METHOD FOR ANOMALY DETECTION IN STREAMING MULTIVARIATE TIME SERIES Ya.A. Kraeva	35
TOWARDS A LOSS FUNCTION FOR TRAINING NEURAL NETWORK MODELS OF TIME SERIES IMPUTATION A.A. Yurtin	53



This issue is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

О ЧИСЛЕННОМ МЕТОДЕ ДЛЯ ЗАДАЧИ СТОКСА С ГРАНИЧНЫМИ УСЛОВИЯМИ НЕЙМАНА В НЕВЫПУКЛОЙ ОБЛАСТИ

© 2024 А.В. Рукавишников

Хабаровское отделение Института прикладной математики ДВО РАН

(680030 Хабаровск, ул. Серышева, д. 60)

E-mail: 78321a@mail.ru

Поступила в редакцию: 15.11.2024

Рассмотрена задача Стокса с граничными условиями Неймана с входящим углом на границе двумерной области. Введено понятие R_ν -обобщенного решения в множествах весовых пространств Соболева. Построен весовой метод конечных элементов на равномерной сетке, основанный на конечно-элементной паре Тейлора–Худа второго порядка и введения в базис весовой функции в некоторых степенях ν^* и μ^* для компонент поля скоростей и скалярной функции давления соответственно. Весовая функция в области совпадает с функцией расстояния от точки до вершины входящего угла в некоторой δ -окрестности и константе δ вне ее. Проведены численные эксперименты в невыпуклой области. Получен порядок сходимости приближенного решения к точному решению задачи, независящий от величины входящего угла и превышающий порядок сходимости для классического МКЭ. Результат о сходимости достигается без геометрического сгущения сетки в окрестности точки сингулярности. Проведена серия численных экспериментов для различных величин входящего угла и найдена область подходящих свободных параметров предложенного подхода. Для любой точки построенной области достигается оптимальный, с точки зрения сходимости, результат. Область выбора подходящих свободных параметров отличается от области для рассматриваемой задачи с граничными условиями Дирихле.

Ключевые слова: угловая сингулярность, задача Стокса с граничными условиями Неймана, R_ν -обобщенное решение, весовой МКЭ.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Рукавишников А.В. О численном методе для задачи Стокса с граничными условиями Неймана в невыпуклой области // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 4. С. 5–18. DOI: 10.14529/cmse240401.

Введение

Многие математические модели природных процессов описываются краевыми задачами с помощью систем уравнений с частными производными, имеющие ту или иную особенность. Особенность решения таких систем в двумерной замкнутой области Ω может быть связана с вырождением исходных данных или наличием входящих углов на границе области. Будем говорить, что краевая задача имеет сильную сингулярность, если ее решение не принадлежит пространству Соболева $W_2^1(\Omega)$. Другими словами, интеграл Дирихле от ее решения расходится. В случае, когда решение принадлежит пространству Соболева $W_2^1(\Omega)$, но не принадлежит пространству Соболева $W_2^2(\Omega)$ краевая задача называется задачей со слабой сингулярностью или просто с сингулярностью. Например, обобщенное решение краевой эллиптической задачи с граничными условиями Дирихле в двумерной области, содержащей входящий угол на границе, т.е. угол из диапазона $(\pi, 2\pi]$, принадлежит лишь пространству $W_2^{1+\beta-\varepsilon}(\Omega)$, $\beta \in [0.25, 1)$, где ε — произвольное положительное действительное число. Поэтому приближенное решение, полученное классическими методами

конечных разностей или конечных элементов, сходится к точному решению не быстрее чем со скоростью $\mathcal{O}(h^\beta)$ [1].

Для краевых задач с сингулярностью существуют различные численные подходы, основанные на выделении сингулярных компонент обобщенного решения и сгущения сетки к точке сингулярности (см., например, [2][3]). Эти методы существенно усложняют схему метода конечных элементов (МКЭ), что влияет на скорость вычислительного процесса и точность результата.

В [4] для эллиптических задач с граничными условиями Дирихле было предложено определять решение как R_ν -обобщенное в весовых пространствах Соболева. Теоретические результаты, связанные со существованием, единственностью и дифференциальными свойствами R_ν -обобщенного решения различных задач изучены в [5-9]. На основе этого подхода были созданы численные методы, позволяющие получать преимущество по порядку относительно шага сетки над классическими подходами (см., например, [10-12]).

В представленной работе рассмотрим задачу Стокса с граничными условиями Неймана, определим R_ν -обобщенное решение задачи в весовых множествах. Теоретический анализ обобщенного решения задачи Стокса (Навье—Стокса) с граничными условиями Неймана выполнен в работах [13][16]. В [13] изучена регулярность решения задачи в пространствах Соболева, существование и единственность слабых решений Навье—Стокса. В [14] показано как свойства оператора Стокса приводят к локальным или глобальным решениям для малых начальных данных. В [15] проведено теоретическое исследование решения нестационарной задачи, которая является линеаризацией модельной задачи со свободной границей. Доказаны оценки решения в нормах пространств Лебега и получено преимущество по сравнению со случаем граничного условия без проскальзывания. В [16] определяется связь свойств оператора Стокса—Неймана с оператором Стокса—Дирихле, в частности, связь их собственных значений. Здесь предложим численный метод решения задачи, основанный на определении R_ν -обобщенного решения и введении в базис весовой функции в некоторых степенях. Проведен ряд численных экспериментов задачи в невыпуклой области. Экспериментально получим, что порядок сходимости приближенного решения к точному решению задачи не зависит от величины входящего угла.

Статья организована следующим образом. Раздел [1] содержит необходимые обозначения, определения пространств и множеств. Раздел [2] посвящен определению классического и R_ν -обобщенного решений задачи Стокса с граничными условиями Неймана. В разделе [3] построена схема весового метода конечных элементов. В разделе [4] представлены результаты численных экспериментов и определены области выбора свободных параметров подхода. В заключении приведены выводы и намечены направления дальнейших исследований.

1. Необходимые обозначения. Определения пространств и множеств

Область Ω есть невыпуклый многоугольник с границей Γ , содержащий входящий угол на границе в двумерном евклидовом пространстве \mathcal{R}^2 , где $\mathbf{x} = (x_1, x_2)$ его элемент, $d\mathbf{x} = dx_1 dx_2$ его мера и $\|\mathbf{x}\| = (x_1^2 + x_2^2)^{1/2}$ его норма. Пусть вершина входящего угла совпадает с началом координат $\mathcal{O} = (0, 0)$. Вершину входящего угла будем называть точкой сингулярности. Пусть $\bar{\Omega} = \Omega \cup \Gamma$ — замыкание области Ω .

Введем необходимые обозначения. Пусть заданы векторные поля $\mathbf{a} = (a_1, a_2)$, $\mathbf{c} = (c_1, c_2)$, $\mathbf{d} = (d_1, d_2)$ и скалярное поле b , тогда

$$\Delta \mathbf{a} = \left(\frac{\partial^2 a_1}{\partial x_1^2} + \frac{\partial^2 a_1}{\partial x_2^2}, \frac{\partial^2 a_2}{\partial x_1^2} + \frac{\partial^2 a_2}{\partial x_2^2} \right)^T - \text{оператор Лапласа поля } \mathbf{a},$$

$$\operatorname{div} \mathbf{a} = \frac{\partial a_1}{\partial x_1} + \frac{\partial a_2}{\partial x_2} - \text{дивергенция поля } \mathbf{a}, \nabla b = \left(\frac{\partial b}{\partial x_1}, \frac{\partial b}{\partial x_2} \right)^T - \text{градиент поля } b.$$

Кроме того,

$$\mathbf{a} \cdot \nabla b = \sum_{i=1}^2 a_i \frac{\partial b}{\partial x_i}, \quad \nabla \mathbf{a} : \nabla \mathbf{c} = \sum_{i,j=1}^2 \frac{\partial a_i}{\partial x_j} \frac{\partial c_i}{\partial x_j},$$

$$\nabla \mathbf{a} : (\mathbf{c} \nabla b) = \sum_{i,j=1}^2 \frac{\partial a_i}{\partial x_j} c_i \frac{\partial b}{\partial x_j}, \quad (\mathbf{a} \nabla b) : (\mathbf{c} \nabla b) = \sum_{i,j=1}^2 a_i \frac{\partial b}{\partial x_j} c_i \frac{\partial b}{\partial x_j},$$

$$[\mathbf{a}(\mathbf{d} \cdot \nabla b)] : \mathbf{c} = \sum_{i,j=1}^2 a_j \left(d_i \frac{\partial b}{\partial x_i} \right) c_j, \quad (\nabla \mathbf{a} \mathbf{d}) : \mathbf{c} = \sum_{i,j=1}^2 \frac{\partial a_j}{\partial x_i} d_i c_j.$$

Через Ω_δ обозначим область, образованную пересечением круга с центром в начале координат \mathcal{O} и радиусом δ с $\bar{\Omega}$, т.е. $\Omega_\delta = \{\mathbf{x} \in \bar{\Omega} : \|\mathbf{x}\| \leq \delta \ll 1, \delta > 0\}$. Определим функцию $\rho(\mathbf{x})$ в $\bar{\Omega}$, которую назовем весовой, имеющую следующий вид:

$$\rho(\mathbf{x}) = \begin{cases} \|\mathbf{x}\|, & \text{если } \mathbf{x} \in \Omega_\delta, \\ \delta, & \text{если } \mathbf{x} \in \bar{\Omega} \setminus \Omega_\delta. \end{cases}$$

Определим необходимые нам весовые пространства и множества обобщенных функций. Через $L_{2,\alpha}(\Omega)$ обозначим весовое пространство функций $v(\mathbf{x})$ с ограниченной нормой

$$\|v\|_{L_{2,\alpha}(\Omega)} = \left(\int_{\Omega} \rho^{2\alpha}(\mathbf{x}) v^2(\mathbf{x}) d\mathbf{x} \right)^{1/2}. \quad (1)$$

Причем, если $\alpha = 0$, то $L_{2,0}(\Omega) \equiv L_2(\Omega)$. Обозначим через $W_{2,\alpha}^1(\Omega)$ пространство функций $v(\mathbf{x})$ таких, что $v(\mathbf{x}) \in W_{2,\alpha}^1(\Omega)$, если ограничена интегральная норма

$$\|v\|_{W_{2,\alpha}^1(\Omega)} = \left(\|v\|_{L_{2,\alpha}(\Omega)}^2 + \sum_{|l|=1} \|\rho^\alpha |D^l v(\mathbf{x})|\|_{L_2(\Omega)}^2 \right)^{1/2}, \quad (2)$$

где $D^l v(\mathbf{x}) = \frac{\partial^{|l|} v(\mathbf{x})}{\partial x_1^{l_1} \partial x_2^{l_2}}$ — l -я обобщенная производная функции $v(\mathbf{x})$: $|l| = l_1 + l_2$, l_i — неотрицательные целые числа, $i = 1, 2$.

Введем следующие условия для функции $v(\mathbf{x})$:

$$|v(\mathbf{x})| \leq C_1 \delta^{\alpha-\tau} \rho^{\tau-\alpha}(\mathbf{x}), \quad \mathbf{x} \in \Omega_\delta, \quad (3)$$

$$|D^1 v(\mathbf{x})| \leq C_1 \delta^{\alpha-\tau} \rho^{\tau-\alpha-1}(\mathbf{x}), \quad \mathbf{x} \in \Omega_\delta, \quad (4)$$

$$\int_{\Omega \setminus \Omega_\delta} \rho^{2\alpha}(\mathbf{x}) v^2(\mathbf{x}) d\mathbf{x} \geq (C_2)^2 > 0, \quad (5)$$

где C_1 — положительная константа, τ — малый положительный параметр, независимый от δ, α и $v(\mathbf{x})$. Далее определим множества в введенных весовых пространствах. Через

$L_{2,\alpha}(\Omega, \delta)$ обозначим множество функций $v(\mathbf{x})$ из пространства $L_{2,\alpha}(\Omega)$, удовлетворяющих условиям (3) и (5) с ограниченной нормой (1). Определим множество $L_{2,\alpha}^0(\Omega, \delta)$ функций $v(\mathbf{x})$ таких, что $v(\mathbf{x}) \in L_{2,\alpha}^0(\Omega, \delta)$, если $v(\mathbf{x}) \in L_{2,\alpha}(\Omega, \delta)$ и $\int_{\Omega} \rho^{\alpha}(\mathbf{x})v(\mathbf{x})d\mathbf{x} = 0$ с ограниченной нормой (1). Если $\alpha = 0$, то $L_2^0 := L_{2,0}^0(\Omega, \delta)$. Через $W_{2,\alpha}^1(\Omega, \delta)$ обозначим множество функций $v(\mathbf{x})$ из весового пространства $W_{2,\alpha}^1(\Omega)$, удовлетворяющих условиям (3)–(5) с ограниченной нормой (2). Если $\alpha = 0$, то $W_2^1 := W_{2,0}^1(\Omega, \delta)$.

Для векторных полей, пространства и множества будем выделять жирным шрифтом, например, $\mathbf{v} = (v_1, v_2) : \mathbf{v} \in \mathbf{L}_{2,\alpha}(\Omega, \delta)$, если $v_i \in L_{2,\alpha}(\Omega, \delta), i = 1, 2$, с ограниченной нормой $\|\mathbf{v}\|_{\mathbf{L}_{2,\alpha}(\Omega)} = \left(\sum_{i=1}^2 \|v_i\|_{L_{2,\alpha}(\Omega)}^2 \right)^{1/2}$.

2. Определение классического и R_{ν} -обобщенного решений задачи Стокса с граничными условиями Неймана. Связь R_{ν} -обобщенного и обобщенного решений

Задача Стокса с граничными условиями Неймана состоит в том, что, зная функции $\mathbf{f} = (f_1, f_2)$ в Ω и \mathbf{r} на Γ необходимо найти поля скоростей $\mathbf{u} = (u_1, u_2)$ и давления p , которые удовлетворяют системе дифференциальных уравнений и граничных условий:

$$-\Delta \mathbf{u} + \nabla p = \mathbf{f} \quad \text{в} \quad \Omega, \quad (6)$$

$$\operatorname{div} \mathbf{u} = 0 \quad \text{в} \quad \Omega, \quad (7)$$

$$\left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - p \right) \mathbf{n} = \mathbf{r} \quad \text{на} \quad \Gamma, \quad (8)$$

где $\mathbf{n} = (n_1, n_2)$ единичный вектор внешней нормали к Γ .

Определим билинейные и линейную формы:

$$a(\mathbf{w}, \mathbf{v}) = \int_{\Omega} \nabla \mathbf{w} : \nabla (\rho^{2\nu} \mathbf{v}) dx, \quad (9)$$

$$b_1(\mathbf{w}, q) = - \int_{\Omega} q \operatorname{div} (\rho^{2\nu} \mathbf{w}) dx, \quad b_2(\mathbf{w}, q) = - \int_{\Omega} (\rho^{2\nu} q) \operatorname{div} \mathbf{w} dx, \quad (10)$$

$$l(\mathbf{v}) = \int_{\Omega} \mathbf{f} \cdot (\rho^{2\nu} \mathbf{v}) dx + \int_{\Gamma} \mathbf{r} \cdot (\rho^{2\nu} \mathbf{v}) ds,$$

где ds — дифференциал по Γ .

Пусть

$$\mathbf{f} \in \mathbf{L}_{2,\beta}(\Omega), \mathbf{r} \in \mathbf{L}_{2,\nu}(\Gamma), 0 \leq \beta \leq \nu. \quad (11)$$

Введем понятие R_{ν} -обобщенного решения задачи Стокса (6)–(8) с граничными условиями Неймана в весовых множествах.

Определение 1 (R_{ν} -обобщенного решения). Пару функций $(\mathbf{u}_{\nu}, p_{\nu}) \in \mathbf{W}_{2,\nu}^1(\Omega, \delta) \times L_{2,\nu}^0(\Omega, \delta)$ назовем R_{ν} -обобщенным решением задачи Стокса (6)–(8), если для всех пар функций $(\mathbf{v}, q) \in \mathbf{W}_{2,\nu}^1(\Omega, \delta) \times L_{2,\nu}^0(\Omega, \delta)$ справедливы интегральные тождества:

$$a(\mathbf{u}_{\nu}, \mathbf{v}) + b_1(\mathbf{v}, p_{\nu}) = l(\mathbf{v}), \quad (12)$$

$$b_2(\mathbf{u}_{\nu}, q) = 0, \quad (13)$$

где \mathbf{f} и \mathbf{r} удовлетворяют условиям (11) и $\mathbf{u}_\nu = (u_{1,\nu}, u_{2,\nu})$.

Определим связь между R_ν -обобщенным и обобщенным решениями задачи Стокса (6)–(8).

Во-первых, по аналогии с Леммой 4 (17) имеет место следующее утверждение: $z \in W_{2,\nu}^1(\Omega, \delta)$ тогда и только тогда, когда $\rho^\nu z \in W_2^1$. По определению нормы (1): $q \in L_{2,\nu}^0(\Omega, \delta)$ тогда и только тогда, когда $\rho^\nu q \in L_2^0$.

Во-вторых, справедливы следующие представления для $\mathbf{v} = (v_1, v_2)$ и $\mathbf{w} = (w_1, w_2)$:

$$\frac{\partial(\rho^{2\nu} v_i)}{\partial x_j} = \rho^\nu \frac{\partial(\rho^\nu v_i)}{\partial x_j} + (\rho^\nu v_i) \frac{\partial \rho^\nu}{\partial x_j},$$

$$\frac{\partial w_i}{\partial x_j} = \rho^{-\nu} \frac{\partial(\rho^\nu w_i)}{\partial x_j} - \rho^{-\nu} w_i \frac{\partial \rho^\nu}{\partial x_j}, \quad i, j = 1, 2.$$

Согласно этому, билинейная форма $a(\mathbf{w}, \mathbf{v})$ в (9) может быть представлена в следующем виде:

$$a(\mathbf{w}, \mathbf{v}) = \tilde{a}(\mathbf{w}, \mathbf{v}) + a_1(\mathbf{w}, \mathbf{v}),$$

где

$$\tilde{a}(\mathbf{w}, \mathbf{v}) = \int_{\Omega} \nabla(\rho^\nu \mathbf{w}) : \nabla(\rho^\nu \mathbf{v}) dx,$$

$$a_1(\mathbf{w}, \mathbf{v}) = \int_{\Omega_s} \left[\nabla(\rho^\nu \mathbf{w}) : (\mathbf{v} \nabla \rho^\nu) - (\mathbf{w} \nabla \rho^\nu) : \nabla(\rho^\nu \mathbf{v}) - (\mathbf{w} \nabla \rho^\nu) : (\mathbf{v} \nabla \rho^\nu) \right] dx.$$

Билинейные формы $b_1(\mathbf{w}, q)$ и $b_2(\mathbf{w}, q)$ в (8) могут быть представлены в таком виде:

$$b_1(\mathbf{w}, q) = \tilde{b}(\mathbf{w}, q) + d(\mathbf{w}, q) \quad \text{и} \quad b_2(\mathbf{w}, q) = \tilde{b}(\mathbf{w}, q) - d(\mathbf{w}, q),$$

где

$$\tilde{b}(\mathbf{w}, q) = - \int_{\Omega} (\rho^\nu q) \operatorname{div}(\rho^\nu \mathbf{w}) dx, \quad d(\mathbf{w}, q) = - \int_{\Omega_s} (\rho^\nu q) (\mathbf{w} \cdot \nabla \rho^\nu) dx.$$

Кроме того,

$$\tilde{l}(\tilde{\mathbf{v}}) = \int_{\Omega} \tilde{\mathbf{f}} \cdot \tilde{\mathbf{v}} dx + \int_{\Gamma} \tilde{\mathbf{r}} \cdot \tilde{\mathbf{v}} ds = l(\mathbf{v}),$$

где $\tilde{\mathbf{f}} = \rho^\nu \mathbf{f}$, $\tilde{\mathbf{r}} = \rho^\nu \mathbf{r}$ и $\tilde{\mathbf{v}} = \rho^\nu \mathbf{v}$.

На основании вышеизложенного, определим обобщенное решение задачи (6)–(8).

Определение 2 (обобщенного решения). Пару функций $(\tilde{\mathbf{u}}, \tilde{p}) \in \mathbf{W}_2^1 \times L_2^0$ назовем обобщенным решением задачи Стокса (6)–(8), если для всех пар функций $(\tilde{\mathbf{v}}, \tilde{q}) \in \mathbf{W}_2^1 \times L_2^0$ справедливы интегральные тождества:

$$\tilde{a}(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}) + \tilde{b}(\tilde{\mathbf{v}}, \tilde{p}) = \tilde{l}(\tilde{\mathbf{v}}), \quad (14)$$

$$\tilde{b}(\tilde{\mathbf{u}}, \tilde{q}) = 0, \quad (15)$$

где $\tilde{\mathbf{u}} = (\tilde{u}_1, \tilde{u}_2)$.

Замечание 1. Вариационная постановка задачи (12), (13) при определении R_ν -обобщенного решения задачи (6)–(8), в отличие от вариационной задачи (14), (15) при определении обобщенного решения задачи (6)–(8), является несимметричной ввиду наличия формы $d(\cdot, \cdot)$, определенной в δ -окрестности точки сингулярности.

Замечание 2. Отличие определения решений имеет место в δ -окрестности входящего угла.

Замечание 3. Покажем, что приближенное решение, полученное на основании определенного R_ν -обобщенного решения имеет порядок сходимости к точному решению задачи, независящий от величины входящего угла, в отличие от приближенного обобщенного решения, порядок сходимости которого уменьшается с увеличением величины входящего угла. Результат достигается благодаря качественному учету погрешности в Ω_δ . При этом погрешность не распространяется из Ω_δ во внутреннюю часть расчетной области. Результат достигается без сгущения сетки к точке сингулярности.

3. Схема весового метода конечных элементов

Построим схему весового метода конечных элементов для задачи Стокса (6)–(8), основанную на определении R_ν -обобщенного решения. Для этой цели проведем квазиравномерную триангуляцию T_h области $\bar{\Omega}$ (1). Пусть L_s — произвольный треугольник триангуляции со стороной порядка h , его назовем конечным элементом. Обозначим через $\Omega_h = \bigcup_{L_s \in T_h} L_s$ объединение всех конечных элементов. Для компонент вектора скоростей, в качестве узлов аппроксимации, будем использовать вершины и середины сторон L_s . Их множество обозначим через R^{vel} , а для скалярной функции давления вершины конечных элементов. Их множество обозначим через R^{pres} и определим конечно-элементную пару Тейлора–Худа 2-го порядка (см., например, (18)) $\mathbf{X}_h \times Z_h$, где $\mathbf{X}_h = X_h \times X_h$ и $X_h = \{v^h \in C(\Omega) : v^h|_{L_s} \in P_2(L_s) \forall L_s \in T_h\}$; $Z_h = \{z^h \in C(\Omega) : z^h|_{L_s} \in P_1(L_s) \forall L_s \in T_h\}$.

На основании вышеизложенного, представим особые базисные функции и построим схему весового МКЭ для приближенного решения задачи Стокса (6)–(8). Каждому узлу $N_k \in R^{vel}$ и $M_l \in R^{pres}$ соответствует базисная функция

$$\theta_k(\mathbf{x}) = \rho^{-\nu^*}(\mathbf{x})\varphi_k(\mathbf{x}) \text{ и } \chi_l(\mathbf{x}) = \rho^{-\mu^*}(\mathbf{x})\psi_k(\mathbf{x}), \quad k, l = 0, 1, \dots,$$

где $\varphi_k(\mathbf{x}) \in X_h$, $\varphi_k(N_i) = \delta_{ki}$ и $\psi_l(\mathbf{x}) \in Z_h$, $\psi_l(M_i) = \delta_{li}$, $i, k, l = 0, 1, \dots$, δ_{ij} — символ Кронекера, ν^* и μ^* свободные (положительные) параметры подхода.

Через V_h и Q_h обозначим множества для компонент вектора скоростей и функции давления как линейные оболочки систем базисных функций $\{\theta_k\}_k$ и $\{\chi_l\}_l$ соответственно

$$u_{\nu,1}^h(\mathbf{x}) = \sum_k d_{2k}\theta_k(\mathbf{x}), \quad u_{\nu,2}^h(\mathbf{x}) = \sum_k d_{2k+1}\theta_k(\mathbf{x}), \quad p_\nu^h(\mathbf{x}) = \sum_l e_l\chi_l(\mathbf{x}), \quad (16)$$

где $d_j = \rho^{\nu^*}(N_{[j/2]})\tilde{d}_j$, $e_i = \rho^{\mu^*}(M_i)\tilde{e}_i$. Коэффициенты d_j и e_i в (16) определяются как решение системы линейных алгебраических уравнений (см. (19)) ниже. Пусть $\mathbf{V}_h = V_h \times V_h$ и $\mathbf{V}_h \subset \mathbf{W}_{2,\nu}^1(\Omega_h, \delta)$ и $Q_h \subset L_{2,\nu}^0(\Omega_h, \delta)$.

Введем понятие приближенного R_ν -обобщенного решения задачи Стокса (6)–(8).

Определение 3. Приближенным R_ν -обобщенным решением задачи Стокса (6)–(8) по весовому методу конечных элементов назовем пару функций $(\mathbf{u}_\nu^h, p_\nu^h) \in \mathbf{V}_h \times Q_h$ такую, что для произвольной пары функций $(\mathbf{v}^h, q^h) \in \mathbf{V}_h \times Q_h$ справедливы интегральные тождества:

$$a(\mathbf{u}_\nu^h, \mathbf{v}^h) + b_1(\mathbf{v}^h, p_\nu^h) = l(\mathbf{v}^h), \quad (17)$$

$$b_2(\mathbf{u}_\nu^h, q^h) = 0, \quad (18)$$

где \mathbf{f} и \mathbf{g} удовлетворяют условиям (11) и $\mathbf{u}_\nu^h = (u_{1,\nu}^h, u_{2,\nu}^h)$.

Задача (17), (18) порождает систему линейных алгебраических уравнений

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{B}_2^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{y} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \mathbf{0} \end{bmatrix}, \quad (19)$$

где $\mathbf{y} = (d_0, d_2, d_4, \dots, d_1, d_3, d_5, \dots)^T$, $\mathbf{z} = (e_0, e_1, e_2, \dots)^T$, \mathbf{F} — вектор-столбец значений линейной формы $l(\theta_k)$.

Итерационный процесс с переобуславливанием матрицы системы вида (19) построен и реализован (см., например, (11)). В следующем параграфе проведем необходимые численные эксперименты и покажем преимущество предложенного нами весового МКЭ.

4. Численные эксперименты. Определение области выбора свободных параметров подхода

В этом разделе представим результаты численных экспериментов решения задачи Стокса (6)–(8) с граничными условиями Неймана для двух конфигураций невыпуклой области (см. рис. 1).

Таблица 1. Зависимость погрешностей приближенного R_ν -обобщенного решения, используя весовой МКЭ, от шага сетки h (по строкам) и степеней ν и ν^* ($\mu^* = \nu^*$) в норме $\|\mathbf{u}_\nu - \mathbf{u}_\nu^h\|_{\mathbf{w}_{2,\nu}^1(\Omega)}$, $\omega = \frac{3\pi}{2}$

(δ, ν)	(0.012, 2.0)	(0.0125, 1.8)
h	$\nu^* = 0.375$	$\nu^* = 0.35$
0.01	1.201e-5	3.051e-5
0.005	6.004e-6	1.525e-5
0.0025	2.998e-6	7.615e-6

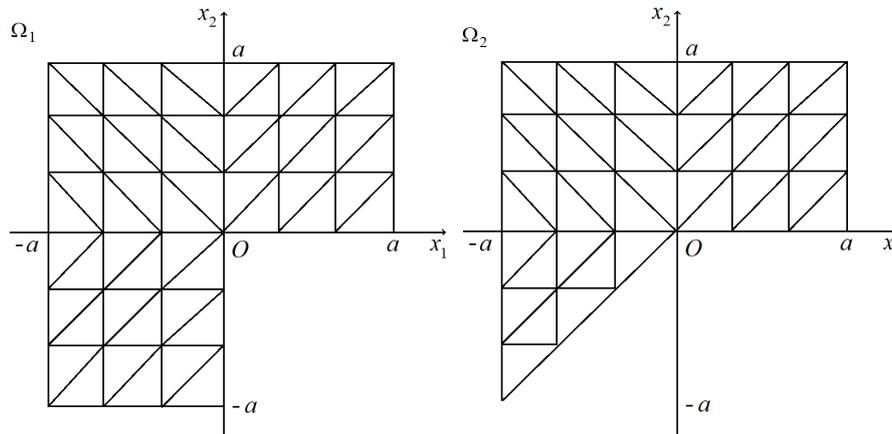


Рис. 1. Триангуляции областей Ω_1 и Ω_2

Таблица 2. Зависимость погрешностей приближенного обобщенного решения, используя классический МКЭ, от шага сетки h (по столбцам) ($\nu = \mu^* = \nu^* = 0, \delta = 1$) в норме $\|\tilde{\mathbf{u}} - \tilde{\mathbf{u}}^h\|_{\mathbf{W}_{2,0}^1(\Omega)}$, $\omega = \frac{3\pi}{2}$

h	0.01	0.005	0.0025
$\ \tilde{\mathbf{u}} - \tilde{\mathbf{u}}^h\ _{\mathbf{W}_{2,0}^1(\Omega)}$	1.647e-1	1.140e-1	7.822e-2

Покажем преимущество весового подхода над классическим МКЭ. Представим точное решение задачи [6]–[8] в полярных координатах (r, φ) в следующей форме:

$$u_1(r, \varphi) = r^{\beta_k} \chi_{1,k}(\varphi), \quad u_2(r, \varphi) = r^{\beta_k} \chi_{2,k}(\varphi), \quad p(r, \varphi) = r^{\beta_k - 1} \gamma_k(\varphi),$$

$$\begin{pmatrix} \chi_{1,k}(\varphi) \\ \chi_{2,k}(\varphi) \end{pmatrix} = \begin{pmatrix} (1 + \beta_k)\Lambda_k(\varphi) & \Lambda'_k(\varphi) \\ \Lambda'_k(\varphi) & -(1 + \beta_k)\Lambda_k(\varphi) \end{pmatrix} \begin{pmatrix} \sin \varphi \\ \cos \varphi \end{pmatrix},$$

$$\gamma_k(\varphi) = \frac{(1 + \beta_k)^2 \Lambda'_k(\varphi) + \Lambda_k'''(\varphi)}{\beta_k - 1},$$

где

$$\Lambda_k(\varphi) = \cos((1 - \beta_k)\varphi) - \cos((1 + \beta_k)\varphi) +$$

$$+(1 + \beta_k)^{-1} \sin((1 + \beta_k)\varphi) \cos(\omega_k \beta_k) - (\beta_k - 1)^{-1} \sin((\beta_k - 1)\varphi) \cos(\omega_k \beta_k), \quad k = 1, 2,$$

где $\beta_k = \min\{\beta : \sin(\beta \omega_k) + \beta \sin \omega_k = 0 \text{ и } \beta > 0\}$. Тогда для $\omega_1 = \frac{3\pi}{2}$, имеем $\beta_1 \approx 0.54448$, а для $\omega_2 = \frac{5\pi}{4} - \beta_2 \approx 0.67358$. Таким образом, величина β_k определяет асимптотику поведения решения в окрестности точки сингулярности. При этом $u_i \notin W_2^2(\Omega), i = 1, 2$, и $p \notin W_2^1(\Omega)$ — типичная ситуация для поведения решения в невыпуклой области с входящим углом на границе Γ . Полагаем, что $a = 1, h = \frac{2}{N}$, где N — количество отрезков разбиения по каждому направлению, h — шаг сетки.

Величину погрешности приближенного R_ν -обобщенного решения поля скоростей \mathbf{u}_ν^h и приближенного обобщенного решения ($\nu = \nu^* = \mu^* = 0, \delta = 1$) поля скоростей $\tilde{\mathbf{u}}^h$ определяем как модуль разности между точным и приближенным решениями в нормах пространств $\mathbf{W}_{2,\nu}^1(\Omega)$ и $\mathbf{W}_{2,0}^1(\Omega)$ соответственно.

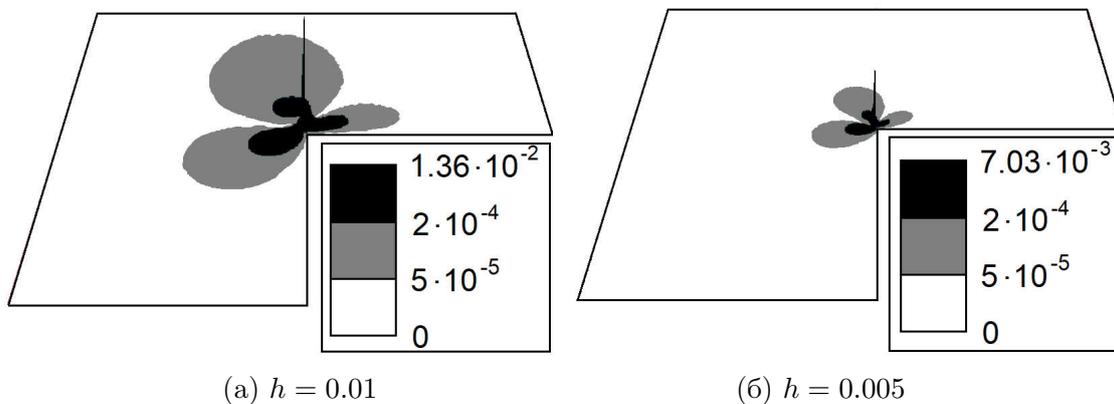


Рис. 2. Абсолютная погрешность первой компоненты поля скоростей приближенного R_ν -обобщенного решения ($\nu = 2.0, \nu^* = \mu^* = 0.375, \delta = 0.012$), $\omega = \frac{3\pi}{2}$

Результаты представлены в табл. 1-4 и рис. 2 и 3. Параметры ν, ν^*, μ^* и δ , представленного весового МКЭ, определены экспериментально.

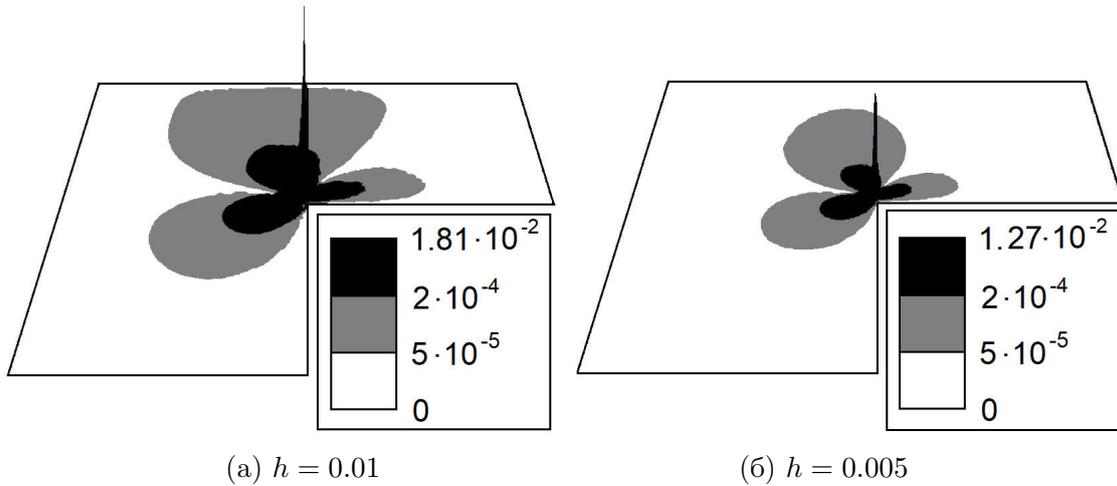


Рис. 3. Абсолютная погрешность первой компоненты поля скоростей приближенного обобщенного решения ($\nu = 0, \nu^* = \mu^* = 0, \delta = 1$), $\omega = \frac{3\pi}{2}$

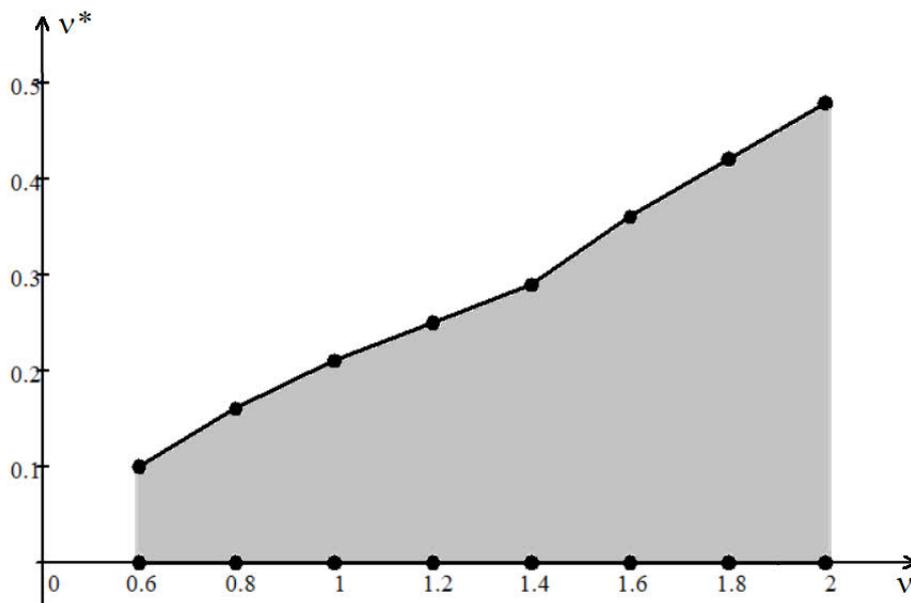


Рис. 4. Область подходящих свободных параметров метода для угла $\omega = \frac{3\pi}{2}$

Далее поставим себе задачу об определении области в переменных $(\nu, \nu^*)(\mu^* = \nu^*)$, для которой (в каждой ее точке) значение погрешности отличается от оптимальной не более чем на 6 процентов для всех величин $h = 0.01, 0.005, 0.0025$. В качестве радиуса δ -окрестности выбираем величину из диапазона $[0.01, 0.03]$. В экспериментах рассматриваем шаг изменения по переменной ν^* , равный 0.01 для всех положительных значений и шаг по переменной ν , равный 0.2 из диапазона $[0.6, 2.0]$ для Ω_1 и $[0.4, 2.0]$ для Ω_2 . На рис. 4 и 5 показаны экспериментально найденные области выбора свободных параметров предложенного подхода.

Таблица 3. Зависимость погрешностей приближенного R_ν -обобщенного решения, используя весовой МКЭ, от шага сетки h (по строкам) и степеней ν и ν^* ($\mu^* = \nu^*$) в норме $\|\mathbf{u}_\nu - \mathbf{u}_\nu^h\|_{\mathbf{W}_{2,\nu}^1(\Omega)}$, $\omega = \frac{5\pi}{4}$

(δ, ν)	(0.0125, 2.0)	(0.015, 1.8)
h	$\nu^* = 0.275$	$\nu^* = 0.25$
0.01	5.113e-6	1.756e-5
0.005	2.545e-6	8.731e-6
0.0025	1.277e-6	4.369e-6

Таблица 4. Зависимость погрешностей приближенного обобщенного решения, используя классический МКЭ, от шага сетки h (по столбцам) ($\nu = \mu^* = \nu^* = 0$, $\delta = 1$) в норме $\|\tilde{\mathbf{u}} - \tilde{\mathbf{u}}^h\|_{\mathbf{W}_{2,0}^1(\Omega)}$, $\omega = \frac{5\pi}{4}$

h	0.01	0.005	0.0025
$\ \tilde{\mathbf{u}} - \tilde{\mathbf{u}}^h\ _{\mathbf{W}_{2,0}^1(\Omega)}$	8.329e-2	5.202e-2	3.262e-2

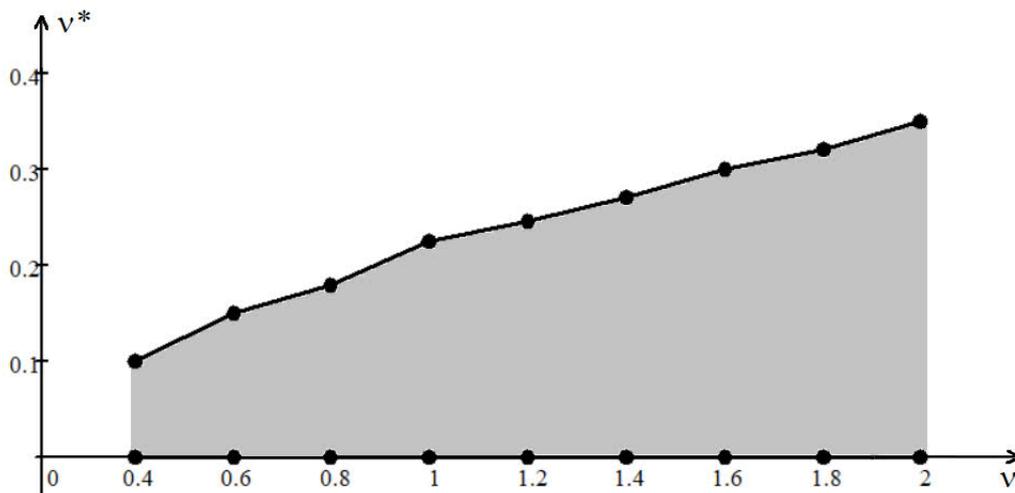


Рис. 5. Область подходящих свободных параметров метода для угла $\omega = \frac{5\pi}{4}$

Заключение

В статье введено понятие R_ν -обобщенное решение задачи Стокса с граничными условиями Неймана в несимметричной смешанной постановке задачи. Компоненты решения определяются в множествах весовых пространств С.Л. Соболева. Построен весовой метод конечных элементов, основанный на определении конечно-элементной пары Тейлора—Худа 2го порядка.

Результаты численных экспериментов задачи Стокса с граничными условиями Неймана в невыпуклой области приводят к следующим выводам.

- Приближенное обобщенное решение, полученное с помощью классического МКЭ, сходится к точному решению задачи со скоростью не выше $\mathcal{O}(h^{0.54})$ и $\mathcal{O}(h^{0.67})$ для Ω_1 и

Ω_2 соответственно в норме пространства $\mathbf{W}_{2,0}^1(\Omega)$ (табл. 2 и 4), в то время как приближенное R_ν -обобщенное решение, полученное благодаря весового МКЭ, сходится к точному решению со скоростью $\mathcal{O}(h)$ в норме пространства $\mathbf{W}_{2,\nu}^1(\Omega)$ для всех величин входящего угла (табл. 1 и 3). Результаты согласуются с априорными оценками скорости сходимости приближенного решения к точному решению задачи (см. 1 [10]).

- Результат достигается без сгущения сетки в окрестности точки сингулярности.
- Предложенный весовой МКЭ позволяет производить высокоточные вычисления в окрестности вершины входящего угла и не дает погрешности распространяться во внутреннюю часть расчетной области, где решение обладает достаточной гладкостью (см. рис. 2).
- Построена область подходящих свободных параметров подхода. Малому изменению параметров области соответствует малое изменение погрешности.

В дальнейших исследованиях, основываясь на определении R_ν -обобщенного решения и предложенного весового МКЭ, предполагаем получить численные результаты для различных форм нелинейных уравнений Навье—Стокса, таких как конвективная, кососимметричная, консервативная и других.

Кроме того, классический МКЭ теряет в два раза свой порядок точности для краевой задачи с граничными условиями Дирихле—Неймана на сторонах входящего угла по сравнению с МКЭ для краевой задачи с граничными условиями Дирихле—Дирихле и Неймана—Неймана (см. 1). Предложенный нами подход не теряет свой порядок точности — это было установлено для задачи теории упругости [19]. В одной из ближайших работ исследуем весовой МКЭ для рассматриваемой задачи в области с входящим углом на границе и граничными условиями Дирихле—Неймана на сторонах этого угла.

Работа выполнена в рамках государственного задания ИПМ ДВО РАН (№ 075-00459-24-00). Результаты получены с использованием оборудования Центра коллективного пользования «Дальневосточный вычислительный ресурс» ИАПУ ДВО РАН (<https://cc.dvo.ru>).

Литература

1. Ciarlet P. The Finite Element Method for Elliptic Problems. Amsterdam: North-Holland, 1978. 529 p.
2. Burda P., Novotny J., Sitek J. Precise FEM solution of a corner singularity using an adjusted mesh // International Journal for Numerical Methods in Fluids. 2005. Vol. 47. P. 1285–1292. DOI: [10.1002/flid.929](https://doi.org/10.1002/flid.929).
3. Choi H.J., Kweon J.R. A finite element method for singular solutions of the Navier–Stokes // Journal of Computational and Applied Mathematics. 2016. Vol. 292. P. 342–362. DOI: [10.1016/j.cam.2015.07.006](https://doi.org/10.1016/j.cam.2015.07.006).
4. Rukavishnikov V.A. Differential properties of an R_ν -generalized solution of the Dirichlet problem // Soviet Mathematics Doklady. 1990. Vol. 40. P. 653–655.
5. Rukavishnikov V.A. Methods of numerical analysis for boundary value problem with strong singularity // Russian Journal of Numerical Analysis and Mathematical Modelling. 2009. Vol. 24. P. 565–590. DOI: [10.1515/RJNAMM.2009.035](https://doi.org/10.1515/RJNAMM.2009.035).
6. Rukavishnikov V.A., Rukavishnikova E.I. Existence and uniqueness of an R_ν -generalized solution of the Dirichlet problem for the Lamé system with a corner singularity // Differential

- Equations. 2019. Vol. 55, no. 6. P. 832–840. DOI: [10.1134/S0012266119060107](https://doi.org/10.1134/S0012266119060107).
7. Rukavishnikov V.A., Rukavishnikova E.I. On the Dirichlet problem with corner singularity // Mathematics. 2020. Vol. 8, no. 11. P. 1870. DOI: [10.3390/math8111870](https://doi.org/10.3390/math8111870).
 8. Rukavishnikov V.A., Rukavishnikov A.V. On the existence and uniqueness of an R_ν -generalized solution to the Stokes problem with corner singularity // Mathematics. 2022. Vol. 10, no. 10. P. 1752. DOI: [10.3390/math10101752](https://doi.org/10.3390/math10101752).
 9. Rukavishnikov V.A., Rukavishnikov A.V. Theoretical analysis and construction of numerical method for solving the Navier-Stokes equations in rotation form with corner singularity // Journal of Computational and Applied Mathematics. 2023. Vol. 429. P. 115218. DOI: [10.1016/j.cam.2023.115218](https://doi.org/10.1016/j.cam.2023.115218).
 10. Rukavishnikov V.A., Rukavishnikova E.I. Weighted finite element method and body of optimal parameters for elasticity problem with singularity // Computers & Mathematics with Applications. 2023. Vol. 151. P. 408–417. DOI: [10.1016/j.camwa.2023.10.021](https://doi.org/10.1016/j.camwa.2023.10.021).
 11. Rukavishnikov A.V., Rukavishnikov V.A. New numerical approach for the steady-state Navier–Stokes equations with corner singularity // International Journal of Computational Methods. 2022. Vol. 19, no. 9. P. 2250012. DOI: [10.1142/S0219876222500128](https://doi.org/10.1142/S0219876222500128).
 12. Rukavishnikov V.A., Rukavishnikova E.I. Numerical method for Dirichlet problem with degeneration of the solution on the entire boundary // Symmetry. 2019. Vol. 11, no. 12. P. 1455. DOI: [10.3390/sym11121455](https://doi.org/10.3390/sym11121455).
 13. Mitrea M., Monniaux S., Wright M. The Stokes operator with Neumann boundary conditions in Lipschitz domains // Journal of Mathematical Sciences. 2011. Vol. 176, no. 3. P. 409–457. DOI: [10.1007/s10958-011-0400-0](https://doi.org/10.1007/s10958-011-0400-0).
 14. Monniaux S. Various boundary conditions for Navier-Stokes equations in bounded Lipschitz domains // Discrete and Continuous Dynamical Systems - S. 2013. Vol. 6, no. 5. P. 1355–1369. DOI: [10.3934/dcdss.2013.6.1355](https://doi.org/10.3934/dcdss.2013.6.1355).
 15. Shibata Y., Shimizu S. On the Stokes equation with Neumann boundary condition. // Banach Center Publications. 2005. Vol. 70. P. 239–250.
 16. Denis C., ter Elst A.F.M. The Stokes Dirichlet-to-Neumann operator // Journal of Evolution Equations. 2024. Vol. 24. P. 22. DOI: [10.1007/s00028-023-00930-x](https://doi.org/10.1007/s00028-023-00930-x).
 17. Rukavishnikov V.A., Rukavishnikov A.V. On the properties of operators of the Stokes problem with corner singularity in nonsymmetric variational formulation // Mathematics. 2022. Vol. 10, no. 6. P. 889. DOI: [10.3390/math10060889](https://doi.org/10.3390/math10060889).
 18. Boffi D., Brezzi F., Fortin M. Mixed finite element methods and applications. Berlin: Springer, 2013. 685 p. DOI: [10.1007/978-3-642-36519-5](https://doi.org/10.1007/978-3-642-36519-5).
 19. Rukavishnikov V.A., Rukavishnikova E.I. Weighted finite-element method for Elasticity problems with singularity. Finite element method. Simulations, numerical analysis and solution techniques. London: IntechOpen, 2018. P. 295–311. DOI: [10.5772/intechopen.72733](https://doi.org/10.5772/intechopen.72733).

Рукавишников Алексей Викторович, к.ф.-м.н., доцент, ведущий научный сотрудник, Хабаровского отделения Института прикладной математики ДВО РАН (Хабаровск, Российская Федерация)

ON NUMERICAL METHOD FOR THE STOKES PROBLEM WITH NEUMANN BOUNDARY CONDITIONS IN NON-CONVEX DOMAIN

© 2024 A.V. Rukavishnikov

Khabarovsk Division of Institute for Applied Mathematics, FEB RAS

(st. Seryshev 60, Khabarovsk, 680038 Russia)

E-mail: 78321a@mail.ru

Received: 15.11.2024

The Stokes problem with Neumann boundary conditions with an incoming angle at the boundary of the two-dimensional domain is considered. The concept of an R_ν -generalized solution in sets of weighted Sobolev spaces is introduced. A weighted finite element method on a uniform grid is constructed based on the second-order Taylor-Hood finite element pair and the introduction to the basis of a weight function in some powers ν^* and μ^* for the components of the velocity field and the scalar pressure function, respectively. The weight function in the domain coincides with the function of the distance from the point to the vertex of the incoming angle in some δ -neighborhood and the constant δ outside it. Numerical experiments in the non-convex domain is carried out. The convergence rate of the approximate solution to the exact one is obtained, which is independent of an incoming angle value and exceeds the convergence rate for the classical FEM. The convergence result is achieved without geometric refinement of the mesh in the vicinity of the singularity point. A series of numerical experiments for different values of an incoming angle was carried out and the domain of suitable free parameters of the proposed approach is found. For any point of the constructed domain, an optimal result, from the point of view of convergence, is achieved. The area of choice of suitable free parameters differs from the area for the considered problem with Dirichlet boundary conditions.

Keywords: angle singularity, Stokes problem with Neumann boundary conditions, R_ν -generalized solution, weighted FEM.

FOR CITATION

Rukavishnikov A.V. On Numerical Method for the Stokes Problem with Neumann Boundary Conditions in Non-convex Domain. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2024. Vol. 13, no. 4. P. 5–18. (in Russian) DOI: 10.14529/cmse240401.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Ciarlet P. The Finite Element Method for Elliptic Problems. Amsterdam: North-Holland, 1978. 529 p.
2. Burda P., Novotny J., Sitek J. Precise FEM solution of a corner singularity using an adjusted mesh. International Journal for Numerical Methods in Fluids. 2005. Vol. 47. P. 1285–1292. DOI: [10.1002/flid.929](https://doi.org/10.1002/flid.929).
3. Choi H.J., Kweon J.R. A finite element method for singular solutions of the Navier–Stokes. Journal of Computational and Applied Mathematics. 2016. Vol. 292. P. 342–362. DOI: [10.1016/j.cam.2015.07.006](https://doi.org/10.1016/j.cam.2015.07.006).
4. Rukavishnikov V.A. Differential properties of an R_ν -generalized solution of the Dirichlet

- problem. Soviet Mathematics Doklady. 1990. Vol. 40. P. 653–655.
5. Rukavishnikov V.A. Methods of numerical analysis for boundary value problem with strong singularity. Russian Journal of Numerical Analysis and Mathematical Modelling. 2009. Vol. 24. P. 565–590. DOI: [10.1515/RJNAMM.2009.035](https://doi.org/10.1515/RJNAMM.2009.035).
 6. Rukavishnikov V.A., Rukavishnikova E.I. Existence and uniqueness of an R_ν -generalized solution of the Dirichlet problem for the Lamé system with a corner singularity. Differential Equations. 2019. Vol. 55, no. 6. P. 832–840. DOI: [10.1134/S0012266119060107](https://doi.org/10.1134/S0012266119060107).
 7. Rukavishnikov V.A., Rukavishnikova E.I. On the Dirichlet problem with corner singularity. Mathematics. 2020. Vol. 8, no. 11. P. 1870. DOI: [10.3390/math8111870](https://doi.org/10.3390/math8111870).
 8. Rukavishnikov V.A., Rukavishnikov A.V. On the existence and uniqueness of an R_ν -generalized solution to the Stokes problem with corner singularity. Mathematics. 2022. Vol. 10, no. 10. P. 1752. DOI: [10.3390/math10101752](https://doi.org/10.3390/math10101752).
 9. Rukavishnikov V.A., Rukavishnikov A.V. Theoretical analysis and construction of numerical method for solving the Navier-Stokes equations in rotation form with corner singularity. Journal of Computational and Applied Mathematics. 2023. Vol. 429. P. 115218. DOI: [10.1016/j.cam.2023.115218](https://doi.org/10.1016/j.cam.2023.115218).
 10. Rukavishnikov V.A., Rukavishnikova E.I. Weighted finite element method and body of optimal parameters for elasticity problem with singularity. Computers & Mathematics with Applications. 2023. Vol. 151. P. 408–417. DOI: [10.1016/j.camwa.2023.10.021](https://doi.org/10.1016/j.camwa.2023.10.021).
 11. Rukavishnikov A.V., Rukavishnikov V.A. New numerical approach for the steady-state Navier–Stokes equations with corner singularity. International Journal of Computational Methods. 2022. Vol. 19, no. 9. P. 2250012. DOI: [10.1142/S0219876222500128](https://doi.org/10.1142/S0219876222500128).
 12. Rukavishnikov V.A., Rukavishnikova E.I. Numerical method for Dirichlet problem with degeneration of the solution on the entire boundary. Symmetry. 2019. Vol. 11, no. 12. P. 1455. DOI: [10.3390/sym11121455](https://doi.org/10.3390/sym11121455).
 13. Mitrea M., Monniaux S., Wright M. The Stokes operator with Neumann boundary conditions in Lipschitz domains. Journal of Mathematical Sciences. 2011. Vol. 176, no. 3. P. 409–457. DOI: [10.1007/s10958-011-0400-0](https://doi.org/10.1007/s10958-011-0400-0).
 14. Monniaux S. Various boundary conditions for Navier–Stokes equations in bounded Lipschitz domains. Discrete and Continuous Dynamical Systems - S. 2013. Vol. 6, no. 5. P. 1355–1369. DOI: [10.3934/dcdss.2013.6.1355](https://doi.org/10.3934/dcdss.2013.6.1355).
 15. Shibata Y., Shimizu S. On the Stokes equation with Neumann boundary condition. Banach Center Publications. 2005. Vol. 70. P. 239–250.
 16. Denis C., ter Elst A.F.M. The Stokes Dirichlet-to-Neumann operator. Journal of Evolution Equations. 2024. Vol. 24. P. 22. DOI: [10.1007/s00028-023-00930-x](https://doi.org/10.1007/s00028-023-00930-x).
 17. Rukavishnikov V.A., Rukavishnikov A.V. On the properties of operators of the Stokes problem with corner singularity in nonsymmetric variational formulation. Mathematics. 2022. Vol. 10, no. 6. P. 889. DOI: [10.3390/math10060889](https://doi.org/10.3390/math10060889).
 18. Boffi D., Brezzi F., Fortin M. Mixed finite element methods and applications. Berlin: Springer, 2013. 685 p. DOI: [10.1007/978-3-642-36519-5](https://doi.org/10.1007/978-3-642-36519-5).
 19. Rukavishnikov V.A., Rukavishnikova E.I. Weighted finite-element method for Elasticity problems with singularity. Finite element method. Simulations, numerical analysis and solution techniques. London: IntechOpen, 2018. P. 295–311. DOI: [10.5772/intechopen.72733](https://doi.org/10.5772/intechopen.72733).

PRECISE LOCALIZATION OF PDF417 CODE BASED ON FAST HOUGH TRANSFORM

© 2024 D.G. Mitrofanov^{1,2}, P.K. Zlobin¹, J.A. Shemiakina¹,
P.V. Bezmaternykh^{1,3}

¹Smart Engines Service LLC (pr. 60-letiya Oktyabrya 9, Moscow, 117312 Russia),

²Lomonosov Moscow State University (GSP-1, Leninskie Gory 1, Moscow, 119991 Russia),

³FRC “Computer Science and Control” RAS (st. Vavilova 44, bld. 2, Moscow, 119333 Russia)

E-mail: mitrofanov.dmitryi@gmail.com, zlobin.pavel.k@gmail.com,

jshemiakina@smartengines.com, bezmaternykh@isa.ru

Received: 30.09.2024

The PDF417 is a popular barcode symbology which is widely used in a huge variety of business processes. In this paper, we propose an original method for precise PDF417 code localization. It can successfully process projectively distorted images captured via the mobile device cameras. The core of this method is the analysis of the Fast Hough Transform image. This analysis is aimed to: (a) determine the line, corresponding to the vanish point of vertical symbol sides, using the RANSAC algorithm; (b) select the best pair of Hough-points corresponding to the horizontal symbol sides. We also propose the evaluation methodology for assessing the accuracy of precise PDF417 localization and a new dataset SE-PDF417-SYN-400, which consists of 400 synthesized PDF417 images and is publicly available. The accuracy of the proposed method on SE-PDF417-SYN-400 is equal to 0.948, and its error rate is about four times less than the one obtained by the popular ZXing detector. The average running times on iPhone 8 and iPhone 14 Pro Max mobile devices are equal to 77 and 34 ms per image correspondingly.

Keywords: barcode reading, PDF417, Fast Hough Transform, vanish point, RANSAC.

FOR CITATION

Mitrofanov D.G., Zlobin P.K., Shemiakina J.A., Bezmaternykh P.V. Precise Localization of PDF417 Code Based on Fast Hough Transform. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2024. Vol. 13, no. 4. P. 19–34. DOI: 10.14529/cmse240402.

Introduction

Fast and reliable data entry is essential for a huge variety of business processes including identity document (ID) verification and boarding passes validation. The manual approach is very tedious and error-prone, so this is precisely the task where machines could replace humans. To simplify this task, special machine-readable graphical symbols named barcodes were introduced in the middle of the previous century.

Generally, they can be divided into three groups: one-dimensional (1D), stacked, and two-dimensional (2D) codes. The 1D codes are represented as a set of parallel dark thick and thin bars on light background. Typical examples of these types are UPC/EAN, Code 128, Interleaved 2 of 5. Stacked codes may be considered as a set of 1D codes placed on top of each other with some extra meta information bars. The PDF417 symbology [\[1\]](#) is the most representative example of such codes and is the subject of this paper. It is widely used in ID documents issued by American Association of Motor Vehicle Administrators (AAMVA) member jurisdictions and in paper boarding passes supported by the International Air Transport Association (IATA). Samples of these documents are presented in Fig. [\[1\]](#) A general structure of PDF417 code is shown in Fig. [\[2\]](#)



a) The AAMVA sample



b) The IATA sample

Fig 1. The expected PDF417 localization results (in red) for samples taken from the open sources

Nowadays, these codes are commonly captured via mobile device cameras in uncontrolled conditions. In such circumstances, captured barcode images are regularly distorted by perspective transformation [\[2\]](#).

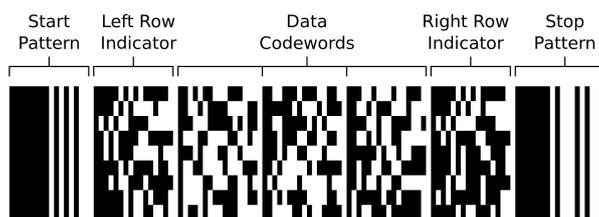


Fig 2. The PDF417 barcode structure. Image is taken from the open source

First goal in the barcode recognition field is to roughly determine the region of interest (ROI) containing the image of a symbol. There are plenty of such detectors, most of them nowadays are based on some supervised learning approaches [\[3\]](#). The next problem is the precise localization of the code symbol. That usually means finding a quadrangle of symbol corner points further denoted as $Q = \langle a, b, c, d \rangle$ (see red quadrangles in Fig. [\[1\]](#)). Let us denote this problem as \mathcal{P} . In this paper, a new method for such barcode corners detection problem is presented. By design, it may tackle with perspective distortion of symbols images and is suitable for the usage on mobile devices. Like some other methods, it is based on the usage of the discrete integral

image transformation. A typical choice in this case is the famous **Hough Transform** (HT). To overcome its computational “bloating”, we propose using its fast discrete approximation, known as **Fast Hough Transform** (FHT) [4][5]. The usage of such approximation is not trivial, so we do provide in-depth description of the resulting FHT image analysis specific for the problem under consideration. We also use RANSAC scheme to estimate bar lines directions that allows to deal with extra lines when the PDF417 symbol is placed on the complicated background.

The contributions of the paper can be formulated as follows. First, we propose the new method for precise PDF417 codes localization. It is based on FHT image analysis and RANSAC scheme. Second, we introduce a way to evaluate the accuracy of such precise localization. Finally, we present a new dataset SE-PDF417-SYN-400 suitable for this problem.

The rest of this paper is organized as follows. In Section [1] we review the related works. Then, in Section [2] we discuss the details of the proposed method. The experiment methodology and results are presented and analyzed in Section [3]. The **Conclusion** concludes this work.

1. Related Works

The barcode localization problem \mathcal{P} implies that we need to detect the position of symbol corners Q within an input image I . Currently, there is a wealth of approaches to deal with this problem. Some of these approaches are not barcode-type specific [6], and some of them hugely rely on the internal barcode structure and so-called “finder patterns”. At the same time, these approaches may impose different constraints on the input image. Some of them require the presence of only one target symbol per image, others request the estimation of symbol size in advance, and major part expect a binary image as an input instead of a grayscale one.

A common PDF417 barcode processing pipeline \mathbb{P} contains the following stages [7][8]: (a) ROI extraction; (b) image preprocessing; (c) symbol localization; (d) symbol segmentation; (e) message decoding. The subject of interest in this paper is the combination of stages (b) and (c).

One of the most popular steps in stage (b) is an application of some binarization procedure to the input image [9]. It simplifies the task of further image analysis and increases the overall runtime performance. By design, barcode symbol elements are black and white, and the usage of binarization step seems to be inherent. Nevertheless, in uncontrolled environment the complexity of barcode binarization task drastically increases due to uneven lighting conditions, presence of shadows or reflections. Moreover, binarization methods may be sensitive to noise and can severely damage some barcode patterns making their further decoding impossible [10]. Today, there is no generally accepted method for barcode binarization. Some special, “symbology oriented” binarization methods, including ones for PDF417, have already been published [11][12]. But their usage generally does not make much difference.

Another conventional image preprocessing technique for PDF417 detection is mathematical morphology (MM). The application of properly selected MM filters can enhance the barcode image and simplify the task of edge detection or the stage of symbol segmentation [8][13][14]. The outcome of MM filter depends on the shape and size of the structured element (SE) being used. If this size is too small, over-segmentation occurs, and vice versa. This SE size must be in accordance with the barcode module size, which can vary a lot in presence of perspective transformation. Therefore, the size of SE can not be set in advance in general case. In paper [15], authors propose to iterate through the set of various structural elements and return the best result. These iterative strategies suffer from significantly increased method running times, which makes them undesirable for the mobile devices.

There are localization methods which detect the “start” and the “stop” patterns of the symbol [7](#) (see Fig. [2](#)). The detection of such patterns commonly includes some template matching strategy. This step is especially dependent on binarization accuracy. The stripes in these patterns can be very thin and, thus, can be easily corrupted by binarization procedure. Having detected precise positions of these patterns we can estimate the Q of the whole barcode. Another approach is to extract the contours from the image and analyze them [7](#) [16](#). The contours of “start” and “stop” patterns can drastically simplify this process.

The Hough transform is a commonly used tool in the barcode localization problem. The example of HT application for 1D codes detection is presented in papers [8](#) [17](#). In the paper [18](#), a promising method for localizing general matrix codes was proposed. It is based on the special modification of HT and its analysis, but unlike our method, it requires high-resolution images. As for PDF417 codes, the paper [19](#) introduces the method of their localization and segmentation based on the analysis of HT image. Unfortunately, in that work the problem of extra elements placed near the PDF417 code symbol is not addressed.

The goal of analysis is to determine lines with strong edges corresponding to code’s columns and estimate their vanish point. These lines are expected to be corresponded to the maxima of HT image and the vanish point corresponds to a line on HT image passing through these maxima. The naive HT calculation is notoriously slow. In this paper, a new method of code localization based on FHT analysis instead of HT is presented. This method does not require preliminary binarization step.

Thus, in the literature we found three groups of approaches to deal with \mathcal{P} problem, which are based on: (a) preliminary image processing with mathematical morphology; (b) special patterns detection; (c) HT image analysis.

Unfortunately, at this point there some problems which preclude objective comparison of methods proposed in listed papers. Some of these works, like [19](#), does not provide any information about resulting performance or accuracy. Another do assess their methods over private datasets, which are commonly very small. Moreover, the evaluation methodology for the \mathcal{P} problem is not established. The commercial barcode readers that promote themselves as effective tools for PDF417 recognition focus only on the end-to-end metric, i.e. the number of correctly read codes. They do not provide any details about their internal subsystems and their performance. Therefore, these solutions can not be used to evaluate the \mathcal{P} problem. In order to overcome all these issues we propose: (a) the \mathcal{P} problem oriented dataset; (b) the evaluation methodology; (c) the baseline.

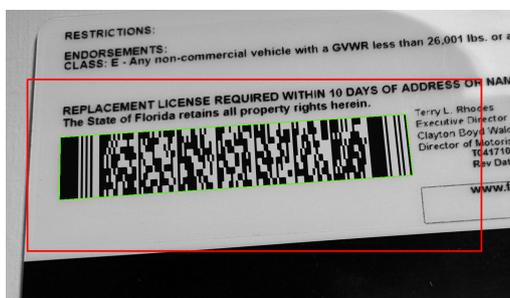


Fig 3. The image shows two examples: a red quadrangle indicating a ROI and a green quadrangle demonstrating the result of localization

2. Proposed Algorithm

Let $I_{W \times H}$ be the region of interest containing the whole PDF417 symbol $S_{R \times C}$, where R and C denotes the number of rows and columns respectively. It is assumed that $I_{W \times H}$ is a grayscale image with certain constraints: (a) there is a gap between the symbol and the input image bounds (at least 5 pixels); (b) the center of image $I_{W \times H}$ lies within the symbol S ; (c) the area of S occupies at least quarter of the image area. The location of the symbol is determined as an ordered quadrangle of code corners $Q = \langle a, b, c, d \rangle$. Thus, our goal is to construct an algorithm \mathcal{A}_{full} , which is able to detect such a quadrangle Q within the image $I_{W \times H}$ (see Fig. 1).

The target quadrangle Q may be defined as intersection of four lines which lay on symbol sides. At this point we assume that the code is oriented “mainly horizontally”: the total length of parallel projection of sides (a, b) and (c, d) on axis X is bigger than on axis Y (Fig. 1). Let us denote the pairs of horizontal and vertical sides as $S_h = \{(a, b), (c, d)\}$ and $S_v = \{(b, c), (a, d)\}$ respectively. In order to find these pairs, we propose to analyze the FHT image of $I_{W \times H}$. Let denote this image as FHT_{full} .

At first stage, the “strongest” points are detected in FHT_{full} . Having found these points, the corresponding sides are determined using two completely different approaches, namely \mathcal{A}_H and \mathcal{A}_V . Now, let us describe these algorithms in depth.

Stage 1: “strongest” points extraction. To reduce the running time and simplify the process of parameter fine-tuning, we scale input image $I_{W \times H}$ in J . Both its sides are scaled equally, and the scale factor f is calculated by Equation (1):

$$f = \frac{D}{\max(W, H)}. \quad (1)$$

In this work, the value of D is set to 800.

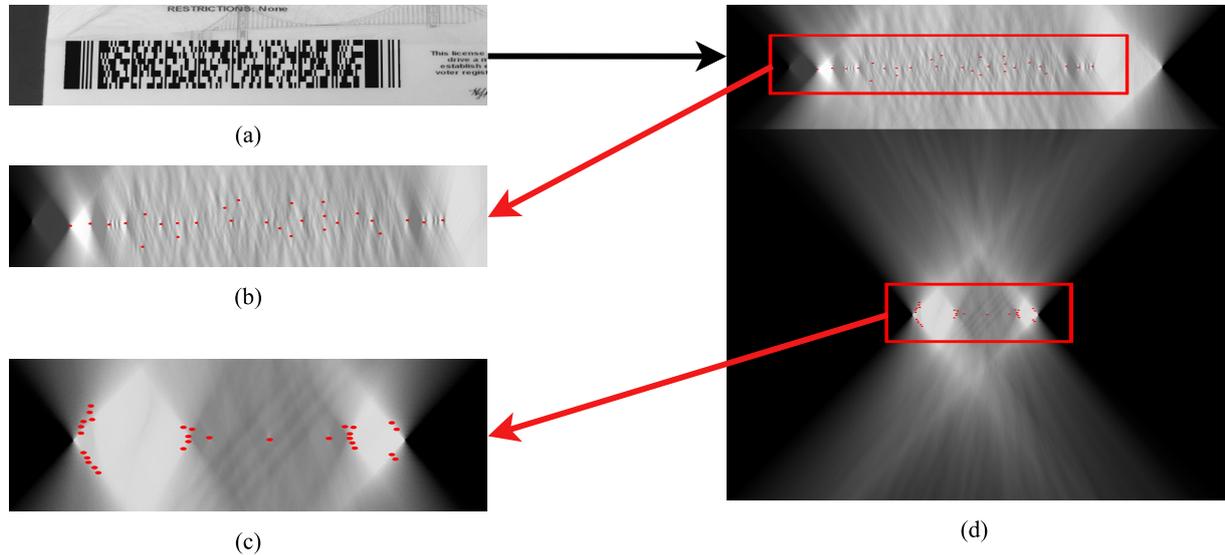


Fig 4. The “strongest” points $\langle P_h, P_v \rangle$ for the source image. (a) Source image I ; (b) detected P_h points; (c) detected P_v points; (d) Sets $\langle P_h, P_v \rangle$ on the FHT image

Then, the image $FHT_{full}(J) = FHT_h(J) \cup FHT_v(J)$ is calculated. Here, FHT_h and FHT_v parts correspond to horizontal and vertical lines, respectively (Fig. 4), and the sign ‘ \cup ’ denotes the procedure of their concatenation. This procedure is described in details in [20]. After that, the local maxima M of FHT_{full} image are calculated. Then, they are sorted by intensity and are

examined from the “highest” to the “lowest” ones. For every such element e , all other maxima in window of size W_{size} centered in e are suppressed, thus filtering out some maxima. The size W_{size} is set to 17 pixels in this work. After that, the sets of brightest points P_h and P_v from Hough spaces FHT_h and FHT_v respectively are selected (Fig. 4) and sorted by their intensity. Further, all but the first N_h is set to 50 and N_v is set to 30 points are removed from consideration and the “survived” ones form the target pair of sets $\langle P_h, P_v \rangle$. The whole procedure of the proposed method \mathcal{A}_{STP} of $\langle P_h, P_v \rangle$ extraction is summarized in Algorithm 1 and its application for a sample image is demonstrated in Fig. 4

Algorithm 1: “strongest” points $\langle P_h, P_v \rangle$ extraction

Data: grayscale image $I_{W \times H}, D > 0, W_{size} > 0, N_h > 0, N_v > 0$

Result: $\langle P_h, P_v \rangle$

- 1 $J \leftarrow ScaleImage(I_{W \times H}, D)$;
 - 2 $FHT_{full} \leftarrow FastHoughTransform(J)$;
 - 3 $M \leftarrow ExtractLocalMaxima(FHT_{full})$;
 - 4 $S \leftarrow NonMaximaSuppression(M, W_{size})$;
 - 5 $\langle P_h, P_v \rangle \leftarrow TakeBestMaxima(S, N_h, N_v)$;
 - 6 *return* $\langle P_h, P_v \rangle$;
-

Stage 2: horizontal lines detection. The goal of this algorithm \mathcal{A}_H is to choose the best pair of “strongest” points $\mathbf{R} = \langle R_l, R_u \rangle$ and, therefore, determine a pair of corresponding symbol code sides $\mathbf{S}_h = \langle H_l, H_u \rangle$.

Let us consider the surrounding area of the upper side H_u of the symbol on source image $I_{W \times H}$. There should be predominantly “white” pixels above it and predominantly “black” pixels below it. As for the lower side H_l , the intensity of pixels are arranged in the opposite way (see Fig. 2). Relying on this observation, we divide the points P_h (obtained on the previous stage) into three sets: (a) \mathcal{H}^u – candidate points for H_u ; (b) \mathcal{H}^l – candidate points for H_l ; (c) the other ones. The third set of points is discarded from further consideration. Let us generate the set of all possible pairs $C_h = \mathcal{H}^u \times \mathcal{H}^l$. On FHT_h image the area corresponding to the barcode symbol should not contain major intensity changes. So, the pairs of points forming a non-monotonous segment are filtered out. To do this, the third central statistical moment M_3 of the segment is calculated. Then, the value $|M_3|$ is compared with the threshold T_M . If it is greater, than the pair is filtered out.

Resulting set is denoted as C_f . Among its elements, the pair $\langle R_l, R_u \rangle = \mathop{\max}_{(i,j) \in C_f} (intensity(i) + intensity(j))$, $i \in \mathcal{H}^l, j \in \mathcal{H}^u$ is chosen as the best one. Thus, the pair $\mathbf{S}_h = \langle H_l, H_u \rangle$ can be restored. The algorithm of obtaining a straight line from a point on FHT image is shown in 20.

Algorithm 2: horizontal sides \mathbf{S}_h calculation

Data: grayscale image $FHT_h, P_h, T_M > 0$

Result: $\langle H_l, H_u \rangle$

- 1 $\mathcal{H}^l, \mathcal{H}^u \leftarrow SplitPoints(FHT_h, P_h)$;
 - 2 $C_h \leftarrow CombinePairs(\mathcal{H}^l, \mathcal{H}^u)$;
 - 3 $C_f \leftarrow FilterPairs(C_h)$;
 - 4 $\mathbf{R} \leftarrow SelectBestPair(C_f)$;
 - 5 $\langle H_l, H_u \rangle \leftarrow RestoreSides(\mathbf{R})$;
 - 6 *return* $\langle H_l, H_u \rangle$;
-

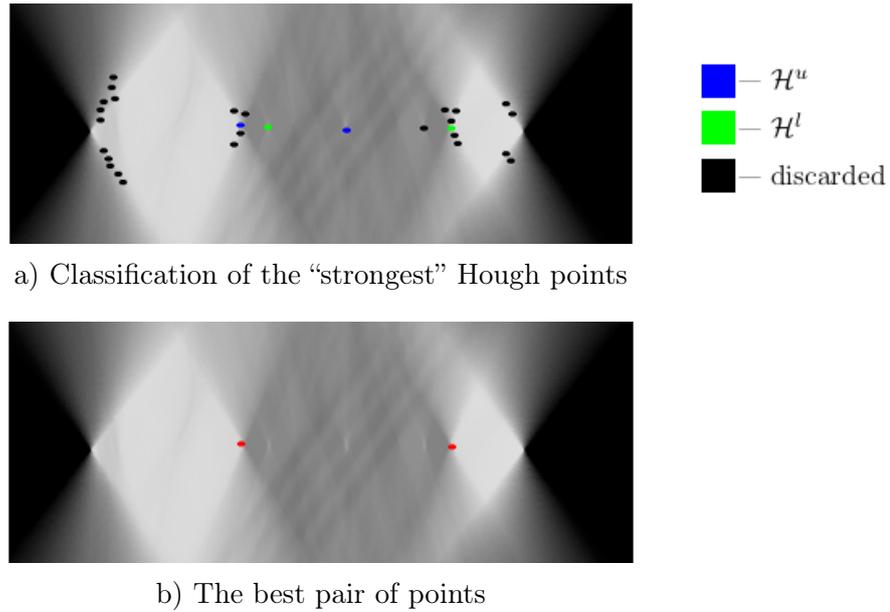


Fig 5. Example of horizontal sides selection

Stage 3: vertical lines detection. Now we need to find $S_v = \langle V_l, V_r \rangle$ — the external vertical sides of the symbol. But this time, we expect to have the similar elements inside the code which separate the columns of PDF417 symbol. Such elements can be observed in Fig. 2. Thus, we need to choose another approach for their detection. The lines corresponding to the edges of guard patterns and internal edges share the same vanish point vp . Let denote the image of vp in FHT_v space as l . This image l is represented as a straight line. We expect that the major part of points from P_v lies on this line. The other part of the points is considered as “outliers”. To robustly determine the parameters of line l , we use the well-known RANSAC method [21]. The intensity profile P_l corresponding to detected line l is extracted from image FHT_v . Such profile is presented in Fig. 6. The next step is to find the PDF417 guard patterns in this profile. This can be easily done by matching the known in advance pattern [1] with the given profile. After that, the external edges of such patterns are taken as the symbol sides S_v . The full algorithm \mathcal{A}_V is presented in Algorithm 3.

Algorithm 3: vertical sides L_v calculation

Data: grayscale image FHT_v, P_v

Result: $\langle V_l, V_r \rangle$

- 1 $l \leftarrow RANSAC(FHT_v, P_v)$;
 - 2 $P_l \leftarrow CalculateProfile(l)$;
 - 3 $\langle G_l, G_r \rangle \leftarrow FindGuardPatterns(P_l)$;
 - 4 $\langle V_l, V_r \rangle \leftarrow GetExternalEdges(G_l, G_r)$;
 - 5 *return* $\langle V_l, V_r \rangle$;
-

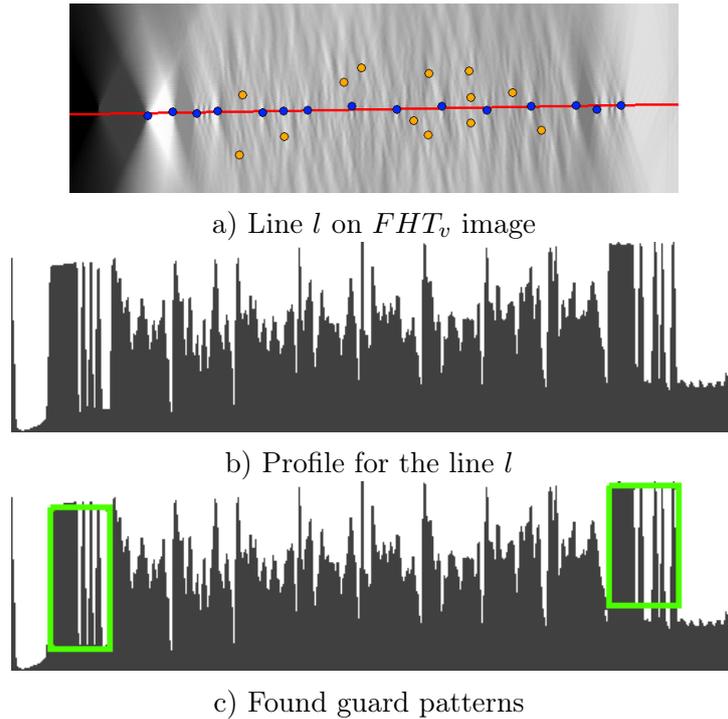


Fig 6. Example of vertical sides calculation

Stage 4: symbol quadrangle restoration. The final step is to restore the required quadrangle $Q = \langle a, b, c, d \rangle$. It can be calculated as cross product of detected sides: $a = V_l \times H_u$, $b = V_r \times H_u$, $c = V_r \times H_l$, $d = V_l \times H_l$, where the sign ‘ \times ’ denotes the intersection of lines.

The whole algorithm \mathcal{A}_{full} for Q detection is outlined in Algorithm 4

Algorithm 4: restoration of the symbol quadrangle Q

Data: grayscale image $I_{W \times H}$, $D > 0$, $W_{size} > 0$, $N_h > 0$, $N_v > 0$

Result: Q

- 1 $(P_v, P_h) \leftarrow \mathcal{A}_{STP}(I_{W \times H}, D, W_{size}, N_h, N_v)$;
 - 2 $S_h \leftarrow \mathcal{A}_H(FHT_h, P_h)$;
 - 3 $S_v \leftarrow \mathcal{A}_V(FHT_v, P_v)$;
 - 4 *return* $IntersectionOfLines(S_h, S_v)$;
-

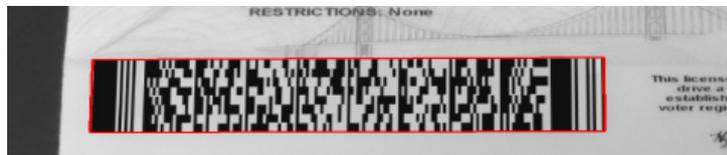


Fig 7. The localization result Q

3. Experiments

3.1. Evaluation Methodology

Let us consider a quadrangle Q found by algorithm \mathcal{A}_{full} . We need to define whether this quadrangle is correctly found or not. In order to do it, we have to know in advance the expected “ground truth” result $G_I = \langle Q_I, R_I, C_I \rangle$. Here, Q_I stands for a quadrangle of symbol location

within the image $I_{W \times H}$; R_I and C_I denote the expected numbers of symbol rows and columns respectively.

In this work, the evaluation is interpreted as a binary value $B(Q, G) \in \{0, 1\}$. Here, one stands for correct answers, zero — otherwise. We propose to calculate this binary value $B(Q, G)$ as comparison of some quadrangle score $E(Q, G) \in \mathbb{R}^+$ with the predefined threshold T : $B(Q, G) = [E(Q, G) < T]$. Here, $[\cdot]$ denotes Iverson notation; the value $T = 0.017$, it is chosen experimentally.

Now, let us define the way of $E(Q, G)$ calculation. At first, we need to introduce the rectified quadrangle TQ with sizes \mathcal{T}_W and \mathcal{T}_H . Then, $\mathcal{T}_W = 17 \cdot C_I$, and $\mathcal{T}_H = 3 \cdot R_I$. The left-top of TQ corner has $(0, 0)$ coordinates. Let the height of a module be $3px$, and the width of a module be $1px$. Now let calculate the parameters of projective transform H , which maps rectangle TQ into quadrangle Q : $RQ = H(Q)$. Let us calculate the maximum L_2 distance between corresponding corners of TQ and RQ . This value normalized by the perimeter of TQ is taken as evaluation value $E(Q)$. The whole algorithm for a single quadrangle is presented in Algorithm [5](#)

Algorithm 5: $E(Q_I, G_I)$ calculation

Data: $Q_I, G_I = \langle Q_I, R_I, C_I \rangle$
Result: $E(Q_I) \in \mathbb{R}^+$

- 1 $\mathcal{T}_W, \mathcal{T}_H = 17 \cdot C_I, 3 \cdot R_I;$
- 2 $TQ = Quad((0, 0), (\mathcal{T}_W, 0), (\mathcal{T}_W, \mathcal{T}_H), (0, \mathcal{T}_H));$
- 3 $H = ProjectiveTransform(TQ, FQ);$
- 4 $RQ = H(Q);$
- 5 $P = 2 \cdot (\mathcal{T}_W + \mathcal{T}_H);$
- 6 **for** i from 1 to 4 **do**
- 7 $d_i = \frac{\|TQ_i - RQ_i\|^2}{P};$
- 8 **end**
- 9 **return** $\max_{i \in [1..4]} (d_i);$

In fact, we have to evaluate this score not for a single image, but for the sequence of ones. The dataset of images with corresponding ground truth values is defined as follows: $\mathbb{D}_{syn} = \{(I_{W \times H}, G) \mid I_{W \times H} \in \mathbb{I}, G \in \mathbb{G}, |\mathbb{I}| = |\mathbb{G}|\}$. Here, \mathbb{I} represents the set of input images with PDF417 symbols, \mathbb{G}_I determines the ground truth for every image $I_{W \times H} \in \mathbb{I}$. N denotes the total number of items in the dataset \mathbb{D}_{syn} . Thus, the overall accuracy \mathcal{S} of the proposed method over the dataset \mathbb{D}_{syn} is evaluated as follows:

$$\mathcal{S}(\mathcal{A}_{full}, \mathbb{D}_{syn}) = \frac{1}{N} \cdot \sum_{I \in \mathbb{I}} B(\mathcal{A}_{full}(I), G_I). \quad (2)$$

The bigger the value of \mathcal{S} is, the better accuracy it shows.

3.2. Data Description

To evaluate the accuracy \mathcal{S} of proposed method, a dataset \mathbb{D}_{real} of real PDF417 codes was collected. For every image in it, the corresponding ground truth values were manually specified. In addition to $\langle Q, R, C \rangle$ they also include original textual message M . The real PDF417 codes often contain some private information, thus, it is not recommended to publish them into public domain due to legal issues. One way to overcome these issues is to synthesize the dataset \mathbb{D}_{syn} from \mathbb{D}_{real} in three steps.

At first, original messages M for the encoding are generated. Since a significant part of real images with PDF417 codes includes ID cards, it is desirable to simulate a similar message structure so as to approximate synthetic PDF417 codes to the real ones. The proposed message structure includes, partially or completely, the following details in random order: (a) first and second name; (b) geographical location; (c) some strings of letters and digits. For example, the message «HIJKLMNO XYZ1234 Lombardy Italy 9876543210 Laura Thomas PQRSTUUVW» is encoded in the barcode in Fig. 7

At the second stage, a rectangular code image S is generated from a given message M according to the PDF417 symbology specification [1]. The final step is to paste an image S into some background taken from \mathbb{D}_{real} knowing the original coordinates of symbols' boundaries Q . There is a number of constraints for such pasting. In order for to fit the requested Q , the image S must be properly scaled. It is necessary to impose restrictions on the min/max scaling of each side. To calculate the min/max valid side dimensions of Q , one has to calculate the min/max width and height of the quadrilateral. Thus, two ranges of lengths are obtained: one for width and one for height. In this case, there are two difficulties: (a) variability in the ratio of width and height; (b) variability in the dimensions of images, which leads to a large range of dimensions of S . In order to avoid excessive compression or stretching of S , adjusting it to the Q , it is necessary to generate a large number of PDF417 of all valid sizes and dimensions from the detected ranges and choose visually the best one from them.

Original colors of barcode modules for S are black and white. But naive pasting of such binary images produces unnatural results. So, the intensities of white and black colors on the real barcode are calculated and replaced with locally averaged ones in order to adopt to real image. The ground truth values for the dataset \mathbb{D}_{syn} are automatically populated from corresponding items in \mathbb{D}_{real} during the generation process. Finally, the \mathbb{D}_{syn} dataset contains 400 images. This dataset is freely available for download at <ftp://smartengines.com/se-pdf417-syn-400>.

3.3. Results and Discussion

First of all, we have to set the baseline for the \mathcal{P} problem. As was mentioned before, we can not use any commercial barcode reader, because they do not provide the required information. So, for these needs we use the well-known open source ZXing barcode reader (3.5.3 version). It can return required information of Q even in case of further code recognition failure. Let denote \mathcal{Z} the method of PDF417 corner points detection. It uses binarization and template matching strategy for this problem solving.

For the dataset \mathbb{D}_{syn} (see section 3.2) the value of $\mathcal{S}(\mathcal{Z}, \mathbb{D}_{syn})$ is equal to 0.803. Thus, the 321 quadrangles of 400 are correctly detected according to metric $B(Q, G) \in \{0, 1\}$ (see section 3.1). As for the proposed method, its accuracy value $\mathcal{S}(\mathcal{A}_{full}, \mathbb{D}_{syn})$ is equal to 0.948. Thus, the 379 quadrangles of 400 are correctly detected. It means that the error rate of the \mathcal{A}_{full} method is about four time less than for the \mathcal{Z} method. The l lines are detected correctly in 391 cases.

Let us examine the list of images where the proposed method does not produce the expected result. The 21 errors are distributed into three equal classes (i), (ii), and (iii). The class (i) contains errors associated with the incorrect choice of a pair of horizontal sides S_h . The class (ii) consists of cases where the algorithm correctly found the line l , but failed at guard pattern detection Fig. 8 The class (iii) includes the cases when the input image contains extra objects that create “strong” outliers on the FHT image Fig. 8

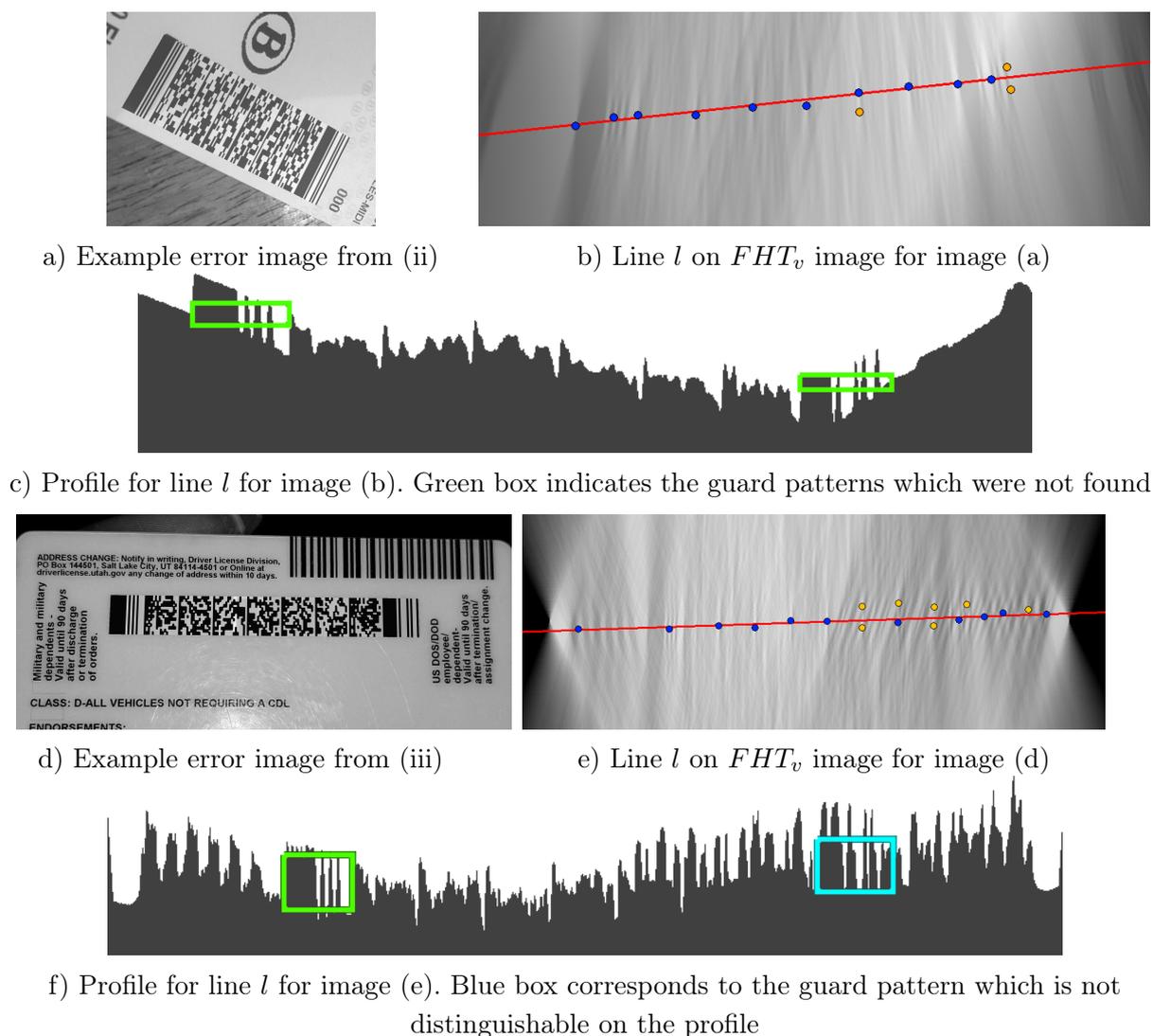


Fig 8. The proposed method error samples

The errors of the (i) and (ii) classes can be corrected by adding some steps into the proposed method. For the first class, the set of M first alternatives can be examined. The value of M can be estimated experimentally according to the limitations of the computational resources. For the second class, it is not obligatory to precisely determine the guard pattern positions. Instead of that, only the outer borders of the barcode itself can be detected. After that, the internal “barcodeness” level can be measured by some extra check. Another way to deal with this problem is to increase the initial resolution of the image being analyzed. In such case, the peaks and valleys in guard pattern structure are supposed to be better distinguishable. As for the last class, some errors from it can not be corrected within the proposed method due to its “integral nature”. Thus, the better initial localization of barcode region is, the better results it shows. Another approach is to inspect some regions on original image in advance and use the inspection results during the candidate selection process.



Fig 9. Examples when the quadrangle was found correctly on images containing extra objects

To estimate the running time of the proposed method on dataset \mathbb{D}_{syn} we used iPhone 8 and iPhone 14 Pro Max mobile devices. The running time of the proposed method implementation are equal to 29.2 and 12.9 seconds correspondingly, thus, the average running times are equal to 77 and 34 milliseconds per image.

Conclusion

In this paper, we propose the precise method of PDF417 code localization based on Fast Hough Transform usage and RANSAC scheme application. This method does not require commonly used preprocessing steps like binarization, edge detection or morphological filtering. It utilizes only preliminary image scaling which helps to normalize its running time and tune up its parameter values. The average running times measured on iPhone 8 and iPhone 14 Pro Max mobile devices are equal to 77 and 34 ms per image respectively. The computational experiments were conducted on an original dataset SE-PDF417-SYN-400, containing 400 synthesized PDF417 symbols. It is freely available for download at <ftp://smartengines.com/se-pdf417-syn-400>. It exhibits that the error rate of the proposed method is about four time less than for ZXing detector.

As for future research, the proposed method can be extended to other types of barcodes, such as linear or other stacked codes. In this case, the part related to the “strongest” points extraction from FHT image remains the same, but the part related to symbol edges selection has to be adopted to barcode type.

References

1. Information technology — Automatic identification and data capture techniques — PDF417 bar code symbology specification: Standard / International Organization for Standardization. 2015. DOI: [10.1007/978-981-19-0386-1_49](https://doi.org/10.1007/978-981-19-0386-1_49).
2. Chen C., Kot A.C., Yang H. A two-stage quality measure for mobile phone captured 2D barcode images. Pattern Recognition. 2013. Vol. 46, no. 9. P. 2588–2598. DOI: [10.1016/j.patcog.2013.01.031](https://doi.org/10.1016/j.patcog.2013.01.031).
3. Wudhikarn R., Charoenkwan P., Malang K. Deep Learning in Barcode Recognition: A Systematic Literature Review. IEEE Access. 2022. Vol. 10. P. 8049–8072. DOI: [10.1109/access.2022.3143033](https://doi.org/10.1109/access.2022.3143033).
4. Ginkel M. van, Hendriks C.L., Vliet L.J. van A short introduction to the Radon and Hough transforms and how they relate to each other. Quantitative Imaging Group, Imaging Science & Technology Department, TU Delft. 2004. P. 1–9.
5. Aliev M., Ershov E., Nikolaev D. On the use of FHT, its modification for practical applications and the structure of Hough image. The Proceedings SPIE. The 11th International

- Conference on Machine Vision (ICMV), Munich, Germany, November 1–3, 2018. Vol. 11041. 2019. DOI: [10.1117/12.2522803](https://doi.org/10.1117/12.2522803).
6. Gaur P., Tiwari S. Recognition of 2D Barcode Images Using Edge Detection and Morphological Operation. *International Journal of Computer Science and Mobile Computing*. 2014. Vol. 3, no. 4. P. 1277–1282.
 7. Liu F., Yin J., Li K., Liu Q. An improved recognition method of PDF417 barcode. *Chinese Conference on Pattern Recognition (CCPR)*, Chongqing, China, October 21–23, 2010. IEEE. 2010. P. 1–5. DOI: [10.1109/CCPR.2010.5659332](https://doi.org/10.1109/CCPR.2010.5659332).
 8. Kim Y.J., Lee J.Y. Algorithm of a Perspective Transform-Based PDF417 Barcode Recognition. *Wireless Personal Communications*. 2016. Vol. 89, no. 3. P. 893–911. DOI: [10.1007/s11277-016-3171-6](https://doi.org/10.1007/s11277-016-3171-6).
 9. Xiao Y., Yi J., Qiao G. Automatic Localization of Multi-type Barcodes in High-Resolution Images. *International Conference in Communications, Signal Processing, and Systems (CSPS)*, Changbaishan, China, July 23–24, 2021. Springer. 2021. P. 392–399. DOI: [10.1007/978-981-19-0390-8_103](https://doi.org/10.1007/978-981-19-0390-8_103).
 10. Bezmaternykh P.V., Vylegzhanin D.V., Gladilin S.A., Nikolaev D.P. 2D barcodes generative recognition. *Artificial Intelligence and Decision Making*. 2010. No. 4. P. 63–69.
 11. Xiao Q., Liu M., Liu Y. A new binarization method for PDF417 bar code by camera phone. *IEEE International Conference on Automation and Logistics (ICAL)*, Qingdao, China, September 1–3, 2008. IEEE. 2008. P. 1904–1908. DOI: [10.1109/ICAL.2008.4636470](https://doi.org/10.1109/ICAL.2008.4636470).
 12. Chen R., Yu Y., Xu X., *et al.* Adaptive binarization of QR code images for fast automatic sorting in warehouse systems. *Sensors*. 2019. Vol. 19, no. 24. P. 5466. DOI: [10.3390/s19245466](https://doi.org/10.3390/s19245466).
 13. Yang X., Gao X., Jia S.Q., Lu Q.Y. A Method for Extracting QR Code from Complex Background Based on Morphology. *Applied Mechanics and Materials*. 2013. Vol. 239. P. 1466–1471. DOI: [10.4028/www.scientific.net/AMM.239-240.1466](https://doi.org/10.4028/www.scientific.net/AMM.239-240.1466).
 14. Li D., Zhang L., Jin X. An improvement for PDF417 code authentication on mobile phone terminals based on code feature analysis and watermarking. *Multimedia Systems*. 2022. Vol. 28, no. 5. P. 1585–1596. DOI: [10.1109/ICIEA.2011.5975630](https://doi.org/10.1109/ICIEA.2011.5975630).
 15. Li J.-H., Wang W.-H., Rao T.-T., *et al.* Morphological segmentation of 2-D barcode gray scale image. *International Conference on Information System and Artificial Intelligence (ISAI)*, Hong Kong, China, June 24–26 2016. IEEE. 2016. P. 62–68. DOI: [10.1109/ISAI.2016.0022](https://doi.org/10.1109/ISAI.2016.0022).
 16. Trummer M., Denzler J. Reading out 2D Barcode PDF417. 2007. DOI: [10.1007/978-1-84628-945-3_19](https://doi.org/10.1007/978-1-84628-945-3_19).
 17. Zamberletti A., Gallo I., Albertini S., Noce L. Neural 1D barcode detection using the Hough transform. *IPSJ Transactions on Computer Vision and Applications*. 2015. Vol. 7. P. 1–9. DOI: [10.2197/ipsjtcva.7.1](https://doi.org/10.2197/ipsjtcva.7.1).
 18. Szentandrás I., Herout A., Dubská M. Fast detection and recognition of QR codes in high-resolution images. *Proceedings of the 28th spring conference on computer graphics*, Budmerice, Slovakia, May 2–4, 2012. 2012. P. 129–136. DOI: [10.1145/2448531.2448548](https://doi.org/10.1145/2448531.2448548).

19. Hu D., Chen X., Yu D., Li D. Algorithm for detecting the rows boundary of the PDF417 barcode. / ed. by L. Zhang, J. Zhang, M. Liao. Oct. 2005. 60431Z. DOI: [10.1117/12.654952](https://doi.org/10.1117/12.654952).
20. Aliev M., Kunina I., Nikolaev D., Polevoy D. On the practical aspects of computing the Hough image by the Brady-Yong algorithm. *Informatsionnye protsessy*. 2023. Vol. 23, no. 2. P. 250–273. DOI: [10.53921/18195822_2023_23_2_250](https://doi.org/10.53921/18195822_2023_23_2_250).
21. Fischler M.A., Bolles R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*. 1981. June. Vol. 24, no. 6. P. 381–395. DOI: [10.1145/358669.358692](https://doi.org/10.1145/358669.358692).

УДК 519.6

DOI: 10.14529/cmse240402

ТОЧНАЯ ЛОКАЛИЗАЦИЯ КОДА PDF417 НА ОСНОВЕ БЫСТРОГО ПРЕОБРАЗОВАНИЯ ХАФА

© 2024 Д.Г. Митрофанов^{1,2}, П.К. Злобин¹, Ю.А. Шемякина¹,
П.В. Безматерных^{1,3}

¹ООО «Смарт Энджинс Сервис» (117312 Москва, пр. 60-летия Октября, д. 9),

²Московский государственный университет имени М.В. Ломоносова
(119991 Москва, ул. Ленинские горы, д. 1),

³ФИЦ «Информатика и управление» РАН (119333 Москва, ул. Вавилова, д. 44, кор. 2)

E-mail: mitrofanov.dmitryi@gmail.com, zlobin.pavel.k@gmail.com,
jshemiakina@smartengines.com, bezmaternyh@isa.ru

Поступила в редакцию: 30.09.2024

PDF417 — это популярная символика штрихкода, которая широко используется в удостоверениях личности и транспортных системах. В этой статье мы предлагаем оригинальный метод локализации символа PDF417. Он может успешно обрабатывать проективно искаженные изображения, снятые с помощью камер мобильных устройств. Основой этого метода является анализ изображения быстрого преобразования Хафа. Целью этого анализа является: (а) определение линии, соответствующей точке схода вертикальных сторон символа, с использованием алгоритма RANSAC; (б) выбор лучшей пары точек Хафа, соответствующих горизонтальным сторонам символа. Мы также предлагаем методологию оценки для оценки точности локализации PDF417 и новый набор данных SE-PDF417-SYN-400, который состоит из 400 синтезированных изображений PDF417 и находится в открытом доступе. Точность предлагаемого метода на SE-PDF417-SYN-400 составляет 0.948, частота ошибок предлагаемого метода примерно в четыре раза меньше, чем у детектора ZXing. Среднее время работы на мобильных устройствах iPhone 8 и iPhone 14 Pro Max равно 77 и 34 мс на изображение.

Ключевые слова: распознавание штрихкодов, PDF417, быстрое преобразование Хафа, точка схода, RANSAC.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Mitrofanov D.G., Zlobin P.K., Shemiakina J.A., Bezmaternykh P.V. Precise Localization of PDF417 Code Based on Fast Hough Transform // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 4. С. 19–34. DOI: 10.14529/cmse240402.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

Литература

1. Information technology — Automatic identification and data capture techniques — PDF417 bar code symbology specification: Standard / International Organization for Standardization. 2015. DOI: [10.1007/978-981-19-0386-1_49](https://doi.org/10.1007/978-981-19-0386-1_49).
2. Chen C., Kot A.C., Yang H. A two-stage quality measure for mobile phone captured 2D barcode images // Pattern Recognition. 2013. Vol. 46, no. 9. P. 2588–2598. DOI: [10.1016/j.patcog.2013.01.031](https://doi.org/10.1016/j.patcog.2013.01.031).
3. Wudhikarn R., Charoenkwan P., Malang K. Deep Learning in Barcode Recognition: A Systematic Literature Review // IEEE Access. 2022. Vol. 10. P. 8049–8072. DOI: [10.1109/access.2022.3143033](https://doi.org/10.1109/access.2022.3143033).
4. Ginkel M. van, Hendriks C.L., Vliet L.J. van A short introduction to the Radon and Hough transforms and how they relate to each other // Quantitative Imaging Group, Imaging Science & Technology Department, TU Delft. 2004. P. 1–9.
5. Aliev M., Ershov E., Nikolaev D. On the use of FHT, its modification for practical applications and the structure of Hough image // The Proceedings SPIE. The 11th International Conference on Machine Vision, Munich, Germany, November 1–3, 2018. Vol. 11041. 2019. DOI: [10.1117/12.2522803](https://doi.org/10.1117/12.2522803).
6. Gaur P., Tiwari S. Recognition of 2D Barcode Images Using Edge Detection and Morphological Operation // International Journal of Computer Science and Mobile Computing. 2014. Vol. 3, no. 4. P. 1277–1282.
7. Liu F., Yin J., Li K., Liu Q. An improved recognition method of PDF417 barcode // Chinese Conference on Pattern Recognition (CCPR), Chongqing, China, October 21–23, 2010. IEEE. 2010. P. 1–5. DOI: [10.1109/CCPR.2010.5659332](https://doi.org/10.1109/CCPR.2010.5659332).
8. Kim Y.J., Lee J.Y. Algorithm of a Perspective Transform-Based PDF417 Barcode Recognition // Wireless Personal Communications. 2016. Vol. 89, no. 3. P. 893–911. DOI: [10.1007/s11277-016-3171-6](https://doi.org/10.1007/s11277-016-3171-6).
9. Xiao Y., Yi J., Qiao G. Automatic Localization of Multi-type Barcodes in High-Resolution Images // International Conference in Communications, Signal Processing, and Systems (CSPS), Changbaishan, China, July 23–24, 2021. Springer. 2021. P. 392–399. DOI: [10.1007/978-981-19-0390-8_103](https://doi.org/10.1007/978-981-19-0390-8_103).
10. Безматерных П.В., Вылегжанин Д.В., Гладилин С.А., Николаев Д.П. Генеративное распознавание двумерных штрихкодов // Искусственный интеллект и принятие решений. 2010. No. 4. P. 63–69.
11. Xiao Q., Liu M., Liu Y. A new binarization method for PDF417 bar code by camera phone // IEEE International Conference on Automation and Logistics (ICAL), Qingdao, China, September 1–3, 2008. IEEE. 2008. P. 1904–1908. DOI: [10.1109/ICAL.2008.4636470](https://doi.org/10.1109/ICAL.2008.4636470).
12. Chen R., Yu Y., Xu X., *et al.* Adaptive binarization of QR code images for fast automatic sorting in warehouse systems // Sensors. 2019. Vol. 19, no. 24. P. 5466. DOI: [10.3390/s19245466](https://doi.org/10.3390/s19245466).

13. Yang X., Gao X., Jia S.Q., Lu Q.Y. A Method for Extracting QR Code from Complex Background Based on Morphology // Applied Mechanics and Materials. 2013. Vol. 239. P. 1466–1471. DOI: [10.4028/www.scientific.net/AMM.239-240.1466](https://doi.org/10.4028/www.scientific.net/AMM.239-240.1466).
14. Li D., Zhang L., Jin X. An improvement for PDF417 code authentication on mobile phone terminals based on code feature analysis and watermarking // Multimedia Systems. 2022. Vol. 28, no. 5. P. 1585–1596. DOI: [10.1109/ICTEA.2011.5975630](https://doi.org/10.1109/ICTEA.2011.5975630).
15. Li J.-H., Wang W.-H., Rao T.-T., *et al.* Morphological segmentation of 2-D barcode gray scale image // International Conference on Information System and Artificial Intelligence (ISAI), Hong Kong, China, June 24–26 2016. IEEE. 2016. P. 62–68. DOI: [10.1109/ISAI.2016.0022](https://doi.org/10.1109/ISAI.2016.0022).
16. Trummer M., Denzler J. Reading out 2D Barcode PDF417. 2007. DOI: [10.1007/978-1-84628-945-3_19](https://doi.org/10.1007/978-1-84628-945-3_19).
17. Zamberletti A., Gallo I., Albertini S., Noce L. Neural 1D barcode detection using the Hough transform // IPSJ Transactions on Computer Vision and Applications. 2015. Vol. 7. P. 1–9. DOI: [10.2197/ipsjtcva.7.1](https://doi.org/10.2197/ipsjtcva.7.1).
18. Szentandrás I., Herout A., Dubská M. Fast detection and recognition of QR codes in high-resolution images // Proceedings of the 28th spring conference on computer graphics. 2012. P. 129–136. DOI: [10.1145/2448531.2448548](https://doi.org/10.1145/2448531.2448548).
19. Hu D., Chen X., Yu D., Li D. Algorithm for detecting the rows boundary of the PDF417 barcode // / ed. by L. Zhang, J. Zhang, M. Liao. Oct. 2005. 60431Z. DOI: [10.1117/12.654952](https://doi.org/10.1117/12.654952).
20. Aliev M., Kunina I., Nikolaev D., Polevoy D. On the practical aspects of computing the Hough image by the Brady-Yong algorithm // Informatsionnye protsessy. 2023. Vol. 23, no. 2. P. 250–273. DOI: [10.53921/18195822_2023_23_2_250](https://doi.org/10.53921/18195822_2023_23_2_250).
21. Fischler M.A., Bolles R.C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography // Communications of the ACM. 1981. June. Vol. 24, no. 6. P. 381–395. DOI: [10.1145/358669.358692](https://doi.org/10.1145/358669.358692).

Митрофанов Дмитрий Георгиевич, студент специалитета механико-математического факультета, Московский государственный университет имени М.В. Ломоносова (Москва, Российская Федерация), лаборант-программист, ООО «Смарт Энджинс Сервис» (Москва, Российская Федерация)

Злобин Павел Константинович, лаборант-программист, ООО «Смарт Энджинс Сервис» (Москва, Российская Федерация)

Шемякина Юлия Александровна, научный сотрудник-программист, ООО «Смарт Энджинс Сервис» (Москва, Российская Федерация)

Безматерных Павел Владимирович, программист первой категории, Федеральный исследовательский центр «Информатика и управление» Российской академии наук (Москва, Российская Федерация), научный сотрудник-программист, ООО «Смарт Энджинс Сервис» (Москва, Российская Федерация)

НЕЙРОСЕТЕВОЙ МЕТОД ОБНАРУЖЕНИЯ АНОМАЛИЙ В МНОГОМЕРНЫХ ПОТОКОВЫХ ВРЕМЕННЫХ РЯДАХ

© 2024 Я.А. Краева

Южно-Уральский государственный университет

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

E-mail: kraevaya@susu.ru

Поступила в редакцию: 20.10.2024

В статье рассмотрена задача детекции аномальных подпоследовательностей многомерного потокового временного ряда, элементы которого поступают в режиме реального времени, возникающая в настоящее время в широком спектре предметных областей: промышленный Интернет вещей, персональное здравоохранение и др. Предложен новый метод решения указанной задачи, получивший название mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). Предложенный метод использует концепцию диссонанса временного ряда (подпоследовательность, имеющая наиболее непохожего на нее ближайшего соседа), обобщенную на многомерный случай. Под многомерным диссонансом понимается N -мерная подпоследовательность d -мерного временного ряда (где $1 \leq N \leq d$), которая наиболее непохожа на все остальные подпоследовательности N -мерных временных рядов, полученных путем составления всевозможных сочетаний из d рядов по N . Детекция аномалий реализуется с помощью нейросетевой модели на основе сиамских нейросетей. Вычислительные эксперименты на реальных временных рядах из различных предметных областей показали, что метод mDiSSiD в среднем опережает по точности обнаружения аномалий передовые аналоги, использующие иные нейросетевые подходы (сверточные и рекуррентные нейронные сети, автоэнкодеры, генеративно-состязательные сети).

Ключевые слова: многомерный временной ряд, поиск аномалий, диссонанс, сноплет, сиамская нейронная сеть.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Краева Я.А. Нейросетевой метод обнаружения аномалий в многомерных потоковых временных рядах // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 4. С. 35–52. DOI: 10.14529/cmse240403.

Введение

В настоящее время в широком спектре приложений востребованы эффективные модели, методы и алгоритмы для детекции аномальных подпоследовательностей временных рядов, элементы которых поступают для обработки в режиме реального времени [1]: Интернет вещей [2], умное управление зданиями [3] [4], персональная медицина [5] и др. На текущий момент научным сообществом разработано большое количество методов и подходов к решению указанной задачи (см., например, обзорные статьи [6] [7], в которых суммарно рассматривается около ста различных методов). Детекция аномалий с частичным привлечением учителя (semi-supervised anomaly detection) является одним из наиболее часто используемых и эффективных подходов [6] [8]. Указанный подход предполагает этап обучения, в рамках которого нейросетевая модель «понимает», каким является нормальное поведение временного ряда, фигурирующего в качестве обучающей выборки. Далее модель применяется к тестовому временному ряду и помечает его подпоследовательности как аномальные, если они не соответствуют обученному нормальному поведению. К основным представителям подхода с частичным привлечением учителя относят методы LSTM-AD [9], DeepAnT [10] и др.

Ранее автором данной статьи был предложен метод DiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of anomalies) [11] для детекции аномальных подпоследовательностей потокового временного ряда с частичным привлечением учителя. Метод включает в себя нейросетевую модель, которая выдает оценку аномальности входной подпоследовательности, и алгоритм автоматизированного построения обучающей выборки для этой модели. Нейросетевая модель основана на архитектуре сиамской нейронной сети (Siamese Neural Network) [12], в которой в качестве подсети фигурирует модификация нейросетевой модели ResNet [13]. Алгоритм построения выборки отбрасывает из обучающей выборки подпоследовательности, соответствующие аномальным и нетипичным активностям субъекта, которые извлекаются на основе концепций диссонанса [14] и снippetsа [15] соответственно. Однако метод DiSSiD решает задачу детекции аномалий только для одномерного потокового временного ряда. В настоящей статье предлагается новый нейросетевой метод mDiSSiD (multivariate DiSSiD), который представляет собой обобщение вышеупомянутой ранней разработки на случай многомерного ряда. Метод mDiSSiD использует понятие многомерного диссонанса [16], которое представляет собой обобщение концепции диссонанса на случай многомерного временного ряда.

Статья организована следующим образом. В разделе [1] приводятся формальные определения базовых понятий. В разделе [2] представлено описание метода mDiSSiD. Раздел [3] содержит результаты вычислительных экспериментов по исследованию эффективности предложенного метода. **Заключение** подводит итоги исследования.

1. Теоретический базис

1.1. Временной ряд и подпоследовательность

Временной ряд T представляет собой последовательность вещественных значений, взятых в хронологическом порядке:

$$T = \{t_i\}_{i=1}^n, \quad t_i \in \mathbb{R}. \quad (1)$$

Число n называется длиной ряда и обозначается $|T|$.

Подпоследовательность $T_{i,m}$ временного ряда T представляет собой непрерывный промежуток из m элементов ряда, начиная с позиции i :

$$T_{i,m} = \{t_k\}_{k=i}^{i+m-1}, \quad 3 \leq m \ll n, \quad 1 \leq i \leq n - m + 1. \quad (2)$$

Множество всех подпоследовательностей ряда T , имеющих длину m , обозначим как S_T^m .

Многомерный временной ряд представляет собой занумерованную последовательность нескольких семантически связанных и синхронизированных по времени одномерных временных рядов равной длины. Обозначим размерность многомерного ряда целым положительным числом d ($d > 1$). Подобно одномерному случаю, введем обозначения \mathbf{T} , $\mathbf{T}_{i,m}$ и \mathbf{t}_i для многомерных временного ряда, его подпоследовательности и точки соответственно, и определим их следующим образом:

$$\mathbf{T} = [\{T^{(k)}\}_{k=1}^d]^\top, \quad (3)$$

$$\mathbf{T}_{i,m} = [\{T_{i,m}^{(k)}\}_{k=1}^d]^\top, \quad (4)$$

$$\mathbf{t}_i = [\{t_i^{(k)}\}_{k=1}^d]^\top. \quad (5)$$

1.2. Сниметы

Сниметы [15] (поведенческие шаблоны) временного ряда представляют собой множество подпоследовательностей, выражающих типичные активности субъекта, деятельность которого описывает данный ряд. Для формального определения сниметов зафиксируем временной ряд T , длину подпоследовательности m ($m \ll n$) и количество активностей субъекта K ($1 \leq K \leq n/m$), интересующее исследователя. Множество сниметов обозначается как C_T^m и представляет собой непустое подмножество сегментов ряда, имеющих длину m ¹

$$C_T^m = \{C_i\}_{i=1}^K, \quad C_i \in Seg_T^m, \quad Seg_T^m = \{T_{m \cdot (i-1)+1, m}\}_{i=1}^{n/m}. \quad (6)$$

Со сниметом $C_i \in C_T^m$ ассоциированы следующие атрибуты: индекс, ближайшие соседи и значимость данного снимета, которые обозначаются соответственно $C_i.index$, $C_i.NN$ и $C_i.frac$, и имеют следующие определения.

Индекс снимета представляет собой номер j сегмента Seg_j , которому соответствует подпоследовательность ряда $T_{m \cdot (j-1)+1, m}$.

Множество ближайших соседей содержит подпоследовательности, которые среди всех сегментов наиболее похожи на данный снимет в смысле расстояния MPdist:

$$C_i.NN = \{T_{j, m} \mid Seg_{C_i.index} = \arg \min_{1 \leq q \leq n/m} MPdist(T_{j, m}, Seg_q), 1 \leq j \leq n - m + 1\}. \quad (7)$$

Значимость снимета C_i вычисляется как отношение мощностей множеств $C_i.NN$ и S_T^m , при этом сниметы упорядочиваются по убыванию их значимости:

$$C_i.frac = \frac{|C_i.NN|}{n - m + 1}, \quad \forall C_i, C_j \in C_T^m : i < j \iff C_i.frac \geq C_j.frac. \quad (8)$$

Для формального определения расстояния $MPdist(A, B)$, $|A| = |B| = m$, фиксируем параметр ℓ ($1 < \ell < m$), который отражает длину семантически значимого непрерывного промежутка точек подпоследовательности. Вычисление $MPdist(A, B)$ предполагает следующие шаги [17]: 1) вычисление матричных профилей A и B , взятых в указанном и обратном порядке; 2) конкатенация профилей; 3) сортировка полученного ряда по возрастанию; 4) выдача k -го элемента результирующего ряда. То же в формальной записи:

$$MPdist_\ell(A, B) = AscSort(P_{ABBA})(k), \quad P_{ABBA} = P_{AB} \bullet P_{BA}, \quad (9)$$

где $AscSort(\cdot)$ — сортировка по возрастанию, символ \bullet обозначает конкатенацию, k ($0 < k < m$) — параметр, имеющий типичное значение $k = \lceil 0.1m \rceil$.

Матричным профилем [18] рядов A и B называется ряд P_{AB} , где i -й элемент есть расстояние между i -й подпоследовательностью ряда A , имеющей длину ℓ , и ее ближайшим соседом в ряде B :

$$P_{AB} = \{ED_{\text{norm}}^2(A_{i, \ell}, B_{j, \ell})\}_{i=1}^{m-\ell+1}, \quad B_{j, \ell} = \arg \min_{1 \leq q \leq m-\ell+1} ED_{\text{norm}}^2(A_{i, \ell}, B_{q, \ell}), \quad (10)$$

где $ED_{\text{norm}}^2(\cdot, \cdot)$ означает квадрат евклидова расстояния между z -нормализованными подпоследовательностями. Аналогичным образом определяется матричный профиль рассматриваемых рядов, взятых в порядке B и A , и обозначается как P_{BA} .

¹Поскольку $m \ll n$, без ограничения общности можно полагать, что n кратно m и представить ряд как набор не пересекающихся сегментов длины m .

1.3. Диссонансы

Подпоследовательности ряда $T_{i,m}$ и $T_{j,m}$ назовем *не пересекающимися*, если $|i - j| \geq m$. Подпоследовательность D ряда T является *диссонансом* [14], если ее ближайший сосед (ближайшая и не пересекающаяся с ней подпоследовательность) находится на расстоянии не менее чем r (заданный аналитиком порог):

$$\min_{M_D \in T} (\text{Dist}(D, M_D)) \geq r, \quad (11)$$

где $\text{Dist}(\cdot, \cdot)$ — неотрицательная симметричная функция расстояния, а M_D обозначает подпоследовательность ряда, не пересекающуюся с данной подпоследовательностью D .

Обобщение концепции диссонанса на случай многомерного ряда выглядит следующим образом [16]. Под многомерным диссонансом понимается N -мерная подпоследовательность d -мерного временного ряда (где $1 \leq N \leq d$), которая наиболее не похожа на все остальные подпоследовательности N -мерных временных рядов, полученных путем составления всевозможных сочетаний из d рядов по N .

2. Метод обнаружения аномалий во временных рядах

В данном разделе представлен новый метод обнаружения аномальных подпоследовательностей многомерного временного ряда в режиме реального времени, получивший название mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). Ниже в разделе [2.1] дана общая схема метода mDiSSiD. Далее подробно описываются компоненты разработанного метода: Препроцессор, формирующий обучающую выборку для нейросетевой модели, и Детектор аномалий, построенный на основе сямской нейронной сети, — в разделах [2.2] и [2.3] соответственно.

2.1. Общая схема

Разработанный метод mDiSSiD включает в себя два следующих компонента: Препроцессор и Детектор аномалий. На вход Препроцессору поступает репрезентативный фрагмент потокового временного ряда для исследуемой предметной области. При этом предполагается, что данный фрагмент ряда содержит все типичные активности субъекта. Далее каждый одномерный временной ряд, который входит в состав исходного многомерного ряда, подвергается очистке и разметке с помощью разработанного ранее алгоритма формирования обучающей выборки, представленного автором данной статьи в работе [11]. В результате алгоритм для каждого измерения формирует очищенное множество подпоследовательностей с указанием для них метки активности, к которой они принадлежат. Далее в рамках каждого измерения Препроцессор выполняет минимаксную нормализацию подпоследовательностей, приводя их значения к диапазону $[0, 1]$, и формирует обучающую выборку для Детектора, который представляет собой ансамбль модифицированных нейросетевых моделей DiSSiD.

Каждый участник ансамбля обрабатывает соответствующее измерение многомерного временного ряда и определяет, с каким из сниппетов ряда поданная на вход подпоследовательность этого измерения имеет наибольшее семантическое сходство. Затем выполняется агрегация результатов, полученных участниками ансамбля, с целью вычисления оценок аномальности для N -мерных подпоследовательностей, составленных из всевозможных сочетаний из d измерений входной многомерной подпоследовательности по N , где $1 \leq N \leq d$.

Многомерная подпоследовательность считается аномалией, если хотя бы одна из полученных оценок аномальности превышает порог, вычисленный для каждой размерности N на валидационной выборке.

2.2. Препроцессор

Препроцессор выполняет подготовку обучающей выборки для нейросетевой модели Детектора аномалий. Для этого зафиксируем репрезентативный фрагмент временного ряда, который содержит все активности субъекта, и обозначим его за \mathbf{T} . Далее каждое измерение $T^{(i)}$ фрагмента \mathbf{T} подвергается предварительной обработке с помощью разработанного ранее алгоритма формирования обучающей выборки [11]. В качестве входных параметров алгоритма задаются длина подпоследовательности m ($m \ll n$), доля аномальных подпоследовательностей в ряде α ($0 < \alpha \ll 1$), количество снишкетов K ($K > 1$) и пороговая значимость снишкета ϕ ($0 < \phi < 1/K$). При этом для обработки всех измерений ряда устанавливаются одинаковые значения входных параметров. Сначала с помощью параллельного алгоритма PSF [19] выполняется разметка подпоследовательностей измерения многомерного ряда в соответствии с активностями субъекта. Тем самым формируется множество подпоследовательностей $S^{(i)}$, определяемое следующим образом:

$$S^{(i)} = \{\langle T_{j,m}^{(i)}, k \rangle \mid T_{j,m}^{(i)} \in C_k^{(i)}.NN, 1 \leq i \leq d, 1 \leq j \leq n - m + 1, 1 \leq k \leq K\}. \quad (12)$$

Далее из дальнейшей обработки исключаются подпоследовательности-шумы $O^{(i)}$, трактуемые как выбросы в рамках каждого снишкета, и подпоследовательности, соответствующие редко встречающимся активностям $C_{\text{weak}}^{(i)}$:

$$C_{\text{weak}}^{(i)} = \left\{ \{C_k^{(i)} \cup C_k^{(i)}.NN\} \in C_{T^{(i)}}^m \mid C_k^{(i)}.frac \leq \phi, 1 \leq i \leq d, 1 \leq k \leq K \right\}. \quad (13)$$

Во-вторых, с помощью параллельного алгоритма PALMAD [20] выполняется поиск диссонансов (аномальных подпоследовательностей) $D^{(i)}$, $|D^{(i)}| = \alpha \cdot (n - m + 1)$, которые также исключаются из дальнейшей обработки. В завершении выполняется очистка размеченного множества подпоследовательностей $S^{(i)}$ от найденных на предыдущих шагах Препроцессора различного рода аномалий:

$$S^{(i)} = S^{(i)} \setminus \{\langle T_{j,m}^{(i)}, k \rangle\}, \text{ где } T_{j,m}^{(i)} \in \{C_{\text{weak}}^{(i)} \cup O^{(i)} \cup D^{(i)}\}. \quad (14)$$

Затем Препроцессор выполняет min-max нормализацию подпоследовательностей, входящих в множество $S^{(i)}$, для каждого измерения $T^{(i)}$, преобразуя их значения в диапазон $[0, 1]$. Нормализованная версия $\hat{T}_{j,m}^{(i)}$ подпоследовательности $T_{j,m}^{(i)}$ вычисляется следующим образом:

$$\hat{T}_{j,m}^{(i)} = (\hat{t}_j^{(i)}, \dots, \hat{t}_{j+m-1}^{(i)}), 1 \leq i \leq d, \quad (15)$$

$$\text{где } \hat{t}_j^{(i)} = \frac{t_j^{(i)} - \min_{1 \leq p \leq n} t_p^{(i)}}{\max_{1 \leq p \leq n} t_p^{(i)} - \min_{1 \leq p \leq n} t_p^{(i)}}.$$

Далее выполняется процедура генерации обучающей выборки для нейросетевой модели Детектора аномалий. Поскольку предусматривается осуществлять поиск аномалий в каждом измерении независимо друг от друга, то для каждого измерения формируется своя обучающая выборка. Обозначим обучающую выборку для i -го измерения многомерного ряда \mathbf{T}

за $\mathcal{L}^{(i)}$. Обучающая выборка $\mathcal{L}^{(i)}$ представляет собой множество прецедентов вида $\langle X, Y \rangle$. Под X понимается набор пар подпоследовательностей, количество которых совпадает с количеством активностей (снипшетов) во временном ряде. Первый элемент в каждой паре одинаков и представляет собой подпоследовательность $\widehat{T}_{j,m}^{(i)}$ из множества $S^{(i)}$, оставшуюся в результате предобработки. В качестве второго (различающего) элемента пары фигурирует соответствующий снипшет $C_k^{(i)}$ или ближайший сосед этого снипшета, взятый из множества $C_k^{(i)}.NN$. Под Y подразумевается K -мерный бинарный вектор, один элемент которого равен 1, если подпоследовательности в паре отражают одинаковую активность, остальные элементы равны 0 — в противном случае. Таким образом, для каждой предобработанной подпоследовательности измерения можно получить $nnQty$ прецедентов, где $nnQty$ — параметр алгоритма. При этом чем больше устанавливается значение $nnQty$, тем более вариативной становится обучающая выборка. Формальное определение обучающей выборки $\mathcal{L}^{(i)}$ выглядит следующим образом:

$$\begin{aligned} \mathcal{L}^{(i)} = \left\{ \langle X^{(i)}, Y^{(i)} \rangle \mid X^{(i)} = \{ \langle x_1^{(i)}, x_{2_k}^{(i)} \rangle \}_{k=1}^K, \right. \\ x_1^{(i)} = \widehat{T}_{j,m}^{(i)}, \quad x_{2_k}^{(i)} = C_k^{(i)} \vee x_{2_k}^{(i)} = \widehat{T}_{p,m}^{(i)}, \quad \widehat{T}_{j,m}^{(i)}, \widehat{T}_{p,m}^{(i)} \in S^{(i)}, \\ Y^{(i)} = \{ [y_1^{(i)} = y_{2_k}^{(i)}] \}_{k=1}^K, \\ y_1^{(i)} = k_1, \quad x_1^{(i)} \in C_{k_1}^{(i)}.NN, \quad y_{2_k}^{(i)} = k_2, \quad x_{2_k}^{(i)} \in C_{k_2}^{(i)}.NN, \\ \left. 1 \leq i \leq d, 1 \leq j, p \leq n - m + 1 \right\}. \end{aligned} \quad (16)$$

2.3. Детектор аномалий

Нейросетевая модель mDiSSiD представляет собой ансамбль из d модификаций модели DiSSiD (для одномерного ряда), где d — размерность исходного многомерного ряда \mathbf{T} . Задача участника ансамбля заключается в том, что необходимо определить, с каким из снипшетов ряда поданная на вход подпоследовательность назначенного ему измерения имеет наибольшее семантическое сходство. На вход модифицированной модели DiSSiD поступает набор из K пар подпоследовательностей $X^{(i)}$, каждая из которых включает входную подпоследовательность $T_{n-m+1,m}^{(i)}$, оканчивающуюся текущим поступившим значением $t_n^{(i)}$, и один из снипшетов $C_k^{(i)}$. На выходе модель выдает K -мерный вектор $\widehat{Y}^{(i)}$, содержащий оценки схожести между подпоследовательностями в парах. Таким образом, участника ансамбля формально можно определить как $f: \mathbb{R}^{K \times 2 \times m} \rightarrow \mathbb{R}^K$. Изучение соответствующего измерения и выдача результатов каждым участником ансамбля ведется независимо друг от друга. Такая организация обработки обусловлена определением N -мерного диссонанса, требующего выполнять сравнение поданной на вход одномерной подпоследовательности с подпоследовательностями из того же измерения.

На рис. 1 представлена структура участника ансамбля, являющейся общей для всех участников. Участник ансамбля включает K подсетей, каждая из которых представляет собой разработанную ранее модель DiSSiD [11] (см. рис. 2). Модель DiSSiD является сиамской нейронной сетью (Siamese Neural Network, SNN) [12], сконструированной из двух сверточных нейронных сетей ResNet, идентичных по конфигурации и наборам обучаемых параметров (весов и смещений). Сначала на вход DiSSiD поступает соответствующая пара подпоследовательностей $\langle x_1^{(i)}, x_{2_k}^{(i)} \rangle$. Далее сиамская сеть выполняет отображение входных данных в скрытые представления (embedding) $h_1^{(i)}$ и $h_2^{(i)}$ изученного латентного пространства \mathbb{R}^s , имеющие меньшую размерность, чем исходные, $s < m$.

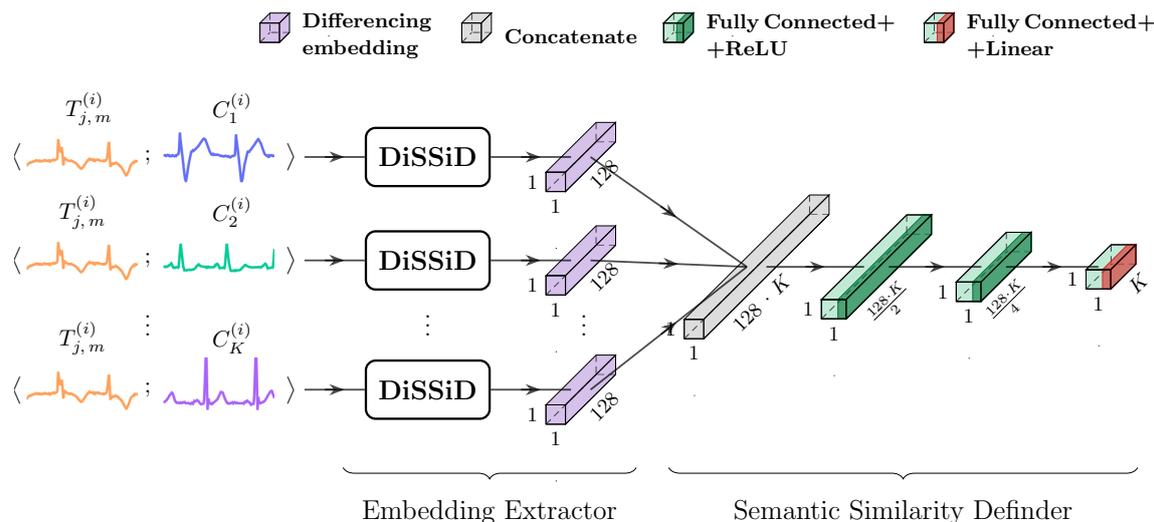


Рис. 1. Структура участника нейросетевой модели mDiSSiD

ResNet включает в себя три одинаковых остаточных блока (residual block), состоящих из трех сверточных слоев с одномерными ядрами свертки размеров 8×1 , 5×1 и 3×1 соответственно. Каждый сверточный слой чередуется с пакетной нормализацией (batch normalization) [21], после которой применяется функция активации Линейный выпрямитель (ReLU, Rectified linear unit). После прохождения всех слоев очередного остаточного блока формируются карты признаков, которые далее складываются с входом этого остаточного блока, пропущенного через сверточный слой с ядром свертки размера 1×1 . Первый остаточный блок формирует 64 карты признаков, остальные два блока — по 128 карт. За остаточными блоками следует слой глобальной усредняющей агрегации (GlobalAveragePooling), формирующий итоговое скрытое представление, i -й элемент которого вычисляется как среднее арифметическое по всем элементам i -й карты признаков.

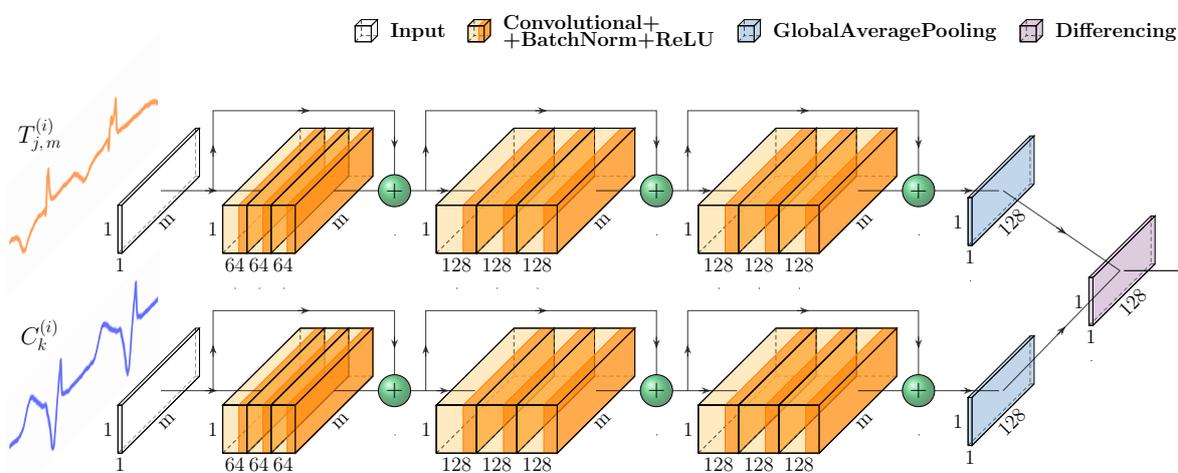


Рис. 2. Нейросетевая модель DiSSiD

Далее на различающем слое (differencing layer) определяется схожесть между сформированными скрытыми представлениями $h_1^{(i)}$ и $h_{2_k}^{(i)}$ путем вычисления вектора $h_{diff_k}^{(i)}$ из расстояний между соответствующими элементами этих скрытых представлений:

$$h_{diff_k}^{(i)}(j) = \text{dist}(h_1^{(i)}(j), h_{2_k}^{(i)}(j)).$$

В данном исследовании в качестве функции расстояния $\text{dist}(\cdot, \cdot)$ выбираются метрики L1 и L2. Затем следует слой конкатенации, выполняющий сцепление результатов, выданных различающими слоями по всем K подсетям участника ансамбля. В результате формируется вектор

$$h_{concat}^{(i)} = h_{diff_1}^{(i)} \bullet \dots \bullet h_{diff_K}^{(i)}, \quad h_{concat}^{(i)} \in \mathbb{R}^{s \times K},$$

где \bullet — операция конкатенации. Замыкают модифицированную модель DiSSiD три полносвязных слоя, которые направлены на вычисление оценок схожести входной подпоследовательности со всеми снippetsами:

$$\hat{Y}^{(i)} = f(X^{(i)}, W_i, B_i),$$

где W_i и B_i — матрицы весов и смещений i -го участника ансамбля соответственно.

На скрытых полносвязных слоях с числом нейронов $(s \cdot K)/2$ и $(s \cdot K)/4$ используется функция активации ReLU, на выходном слое из K нейронов — линейная функция активации. В завершении формируется итоговая оценка для подпоследовательности $T_{n-m+1,m}^{(i)}$, относящейся к соответствующему измерению, путем взятия минимума по всем выданным на последнем слое нейронной сети оценкам:

$$\text{score}(T_{n-m+1,m}^{(i)}) = \min_{1 \leq k \leq K} \hat{Y}_k^{(i)}.$$

Итоговая оценка определяет степень схожести подпоследовательности с наиболее близким к ней снippetом.

Для определения того, является ли входная подпоследовательность $T_{n-m+1,m}$ многомерным диссонансом, сначала формируются N -мерные подпоследовательности путем составления всевозможных сочетаний из d измерений этой подпоследовательности по N , где $1 \leq N \leq d$. Обозначим множество всех подпоследовательностей, содержащих различное количество измерений от 1 до d , за \mathcal{P} . Данное множество \mathcal{P} является булеаном (показательным множеством) и определяется следующим образом:

$$\begin{aligned} \mathcal{P} &= \{1\text{-}\mathcal{P}, 2\text{-}\mathcal{P}, \dots, d\text{-}\mathcal{P}\}, \text{ где} \\ 1\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(i)}\} \mid 1 \leq i \leq d\}, \\ 2\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(i)}, T_{n-m+1,m}^{(j)}\} \mid 1 \leq i, j \leq d, i < j\}, \\ &\dots \\ d\text{-}\mathcal{P} &= \{\{T_{n-m+1,m}^{(1)}, \dots, T_{n-m+1,m}^{(d)}\}\}. \end{aligned}$$

Количество сгенерированных многомерных подпоследовательностей в множестве \mathcal{P} составляет $|\mathcal{P}| = \sum_{N=1}^d C_d^N = \sum_{N=1}^d \frac{d!}{(d-N)!N!} = 2^d - 1$.

Затем для каждой N -мерной подпоследовательности $N\text{-}\mathcal{P}_i$ определяется ее оценка схожести $N\text{-score}(N\text{-}\mathcal{P}_i)$. Для этого формируется набор из итоговых оценок $\{\text{score}(N\text{-}\mathcal{P}_{i,j})\}_{j=1}^N$,

полученных нейросетевой моделью mDiSSiD на предыдущем шаге для входящих в это сочетание элементов (одномерных подпоследовательностей). Далее среди всех имеющихся в данном наборе оценок выбирается минимальная, которая назначается в качестве результирующей оценки схожести для N -мерной подпоследовательности $N-\mathcal{P}_i$:

$$N\text{-score}(N-\mathcal{P}_i) = \min_{1 \leq j \leq N} \text{score}(N-\mathcal{P}_{i,j}).$$

После этого для каждой группы сочетаний, объединенных по равному числу входящих в них элементов, вычисляется оценка аномальности путем взятия максимального значения по оценкам схожести, полученных для всех сочетаний из одной группы:

$$N\text{-anomaly}(\mathbf{T}_{n-m+1,m}) = \max_{1 \leq i \leq |N-\mathcal{P}|} N\text{-Score}(N-\mathcal{P}_i).$$

Максимум означает то, что в качестве определяющего сочетания в группе выбирается наименее похожее. На основании этого сочетания будет делаться вывод о том, является ли многомерная подпоследовательность нормой или аномалией. Входная многомерная подпоследовательность $\mathbf{T}_{n-m+1,m}$ считается многомерным диссонансом, если хотя бы одна из оценок аномальности $N\text{-anomaly}$ превышает некоторый установленный порог, $1 \leq N \leq d$.

Порог для каждого значения N формируется путем взятия k -го перцентиля по набору оценок аномальности, полученных для сочетаний из N измерений многомерных подпоследовательностей валидационной выборки. В данном исследовании в качестве порога применяется значение $k = 95$.

3. Вычислительные эксперименты

В данном разделе представлены результаты вычислительных экспериментов, проведенных на реальных временных рядах, которые имеют истинную разметку аномалий. В экспериментах выполняется сравнение точности предлагаемого метода mDiSSiD с аналогами, относящихся к методам с частичным привлечением учителя. Помимо этого, исследуется влияние функции расстояния между векторными представлениями входных подпоследовательностей (метрики L1 и L2) на эффективность обнаружения аномалий с помощью метода mDiSSiD.

3.1. Параметры экспериментов

Наборы данных. Временные ряды, использованные в экспериментах, взяты из реальных предметных областей и резюмированы в табл. [1](#). Данные взяты из общедоступного фреймворка TimeEval [22](#), предназначенного для проведения вычислительных экспериментов с алгоритмами обнаружения аномалий в многомерных временных рядах.

Ряд OPPORTUNITY (OPP) [23](#) представляет собой показания носимых датчиков движения, закрепленных на человеке, выполняющего в течение 2 час. различные виды утренней повседневной активности: подъем, перемещение по комнате, приготовление завтрака, уборка и отдых.

Ряд Daphnet [24](#) содержит показания виброакселерометров, закрепленных на пациенте с болезнью Паркинсона, для обнаружения у него симптомов замирания походки. Пациент выполнял следующие виды физической активности: ходьба по прямой, ходьба с поворотами, перемещения между разными комнатами с открыванием дверей и др.

Каждый из рядов MITDB-2 (105, 217) и MITDB-3 (105, 116, 217) [25] представляет собой конкатенацию ЭКГ реальных пациентов, страдающих нарушениями сердечного ритма. Каждая из сцепленных ЭКГ отражает одну активность пациента.

Таблица 1. Временные ряды для вычислительных экспериментов

№ п/п	Временной ряд	Длина ряда n	Размерность ряда d	Длина снippetsа m	Длина значимого участка ℓ	Кол-во снippetsов K	Доля аномалий $\alpha, \times 10^{-4}$
1	OPP	26 204	6	200	50	2	5
2	Daphnet	28 800	3	216	72	2	5
3	MITDB-2	100 000	2	600	75	2	2
4	MITDB-3	150 000	2	600	75	3	3

Аналоги. В экспериментах разработанная модель сравнивалась со следующими аналогами, принадлежащими, как и mDiSSiD, к группе методов обнаружения аномалий с частичным привлечением учителя: LSTM-AD [9], AE [26], DeepAnT [10], TAnoGAN [27]. Реализация указанных методов взята из работы [22].

Метрики сравнения. Для оценки качества обнаружения аномалий используется метрика VUS-PR [28], интегрирует в себе как стандартные метрики — точность (precision) и полноту (recall), так и величину смещения найденной аномальной подпоследовательности относительно истинной аномалии. Метрика VUS-PR принимает значения из отрезка $[0, 1]$, большему значению соответствует лучшее качество.

Аппаратная платформа. Вычислительные эксперименты выполнялись на вычислительном узле комплекса «Нейрокомпьютер» Суперкомпьютерного центра ЮУрГУ [29], оснащенного графическим процессором NVIDIA Tesla V100 SXM2 (5120 ядер @1.3 ГГц).

Обучение модели. Для обучения нейросетевой модели mDiSSiD (участника ансамбля) предлагается следующая функция контрастных потерь (contrastive loss) [30], адаптированная под многомерный случай:

$$L(\langle X_j^{(i)}, Y_j^{(i)} \rangle, W_i, B_i) = \sum_{k=1}^K \left(y_{j,k}^{(i)} \cdot (f(x_{j,k}^{(i)}, W_i, B_i))^2 + (1 - y_{j,k}^{(i)}) \cdot (\max(\tau - f(x_{j,k}^{(i)}, W_i, B_i), 0))^2 \right), \quad (17)$$

где i — номер участника ансамбля, $\langle X_j^{(i)}, Y_j^{(i)} \rangle$ — прецедент выборки, K — количество снippetsов, τ — минимальное расстояние между векторными представлениями исходных подпоследовательностей, являющихся ближайшими соседями разных снippetsов (параметр модели). Указанная функция потерь обеспечивает обучение модели, в результате которого похожие подпоследовательности исходного ряда получают векторные представления, отстоящие друг от друга в смысле расстояния не более чем на τ , а непохожие — более чем на τ соответственно.

Перед обучением элементы множества $\mathcal{L}^{(i)}$ случайным образом разделяются на два не пересекающихся подмножества: обучающую и валидационную выборки $\mathcal{L}_{\text{train}}^{(i)}$ и $\mathcal{L}_{\text{valid}}^{(i)}$, используемые для обучения модели и настройки ее гиперпараметров соответственно. Мощности указанных выборок находятся в традиционном соотношении 80% и 20% соответственно.

Эксперименты выполнялись при следующих установленных значениях гиперпараметров: скорость обучения $\mu = 1 \cdot 10^{-3}$, количество эпох $epochs = 50$, размер батча $batch_size = 128$, отступ $\tau = 1$. В качестве оптимизатора стохастического градиентного спуска использовался Adam [31].

3.2. Результаты

Таблица 2. Сравнение точности метода mDiSSiD с аналогами (метрика VUS-PR)

Методы Ряды	AE	DeepAnT	LSTM-AD	TAnoGAN	mDiSSiD (L1)	mDiSSiD (L2)
OPP	0.4812 (2)	0.2105 (6)	0.2178 (5)	0.7632 (1)	0.4154 (4)	0.4273 (3)
Daphnet	0.3569 (4)	0.3817 (3)	0.2347 (5)	0.2105 (6)	0.4356 (2)	0.4632 (1)
MITDB-2	0.5181 (4)	0.4514 (5)	0.4153 (6)	0.5208 (3)	0.6467 (2)	0.6917 (1)
MITDB-3	0.4496 (3)	0.3189 (6)	0.3284 (5)	0.4265 (4)	0.5898 (2)	0.6250 (1)
Средний VUS-PR	0.4515 (4)	0.3406 (5)	0.2990 (6)	0.4803 (3)	0.5219 (2)	0.5518 (1)
Средний ранг	3.25 (4)	5.00 (5)	5.25 (6)	3.50 (3)	2.50 (2)	1.50 (1)

Результаты сравнения точности метода mDiSSiD с аналогами представлены в табл. 2. В ячейке таблицы дано значение меры VUS-PR и в скобках — ранг метода, указанного в соответствующем столбце, среди всех аналогов на временном ряде, указанном в соответствующей строке. Полужирным шрифтом даны результат и место лучшего метода на заданном временном ряде. Две последние строки таблицы являются резюмирующими, в них указаны соответственно средние значения метрики и ранга по всем рядам, а также среднее значение метрики и ранга метода в скобках.

Можно видеть, что при применении метрики L1 в на различающемся слое нейронной сети участника ансамбля метод mDiSSiD в среднем входит в тройку лучших методов по точности обнаружения аномалий. При этом при использовании метрики L2 позволяет добиться лучшей в среднем точности обнаружения аномалий. Это связано с тем, что метрика L2 более чувствительна к выбросам, поэтому она делает более важный и больший акцент на тех парах скрытых представлений на различающей слое, которые соответствуют подпоследовательностям, принадлежащим разным активностям, или когда входная подпоследовательность является аномалией.

Заключение

В статье рассмотрена задача детекции аномальных подпоследовательностей многомерного потокового временного ряда, элементы которого поступают в режиме реального времени, востребованная в настоящее время в широком спектре предметных областей: промышленный Интернет вещей, персональное здравоохранение и др. Описан новый метод решения указанной задачи, получивший название mDiSSiD (multivariate DiSSiD), который обобщает на многомерный случай предложенный автором ранее метод DiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of anomalies) для детекции аномалий в одномерном потоковом временном ряде.

Предложенный метод использует концепцию диссонанса временного ряда (подпоследовательность, имеющая наиболее не похожего на нее ближайшего соседа) [14], обобщенную на многомерный случай. Под многомерным диссонансом понимается [16] N -мерная подпоследовательность d -мерного временного ряда (где $1 \leq N \leq d$), которая наиболее не похожа на все остальные подпоследовательности N -мерных временных рядов, полученных путем составления всевозможных сочетаний из d рядов по N .

Детекция аномалий реализуется с помощью нейросетевой модели на основе сиамских нейросетей. Модель mDiSSiD представляет собой ансамбль из d модификаций модели DiSSiD. Входной слой модифицированной модели DiSSiD представляет собой K пар, где K — количество различаемых активностей исследуемого субъекта, в которых первой компонентой является входная подпоследовательность, а второй — снippet либо один из его ближайших соседей. Далее следуют слои, предусмотренные моделью DiSSiD: остаточные блоки ResNet, слой GlobalAveragePooling и различающий слой. После этого следует слой конкатенации, выполняющий сцепление результатов, выданных различающими слоями по всем измерениям. Замыкают модифицированную модель три полносвязных слоя, обеспечивающие оценки схожести входной подпоследовательности со всеми снippetами. Затем формируется итоговая оценка для подпоследовательности, относящейся к соответствующему измерению, путем взятия минимума по всем выданным на последнем слое нейронной сети оценкам. Итоговая оценка определяет степень схожести подпоследовательности с наиболее близким к ней снippetом.

Вычислительные эксперименты на реальных временных рядах из различных предметных областей (ЭКГ, показания виброакселерометра во время физической активности, мониторинг нагрузки серверов баз данных) показали, что метод mDiSSiD в среднем опережает по точности обнаружения аномалий передовые аналоги, использующие иные нейросетевые архитектуры: сверточные и рекуррентные нейронные сети, автоэнкодеры, генеративно-состязательные сети.

Работа выполнена при финансовой поддержке Российского научного фонда (грант № 23-21-00465).

Литература

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data // ACM Comput. Surv. 2021. Vol. 54, no. 3. 56:1–56:33. DOI: [10.1145/3444690](https://doi.org/10.1145/3444690).
2. Kumar S., Tiwari P., Zymbler M.L. Internet of Things is a revolutionary approach for future technology enhancement: a review // J. Big Data. 2019. Vol. 6. P. 111. DOI: [10.1186/s40537-019-0268-2](https://doi.org/10.1186/s40537-019-0268-2).
3. Цымблер М.Л., Краева Я.А., Латыпова Е.А. и др. Очистка сенсорных данных в интеллектуальных системах управления отоплением зданий // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 3. С. 16–36. DOI: [10.14529/cmse210302](https://doi.org/10.14529/cmse210302).
4. Иванов С.А., Никольская К.Ю., Радченко Г.И. и др. Концепция построения цифрового двойника города // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2020. Т. 9, № 4. С. 5–23. DOI: [10.14529/cmse200401](https://doi.org/10.14529/cmse200401).
5. Volkov I., Radchenko G.I., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: A Review of Current Platforms to Support Smart Healthcare // Program. Comput. Softw. 2021. Vol. 47, no. 8. P. 578–590. DOI: [10.1134/S0361768821080284](https://doi.org/10.1134/S0361768821080284).
6. Schmidl S., Wenig P., Papenbrock T. Anomaly Detection in Time Series: A Comprehensive Evaluation // Proc. VLDB Endow. 2022. Vol. 15, no. 9. P. 1779–1797. URL: <https://www.vldb.org/pvldb/vol15/p1779-wenig.pdf>.

7. Paparrizos J., Kang Y., Boniol P., *et al.* TSB-UAD: An End-to-End Benchmark Suite for Univariate Time-Series Anomaly Detection // Proc. VLDB Endow. 2022. Vol. 15, no. 8. P. 1697–1711. URL: <https://www.vldb.org/pvldb/vol15/p1697-paparrizos.pdf>.
8. Hodge V.J., Austin J. A Survey of Outlier Detection Methodologies // Artif. Intell. Rev. 2004. Vol. 22, no. 2. P. 85–126. DOI: [10.1023/B:AIRE.0000045502.10941.a9](https://doi.org/10.1023/B:AIRE.0000045502.10941.a9).
9. Malhotra P., Vig L., Shroff G., Agarwal P. Long Short Term Memory Networks for Anomaly Detection in Time Series // 23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015. 2015. URL: <https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf>.
10. Munir M., Siddiqui S.A., Dengel A., Ahmed S. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series // IEEE Access. 2019. Vol. 7. P. 1991–2005. DOI: [10.1109/ACCESS.2018.2886457](https://doi.org/10.1109/ACCESS.2018.2886457).
11. Краева Я.А. Обнаружение аномалий временного ряда на основе технологий интеллектуального анализа данных и нейронных сетей // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2023. Т. 12, № 3. С. 50–71. DOI: [10.14529/cmse230304](https://doi.org/10.14529/cmse230304).
12. Chicco D. Siamese Neural Networks: An Overview // Artificial Neural Networks / ed. by H. Cartwright. New York, NY: Springer US, 2021. P. 73–94. DOI: [10.1007/978-1-0716-0826-5_3](https://doi.org/10.1007/978-1-0716-0826-5_3).
13. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition // 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, 2016. P. 770–778. DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
14. 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), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: [10.1109/ICDM.2007.61](https://doi.org/10.1109/ICDM.2007.61).
15. Imani S., Madrid F., Ding W., *et al.* Matrix Profile XIII: Time Series Snippets: A New Primitive for Time Series Data Mining // 2018 IEEE International Conference on Big Knowledge, ICBK 2018, Singapore, November 17-18, 2018 / ed. by X. Wu, Y. Ong, C.C. Aggarwal, H. Chen. IEEE Computer Society, 2018. P. 382–389. DOI: [10.1109/ICBK.2018.00058](https://doi.org/10.1109/ICBK.2018.00058).
16. Tafazoli S., Keogh E.J. Matrix Profile XXVIII: Discovering Multi-Dimensional Time Series Anomalies with K of N Anomaly Detection[†] // Proceedings of the 2023 SIAM International Conference on Data Mining, SDM 2023, Minneapolis-St. Paul Twin Cities, MN, USA, April 27-29, 2023 / ed. by S. Shekhar, Z. Zhou, Y. Chiang, G. Stiglic. SIAM, 2023. P. 685–693. DOI: [10.1137/1.9781611977653.CH77](https://doi.org/10.1137/1.9781611977653.CH77).
17. Gharghabi S., Imani S., Bagnall A.J., *et al.* An ultra-fast time series distance measure to allow data mining in more complex real-world deployments // Data Min. Knowl. Discov. 2020. Vol. 34, no. 4. P. 1104–1135. DOI: [10.1007/s10618-020-00695-8](https://doi.org/10.1007/s10618-020-00695-8).

18. Yeh C.M., Zhu Y., Ulanova L., *et al.* Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets // IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain / ed. by F. Bonchi, J. Domingo-Ferrer, R. Baeza-Yates, *et al.* IEEE Computer Society, 2016. P. 1317–1322. DOI: [10.1109/ICDM.2016.0179](https://doi.org/10.1109/ICDM.2016.0179).
19. Zymbler M., Goglavchev A. Fast Summarization of Long Time Series with Graphics Processor // Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: [10.3390/math10101781](https://doi.org/10.3390/math10101781).
20. Zymbler M., Kraeva Y. High-Performance Time Series Anomaly Discovery on Graphics Processors // Mathematics. 2023. Vol. 11, no. 14. P. 3193. DOI: [10.3390/math11143193](https://doi.org/10.3390/math11143193).
21. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Vol. 37 / ed. by F.R. Bach, D.M. Blei. JMLR.org, 2015. P. 448–456. JMLR Workshop and Conference Proceedings. URL: <http://proceedings.mlr.press/v37/ioffe15.html>.
22. Wenig P., Schmidl S., Papenbrock T. TimeEval: A Benchmarking Toolkit for Time Series Anomaly Detection Algorithms // Proc. VLDB Endow. 2022. Vol. 15, no. 12. P. 3678–3681. URL: <https://www.vldb.org/pvldb/vol15/p3678-schmidl.pdf>.
23. Roggen D., Calatroni A., Rossi M., *et al.* Collecting complex activity datasets in highly rich networked sensor environments // Seventh International Conference on Networked Sensing Systems, INSS 2010, Kassel, Germany, June 15-18, 2010. IEEE, 2010. P. 233–240. DOI: [10.1109/INSS.2010.5573462](https://doi.org/10.1109/INSS.2010.5573462).
24. Bächlin M., Plotnik M., Roggen D., *et al.* Wearable assistant for Parkinson’s disease patients with the freezing of gait symptom // IEEE Trans. Inf. Technol. Biomed. 2010. Vol. 14, no. 2. P. 436–446. DOI: [10.1109/TITB.2009.2036165](https://doi.org/10.1109/TITB.2009.2036165).
25. Moody G., Mark R. The impact of the MIT-BIH Arrhythmia Database // IEEE Engineering in Medicine and Biology Magazine. 2001. Vol. 20, no. 3. P. 45–50. DOI: [10.1109/51.932724](https://doi.org/10.1109/51.932724).
26. Sakurada M., Yairi T. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction // Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, Gold Coast, Australia, QLD, Australia, December 2, 2014 / ed. by A. Rahman, J.D. Deng, J. Li. ACM, 2014. P. 4. DOI: [10.1145/2689746.2689747](https://doi.org/10.1145/2689746.2689747).
27. Bashar M.A., Nayak R. TAnoGAN: Time Series Anomaly Detection with Generative Adversarial Networks // 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020. IEEE, 2020. P. 1778–1785. DOI: [10.1109/SSCI47803.2020.9308512](https://doi.org/10.1109/SSCI47803.2020.9308512).
28. Paparrizos J., Boniol P., Palpanas T., *et al.* Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection // Proc. VLDB Endow. 2022. Vol. 15, no. 11. P. 2774–2787. URL: <https://www.vldb.org/pvldb/vol15/p2774-paparrizos.pdf>.
29. Биленко Р.В., Долганина Н.Ю., Иванова Е.В., Рекачинский А.И. Высокопроизводительные вычислительные ресурсы Южно-Уральского государственного университета // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2022. Т. 11, № 1. С. 15–30. DOI: [10.14529/cmse220102](https://doi.org/10.14529/cmse220102).

30. Hadsell R., Chopra S., LeCun Y. Dimensionality Reduction by Learning an Invariant Mapping // 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA. IEEE Computer Society, 2006. P. 1735–1742. DOI: [10.1109/CVPR.2006.100](https://doi.org/10.1109/CVPR.2006.100).
31. Kingma D., Ba J. Adam: A Method for Stochastic Optimization // International Conference on Learning Representations (ICLR). San Diego, CA, USA, 2015.

Краева Яна Александровна, к.ф.-м.н., старший преподаватель, кафедра системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DOI: [10.14529/cmse240403](https://doi.org/10.14529/cmse240403)

DEEP LEARNING METHOD FOR ANOMALY DETECTION IN STREAMING MULTIVARIATE TIME SERIES

© 2024 Ya.A. Kraeva

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

E-mail: kraevaya@susu.ru

Received: 20.10.2024

The article touches upon the problem of detecting anomalous subsequences of multivariate streaming time series, where the elements arrive in real time, which currently arises in a wide range of subject domains: industrial Internet of Things, personal healthcare, etc. In the article, we introduce a novel method to solve such a problem, called mDiSSiD (Discord, Snippet, and Siamese Neural Network-based Detector of multivariate anomalies). The mDiSSiD method employs the time series discord concept (a subsequence with the most dissimilar nearest neighbor), which is generalized to the multivariate case. Multivariate discord refers to the N -dimensional subsequence of a d -dimensional time series (where $1 \leq N \leq d$), which is the most dissimilar to all other subsequences of N -dimensional time series obtained by composing all the possible combinations of d series of N . Anomaly detection is implemented through a deep learning model based on the Siamese neural network architecture. Experimental evaluation of mDiSSiD over real time series from various subject domains showed that the proposed method is on average ahead of state-of-the-art analogs based on other deep learning approaches (convolutional and recurrent neural networks, autoencoders, and generative-adversarial networks) in terms of anomaly detection accuracy.

Keywords: multivariate time series, anomaly detection, discord, snippet, Siamese neural network.

FOR CITATION

Kraeva Ya.A. Deep Learning Method for Anomaly Detection in Streaming Multivariate Time Series. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2024. Vol. 13, no. 4. P. 35–52. (in Russian) DOI: [10.14529/cmse240403](https://doi.org/10.14529/cmse240403).

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Blázquez-García A., Conde A., Mori U., Lozano J.A. A Review on Outlier/Anomaly Detection in Time Series Data. ACM Comput. Surv. 2021. Vol. 54, no. 3. 56:1–56:33. DOI: [10.1145/3444690](https://doi.org/10.1145/3444690).

2. Kumar S., Tiwari P., Zymbler M.L. Internet of Things is a revolutionary approach for future technology enhancement: a review. *J. Big Data*. 2019. Vol. 6. P. 111. DOI: [10.1186/s40537-019-0268-2](https://doi.org/10.1186/s40537-019-0268-2).
3. Zymbler M.L., Kraeva Y.A., Latypova E.A., *et al.* Cleaning Sensor Data in Intelligent Heating Control System. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2021. Vol. 10, no. 3. P. 16–36. (in Russian) DOI: [10.14529/cmse210302](https://doi.org/10.14529/cmse210302).
4. Ivanov S.A., Nikolskaya K.Y., Radchenko G.I., *et al.* 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](https://doi.org/10.14529/cmse200401).
5. Volkov I., Radchenko G.I., Tchernykh A. Digital Twins, Internet of Things and Mobile Medicine: A Review of Current Platforms to Support Smart Healthcare. *Program. Comput. Softw.* 2021. Vol. 47, no. 8. P. 578–590. DOI: [10.1134/S0361768821080284](https://doi.org/10.1134/S0361768821080284).
6. Schmidl S., Wenig P., Papenbrock T. Anomaly Detection in Time Series: A Comprehensive Evaluation. *Proc. VLDB Endow.* 2022. Vol. 15, no. 9. P. 1779–1797. URL: <https://www.vldb.org/pvldb/vol15/p1779-wenig.pdf>.
7. Paparrizos J., Kang Y., Boniol P., *et al.* TSB-UAD: An End-to-End Benchmark Suite for Univariate Time-Series Anomaly Detection. *Proc. VLDB Endow.* 2022. Vol. 15, no. 8. P. 1697–1711. URL: <https://www.vldb.org/pvldb/vol15/p1697-paparrizos.pdf>.
8. Hodge V.J., Austin J. A Survey of Outlier Detection Methodologies. *Artif. Intell. Rev.* 2004. Vol. 22, no. 2. P. 85–126. DOI: [10.1023/B:AIRE.0000045502.10941.a9](https://doi.org/10.1023/B:AIRE.0000045502.10941.a9).
9. Malhotra P., Vig L., Shroff G., Agarwal P. Long Short Term Memory Networks for Anomaly Detection in Time Series. 23rd European Symposium on Artificial Neural Networks, ESANN 2015, Bruges, Belgium, April 22-24, 2015. 2015. URL: <https://www.esann.org/sites/default/files/proceedings/legacy/es2015-56.pdf>.
10. Munir M., Siddiqui S.A., Dengel A., Ahmed S. DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. *IEEE Access*. 2019. Vol. 7. P. 1991–2005. DOI: [10.1109/ACCESS.2018.2886457](https://doi.org/10.1109/ACCESS.2018.2886457).
11. Kraeva Y.A. Detection of Time Series Anomalies Based on Data Mining and Neural Network Technologies. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2023. Vol. 12, no. 3. P. 50–71. (in Russian) DOI: [10.14529/cmse230304](https://doi.org/10.14529/cmse230304).
12. Chicco D. Siamese Neural Networks: An Overview. *Artificial Neural Networks / ed. by H. Cartwright*. New York, NY: Springer US, 2021. P. 73–94. DOI: [10.1007/978-1-0716-0826-5_3](https://doi.org/10.1007/978-1-0716-0826-5_3).
13. He K., Zhang X., Ren S., Sun J. Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. IEEE Computer Society, 2016. P. 770–778. DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).

14. 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), October 28-31, 2007, Omaha, Nebraska, USA. 2007. P. 381–390. DOI: [10.1109/ICDM.2007.61](https://doi.org/10.1109/ICDM.2007.61).
15. Imani S., Madrid F., Ding W., *et al.* Matrix Profile XIII: Time Series Snippets: A New Primitive for Time Series Data Mining. 2018 IEEE International Conference on Big Knowledge, ICBK 2018, Singapore, November 17-18, 2018 / ed. by X. Wu, Y. Ong, C.C. Aggarwal, H. Chen. IEEE Computer Society, 2018. P. 382–389. DOI: [10.1109/ICBK.2018.00058](https://doi.org/10.1109/ICBK.2018.00058).
16. Tafazoli S., Keogh E.J. Matrix Profile XXVIII: Discovering Multi-Dimensional Time Series Anomalies with K of N Anomaly Detection[†]. Proceedings of the 2023 SIAM International Conference on Data Mining, SDM 2023, Minneapolis-St. Paul Twin Cities, MN, USA, April 27-29, 2023 / ed. by S. Shekhar, Z. Zhou, Y. Chiang, G. Stiglic. SIAM, 2023. P. 685–693. DOI: [10.1137/1.9781611977653.CH77](https://doi.org/10.1137/1.9781611977653.CH77).
17. Gharghabi S., Imani S., Bagnall A.J., *et al.* An ultra-fast time series distance measure to allow data mining in more complex real-world deployments. Data Min. Knowl. Discov. 2020. Vol. 34, no. 4. P. 1104–1135. DOI: [10.1007/s10618-020-00695-8](https://doi.org/10.1007/s10618-020-00695-8).
18. Yeh C.M., Zhu Y., Ulanova L., *et al.* Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View That Includes Motifs, Discords and Shapelets. IEEE 16th International Conference on Data Mining, ICDM 2016, December 12-15, 2016, Barcelona, Spain / ed. by F. Bonchi, J. Domingo-Ferrer, R. Baeza-Yates, *et al.* IEEE Computer Society, 2016. P. 1317–1322. DOI: [10.1109/ICDM.2016.0179](https://doi.org/10.1109/ICDM.2016.0179).
19. Zymbler M., Goglavchev A. Fast Summarization of Long Time Series with Graphics Processor. Mathematics. 2022. Vol. 10, no. 10. P. 1781. DOI: [10.3390/math10101781](https://doi.org/10.3390/math10101781).
20. Zymbler M., Kraeva Y. High-Performance Time Series Anomaly Discovery on Graphics Processors. Mathematics. 2023. Vol. 11, no. 14. P. 3193. DOI: [10.3390/math11143193](https://doi.org/10.3390/math11143193).
21. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. Vol. 37 / ed. by F.R. Bach, D.M. Blei. JMLR.org, 2015. P. 448–456. JMLR Workshop and Conference Proceedings. URL: <http://proceedings.mlr.press/v37/ioffe15.html>.
22. Wenig P., Schmidl S., Papenbrock T. TimeEval: A Benchmarking Toolkit for Time Series Anomaly Detection Algorithms. Proc. VLDB Endow. 2022. Vol. 15, no. 12. P. 3678–3681. URL: <https://www.vldb.org/pvldb/vol15/p3678-schmidl.pdf>.
23. Roggen D., Calatroni A., Rossi M., *et al.* Collecting complex activity datasets in highly rich networked sensor environments. Seventh International Conference on Networked Sensing Systems, INSS 2010, Kassel, Germany, June 15-18, 2010. IEEE, 2010. P. 233–240. DOI: [10.1109/INSS.2010.5573462](https://doi.org/10.1109/INSS.2010.5573462).
24. Bächlin M., Plotnik M., Roggen D., *et al.* Wearable assistant for Parkinson’s disease patients with the freezing of gait symptom. IEEE Trans. Inf. Technol. Biomed. 2010. Vol. 14, no. 2. P. 436–446. DOI: [10.1109/TITB.2009.2036165](https://doi.org/10.1109/TITB.2009.2036165).
25. Moody G., Mark R. The impact of the MIT-BIH Arrhythmia Database. IEEE Engineering in Medicine and Biology Magazine. 2001. Vol. 20, no. 3. P. 45–50. DOI: [10.1109/51.932724](https://doi.org/10.1109/51.932724).

26. Sakurada M., Yairi T. Anomaly Detection Using Autoencoders with Nonlinear Dimensionality Reduction. Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis, Gold Coast, Australia, QLD, Australia, December 2, 2014 / ed. by A. Rahman, J.D. Deng, J. Li. ACM, 2014. P. 4. DOI: [10.1145/2689746.2689747](https://doi.org/10.1145/2689746.2689747).
27. Bashar M.A., Nayak R. TAnoGAN: Time Series Anomaly Detection with Generative Adversarial Networks. 2020 IEEE Symposium Series on Computational Intelligence, SSCI 2020, Canberra, Australia, December 1-4, 2020. IEEE, 2020. P. 1778–1785. DOI: [10.1109/SSCI47803.2020.9308512](https://doi.org/10.1109/SSCI47803.2020.9308512).
28. Paparrizos J., Boniol P., Palpanas T., *et al.* Volume Under the Surface: A New Accuracy Evaluation Measure for Time-Series Anomaly Detection. Proc. VLDB Endow. 2022. Vol. 15, no. 11. P. 2774–2787. URL: <https://www.vldb.org/pvldb/vol15/p2774-paparrizos.pdf>.
29. Bilenko R.V., Dolganina N.Y., Ivanova E.V., Rekachinsky A.I. High-performance Computing Resources of South Ural State University. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2022. Vol. 11, no. 1. P. 15–30. (in Russian) DOI: [10.14529/cmse220102](https://doi.org/10.14529/cmse220102).
30. Hadsell R., Chopra S., LeCun Y. Dimensionality Reduction by Learning an Invariant Mapping. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA. IEEE Computer Society, 2006. P. 1735–1742. DOI: [10.1109/CVPR.2006.100](https://doi.org/10.1109/CVPR.2006.100).
31. Kingma D., Ba J. Adam: A Method for Stochastic Optimization. International Conference on Learning Representations (ICLR). San Diego, CA, USA, 2015.

ОБ ОДНОЙ ФУНКЦИИ ПОТЕРЬ ДЛЯ ОБУЧЕНИЯ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ ВОССТАНОВЛЕНИЯ ВРЕМЕННЫХ РЯДОВ

© 2024 А.А. Юртин

Южно-Уральский государственный университет

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

E-mail: iurtinaa@susu.ru

Поступила в редакцию: 01.09.2024

В статье рассмотрена проблема выбора функции потерь для обучения нейросетевых моделей восстановления пропущенных значений многомерных временных рядов и предложена новая функция потерь, названная MPDE (Mean Profile Distance Error, средняя ошибка профиля расстояния). MPDE для истинной и восстановленной подпоследовательностей ряда, имеющих длину m , вычисляется как среднее значение расстояний между всеми парами окон (непрерывных промежутков) этих подпоследовательностей, имеющими длину ℓ , где $\ell \leq m$ и окна имеют одинаковые начальные индексы. Расстояние между двумя окнами представляет собой модификацию меры схожести MPdist (расстояние матричного профиля) и определяется как взвешенная сумма евклидова и z-нормированного евклидова расстояний между данными окнами. Веса складываемых берутся из отрезка $[0,1]$ и являются параметрами функции потерь. Функция MPDE позволяет при обучении нейросетевой модели учитывать поведенческое сходство сравниваемых подпоследовательностей, учитывая наличие в них сходных окон независимо от мест взаимного расположения этих окон. Функция потерь MPDE имеет высокую вычислительную сложность, поэтому для ее внедрения в фреймворки глубокого обучения разработан параллельный алгоритм, вычисляющий MPDE на графическом процессоре. Алгоритм реализован с помощью фреймворка PyTorch, который позволяет имплементировать MPDE как последовательность автоматически распараллеливаемых операций с многомерными тензорами. Эксперименты на многомерных временных рядах из различных предметных областей показали, что в 78% случаев передовые нейросетевые модели достигают наиболее высокой точности восстановления (по метрике RMSE) при использовании предложенной функции потерь; при этом модели демонстрируют точность восстановления на 40% выше среднего значения, достигнутого при использовании других функций потерь.

Ключевые слова: временной ряд, восстановление пропущенных значений, функция потерь, нейронные сети, PyTorch.

ОБРАЗЕЦ ЦИТИРОВАНИЯ

Юртин А.А. Об одной функции потерь для обучения нейросетевых моделей восстановления временных рядов // Вестник ЮУрГУ. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 4. С. 53–73. DOI: 10.14529/cmse240404.

Введение

В настоящее время интеллектуальная обработка временных рядов играет важную роль в различных предметных областях: экономика [1], энергетика [2], медицина [3], метеорология [4] и др. Одной из ключевых проблем интеллектуального анализа временных рядов является наличие в них пропущенных значений, которые могут возникать из-за неисправности сенсоров, потерь данных в процессе передачи или человеческого фактора. Пропуски в данных существенно затрудняют интеллектуальную обработку временных рядов, поскольку соответствующие методы и алгоритмы, как правило, требуют полной и непрерывной последовательности значений. За последние годы значительные успехи в области разработки эффективных методов восстановления пропущенных значений во временных рядах достигнуты благодаря применению нейросетевых моделей [5, 6]. Современные методы вос-

становления временных рядов способны учитывать временные зависимости и сложные поведенческие шаблоны в данных, повышая тем самым точность восстановления, на основе применения разнообразных нейросетевых архитектур: рекуррентные сети [7] [8], автоэнкодеры [9], трансформеры [10] [11] и др. При применении в восстановлении временных рядов нейросетевых моделей одним из ключевых факторов, влияющих на точность, является функция потерь (loss function) [12]. Функция потерь позволяет вычислить оценку несоответствия восстановленных и истинных значений. К функции потерь, как правило, предъявляются следующие требования [13] [14]: дифференцируемость и эффективность вычисления. *Дифференцируемость* необходима для применения метода градиентного спуска [15], который использует частные производные функции потерь для минимизации и обновления параметров нейросетевой модели. *Вычислительная эффективность* подразумевает быстрое вычисление функции при обучении нейросетевой модели на большом объеме данных. Кроме того, функция потерь выбирается, как правило, с учетом особенностей решаемой задачи, обучающей выборки и назначения нейросетевой модели. Например, в задачах обработки текста, как правило, используют функцию кросс-энтропии, которая позволяет нейросетевой модели оценить разницу между предсказанной и реальной вероятностями появления определенного слова на каждой позиции в предложении [16] [17]. В нейросетевой модели YOLO [18], используемой для обнаружения объектов на изображениях и видео, применяется составная функция потерь, компоненты которой учитывают расположение, класс и фон объекта.

В контексте задачи восстановления пропущенных значений во временных рядах вышеуказанные требования к функции потерь могут быть дополнены, чтобы учитывать при сравнении истинной и восстановленной подпоследовательностей ряда следующие аспекты: поведенческая схожесть, схожесть по форме и схожесть абсолютных значений. *Поведенческое сходство* предполагает, что при сравнении подпоследовательностей учитывается наличие в них сходных непрерывных промежутков меньшей длины независимо от мест взаимного расположения этих промежутков. Другими словами, промежутки в сравниваемых подпоследовательностях могут отражать сходные активности субъектов, не совпадающие по времени. *Сходство по форме* предполагает, что сравниваемые подпоследовательности имеют близкие геометрическую структуру, амплитуду и др. Наконец, *сходство по абсолютным значениям* предполагает тождественность или минимальную разницу между подпоследовательностями при сравнении их соответствующих точек.

В данной работе предлагается новая функция потерь для обучения нейросетевых моделей, предназначенных для восстановления пропущенных значений во временных рядах, получившая название MPDE (Mean Profile Distance Error, средняя ошибка профиля расстояния). MPDE позволяет учитывать поведенческое сходство, сходство по форме и абсолютным значениям при сравнении истинной и восстановленной подпоследовательностей ряда, повышая тем самым точность восстановления. Для эффективного использования предложенной функции потерь, реализован параллельный алгоритм вычисления на фреймворке PyTorch [19].

Статья организована следующим образом. В разделе [2] приводятся формальные определения понятий и нотация, используемые в статье. Предложенная функция потерь и ее параллельная реализация описаны в разделе [3]. Результаты вычислительных экспериментов, исследующих эффективность предложенного метода, приведены в разделе [4]. **Заключение** содержит сводку полученных результатов и направления будущих исследований.

1. Обзор связанных работ

Для обучения нейросетевых моделей восстановления пропущенных значений во временных рядах используются, как правило, те же функции потерь, что и при решении задачи регрессии временных рядов (предсказание значения ряда на основе последовательности его предыдущих значений) [12] [13] [20]. Одними из наиболее часто используемых являются следующие функции потерь: средняя абсолютная ошибка (Mean Absolute Error, MAE) [21], функции потерь на базе среднеквадратичной ошибки (Mean Squared Error, MSE; Root Mean Squared Error, RMSE; Mean Squared Logarithmic Error, MSLE и др.), квантильная ошибка (Quantile Loss) [22], функция потерь на основе логарифма гиперболического косинуса (Log-cosh Loss) [23], функция потерь Хьюбера (Huber Loss) [24] и др. Указанные функции ориентированы на установление соответствия абсолютных значений [12] [13] и не учитывают сходство по форме и поведенческую схожесть. Однако в последнее время исследователями предлагаются функции потерь, позволяющие учитывать форму и поведенческую схожесть подпоследовательностей временных рядов. Например, в работе [25] предложена функция потерь Soft-DTW, основанная на динамической трансформации времени (DTW, Dynamic Time Warping) [26], которая позволяет учитывать временные искажения между подпоследовательностями. В работе [27] предложена функция MPdist для вычисления расстояния между подпоследовательностями временного ряда, учитывающая поведенческую схожесть.

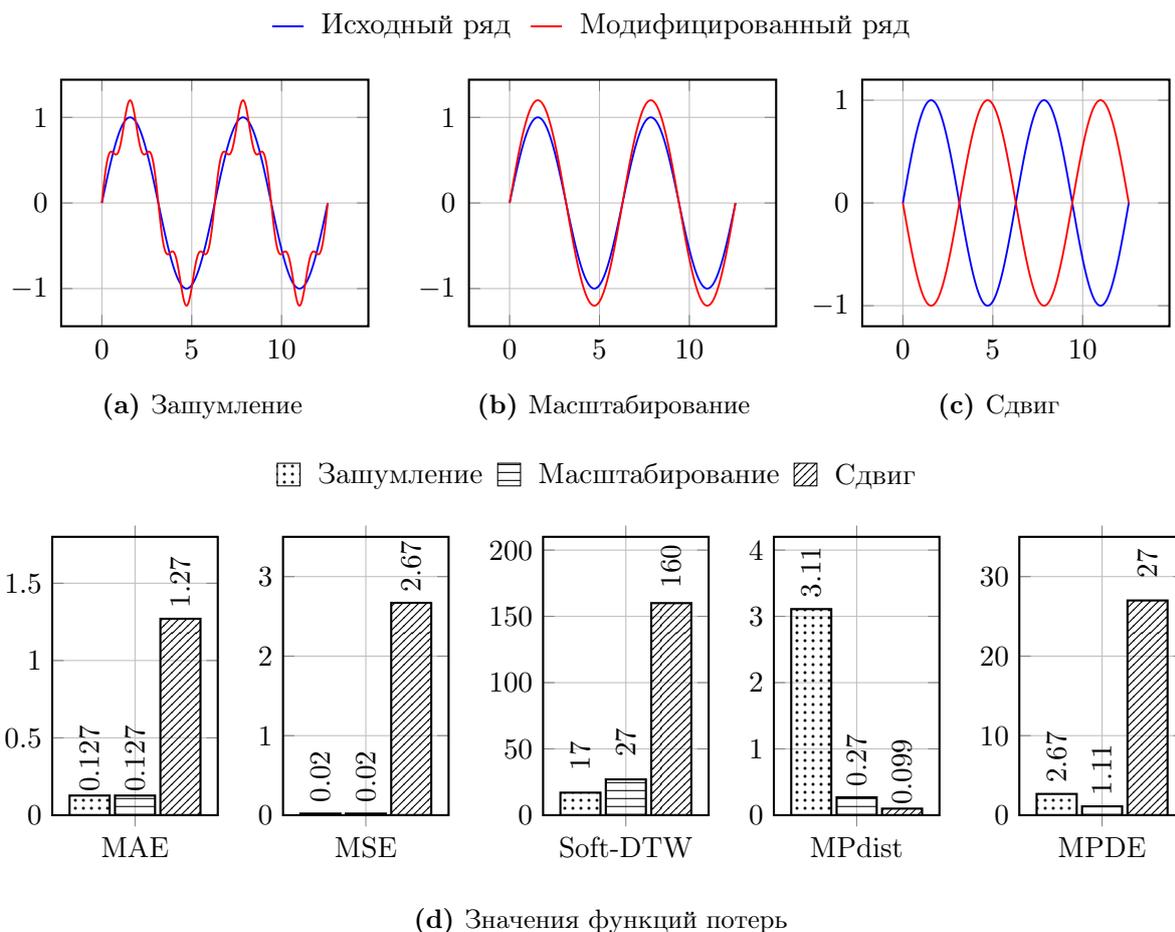


Рис. 1. Значения функций потерь, вычисленные после модификации временного ряда

Рассмотрим, как изменяется значение функций потерь (упомянутых выше и функции MPDE, предложенной в данной работе) в различных ситуациях изменения данных временного ряда (см. рис. 1). Пусть имеется временной ряд, представляющий собой значения синусоида на отрезке $[0, 15]$ взятые с шагом 0.025, к точкам которого применяются три оператора, имитирующие различные ситуации, могущие возникать в процессе обучения нейросетевых моделей восстановления временных рядов: зашумление, масштабирование и сдвиг. Оператор зашумления предполагает добавление к точкам ряда значений другой синусоиды из того же отрезка, имеющей амплитуду 0.02. Оператор масштабирования умножает значение каждой точки ряда на 1.2. Оператор сдвига изменяет фазу на π . Результат действия указанных операторов над имеющимся рядом представлен на рис. 1a, 1b и 1c соответственно. Применение оператора зашумления приводит к отсутствию всех рассмотренных выше видов схожести между исходным и результирующим рядами: поведенческое, по форме и по абсолютным значениям. Оператор масштабирования приводит к потере схожести по абсолютным значениям и форме (последняя — частично), поведенческое сходство сохраняется. Наконец, оператор сдвига сохраняет поведенческую схожесть, но нарушает сходство по абсолютным значениям и форме.

Однако, можно видеть (см. рис. 1d), что результаты вычисления функций потерь для исходного и трансформированного рядов не вполне совпадают с отмеченными выше наблюдениями. Функции потерь MSE и MAE для случаев применения операторов зашумления и масштабирования дают одинаковую ошибку. Функция Soft-DTW оценивает случай внедрения шума в данные как более близкий к истине, чем случай применения масштабирования. Функция MPdist верно оценивает масштабирование как более близкий к истине случай, чем зашумление. Однако в силу своей природы (оценка поведенческого сходства между рядами) MPdist определяет наиболее близким к истине случаем применение сдвига, что некорректно в случае задачи восстановления временного ряда. Можно заключить, что рассмотренные выше функции потерь не учитывают одновременно все рассмотренные выше аспекты схожести подпоследовательностей временного ряда, важные в задаче восстановления пропущенных значений: поведенческое сходство, сходство по форме и сходство по абсолютным значениям. Далее в данной работе предлагается функция потерь, которая нацелена на преодоление указанных ограничений.

2. Теоретический базис

Одномерный временной ряд представляет собой хронологически упорядоченную последовательность вещественных значений:

$$T = \{t_i\}_{i=1}^n, \quad t_i \in \mathbb{R}. \quad (1)$$

Длина временного ряда, n , обозначается как $|T|$.

Подпоследовательность $T_{i,m}$ одномерного временного ряда T — это непрерывный промежуток из m элементов ряда, начиная с i -го элемента:

$$T_{i,m} = \{t_k\}_{k=i}^{i+m-1}, \quad 1 \leq i \leq n - m + 1, \quad 3 \leq m \ll n. \quad (2)$$

Множество всех подпоследовательностей ряда T , имеющих длину m , обозначим как S_T^m . Рассмотрим подпоследовательность $T_{i,m}$ как самостоятельный временной ряд. Тогда его подпоследовательность, имеющую заданную длину ℓ (где $\ell < m$), будем называть *ок-*

ном. Мощность множества окон заданной подпоследовательности обозначим соответственно как c , т.е. $c = |S_{T_i, m}^\ell| = m - \ell + 1$.

Многомерный временной ряд — это набор семантически связанных одномерных временных рядов одинаковой длины, которые синхронизированы во времени. Пусть d обозначает размерность многомерного ряда ($d > 1$), количество измерений — одномерных рядов в нем. Подобно одномерному случаю, многомерный временной ряд, его подпоследовательность и отдельные точки обозначим как \mathbf{T} , $\mathbf{T}_{i, m}$ и \mathbf{t}_i соответственно, и определим их следующим образом:

$$\mathbf{T} = [\{T^{(k)}\}_{k=1}^d]^\top, \quad (3)$$

$$\mathbf{T}_{i, m} = [\{T_{i, m}^{(k)}\}_{k=1}^d]^\top, \quad (4)$$

$$\mathbf{t}_i = [\{t_i^{(k)}\}_{k=1}^d]^\top. \quad (5)$$

Для вычисления расстояний между окнами подпоследовательности в данной работе используются евклидово расстояние и его z-нормализованная версия, обозначаемые как $\text{ED}(\cdot, \cdot)$ и $\widehat{\text{ED}}(\cdot, \cdot)$ соответственно, и вычисляемые следующим образом:

$$\text{ED}(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}. \quad (6)$$

$$\widehat{\text{ED}}(X, Y) = \text{ED}(\widehat{X}, \widehat{Y}), \quad (7)$$

$$\widehat{x}_i = \frac{x_i - \mu_x}{\sigma_x}, \quad \mu_x = \frac{1}{\ell} \sum_{i=1}^{\ell} x_i, \quad \sigma_x = \sqrt{\frac{1}{\ell - 1} \sum_{i=1}^{\ell} (x_i - \mu_x)^2 + \varepsilon}, \quad (8)$$

где ε представляет собой машинный эpsilon, предотвращающий деление на ноль в вычислениях градиентов [28].

3. Функция потерь на основе среднего расстояния профилей

3.1. Формальное определение функции потерь

Предлагаемая функция потерь основана на средней ошибке профиля расстояния (MPDE, Mean Profile Distance Error) и вычисляется между двумя подпоследовательностями многомерного ряда \mathbf{T} , имеющими длину m : $\mathbf{X}, \mathbf{Y} \in S_{\mathbf{T}}^m$, где \mathbf{X} играет роль восстановленных, а \mathbf{Y} — реальных данных. Далее рассмотрим формальное определение функции MPDE для одномерного ряда (т.е. для случая $X, Y \in S_T^m$), а затем обобщим это определение на случай многомерного ряда.

MPDE вычисляется как среднее значение расстояний между всеми парами окон этих подпоследовательностей, имеющими длину ℓ , где $\ell \leq m$ и окна имеют одинаковые индексы. Расстояние между окнами вычисляется как взвешенная сумма двух слагаемых: евклидова расстояния между z-нормализованными версиями окон и евклидова расстояния между исходными данными. Веса указанных слагаемых, α и β соответственно, представляют собой вещественные числа из отрезка $[0, 1]$, и являются параметрами функции потерь. Формально MPDE может быть представлено следующим образом:

$$\text{MPDE}(X, Y) = \frac{1}{c} \sum_{i=1}^c \alpha \text{ED}(X_{i, i+\ell}, Y_{i, i+\ell}) + \beta \widehat{\text{ED}}(X_{i, i+\ell}, Y_{i, i+\ell}), \quad (9)$$

$$0 \leq \alpha, \beta \leq 1.$$

Рассмотрим дифференцируемость функции MPDE. Частная производная MPDE по переменной x_i вычисляется как среднее значение производных расстояний между окнами, содержащими точку x_i :

$$\frac{\partial \text{MPDE}(X, Y)}{\partial x_i} = \frac{1}{c} \sum_{j=i-\frac{\ell}{2}}^{i+\frac{\ell}{2}} \frac{\partial \text{ED}(X_{j,j+\ell}, Y_{j,j+\ell})}{\partial x_i} + \frac{\partial \widehat{\text{ED}}(X_{j,j+\ell}, Y_{j,j+\ell})}{\partial x_i}, \quad (10)$$

Учитывая формул [6](#), [7](#) и [8](#) частная производная функции потерь по переменной x_i может быть записана в следующем виде:

$$\begin{aligned} \frac{\partial \text{MPDE}(X, Y)}{\partial x_i} &= A + B + C, \\ A &= \frac{\alpha}{c} \cdot \sum_{j=i-\frac{\ell}{2}}^{i+\frac{\ell}{2}} \frac{x_i - y_i}{\text{ED}(X_{j,j+\ell}, Y_{j,j+\ell})}, \quad B = \frac{\beta}{c} \cdot \sum_{j=i-\frac{\ell}{2}}^{i+\frac{\ell}{2}} \frac{\widehat{x}_i - \widehat{y}_i}{\text{ED}(X_{j,j+\ell}, Y_{j,j+\ell})}, \\ C &= \frac{\beta}{\ell \cdot c} \cdot \sum_{j=i-\frac{\ell}{2}}^{i+\frac{\ell}{2}} \frac{\partial \text{ED}(X_{j,j+\ell}, Y_{j,j+\ell})}{\partial \mu} \cdot \frac{\partial \mu}{\partial x_i} + \frac{\beta}{\sigma \cdot c} \cdot \sum_{j=i-\frac{\ell}{2}}^{i+\frac{\ell}{2}} \frac{\partial \text{ED}(X_{j,j+\ell}, Y_{j,j+\ell})}{\partial \sigma} \cdot \frac{\partial \sigma}{\partial x_i}. \end{aligned} \quad (11)$$

Рассмотрим слагаемые A , B и C в данной формуле более подробно. Слагаемое A включает в себя сумму расстояний между точками окон, что позволяет во время обучения нейросетевой модели восстановления ряда компенсировать различия в масштабах восстановленной и истинной подпоследовательностей. Слагаемое B задействует разность между нормализованными точками окон. Это дает возможность при обучении модели учесть сходство форм восстановленной и истинной подпоследовательностей. Наконец, слагаемое C включает в себя производные параметров нормализации μ и σ по переменной x_i и позволяет учитывать, как изменение одной точки влияет на параметры нормализации для всего окна. Изменения весов α и β позволяет регулировать вклады слагаемых A и B , C соответственно в итоговое значение функции потерь.

В случае многомерного ряда значение функции потерь от аргументов $\mathbf{X}, \mathbf{Y} \in S_T^m$ вычисляется как среднее арифметическое значений функции MPDE по всем измерениям между соответствующими одномерными подпоследовательностями $X^{(i)}, Y^{(i)} \in S_{T^{(i)}}^m$:

$$\text{MPDE}(\mathbf{X}, \mathbf{Y}) = \frac{1}{d} \sum_{i=1}^d \text{MPDE}(X^{(i)}, Y^{(i)}). \quad (12)$$

Очевидно, однако, что функция MPDE имеет существенно более высокую вычислительную сложность, чем известные аналоги, поскольку в ней необходимо вычисление евклидовых расстояний между всеми исходными и z -нормализованными окнами входных подпоследовательностей. Для решения данной проблемы и применения функции MPDE на практике для обучения нейросетевых моделей восстановления временных рядов в данной работе предлагается параллельный алгоритм вычисления данной функции потерь, который рассмотрен в следующем разделе.

3.2. Алгоритм вычисления функции потерь

Для параллельной реализации вычисления функции потерь, описанной в разделе [3.1](#) используется фреймворк PyTorch [19](#), который в настоящее время является де-факто стандартом реализации нейросетевых моделей. Основной структурой для хранения данных в

PyTorch является тензор (Tensor), представляющий собой многомерную матрицу вещественных значений. Фреймворк предоставляет разработчику набор базовых методов для работы с тензорами: вычисление среднего значения, квадратного корня, стандартного отклонения и др. Фреймворк также поддерживает автоматическое дифференцирование базовых методов и их параллельное исполнение на графических процессорах [29]. Для реализации используются базовые методы, выполняющие два вида операций: агрегация и извлечение данных. Методы агрегации выполняют над входным тензором одну из следующих операций: вычисление среднего `mean`, суммы `sum`, квадратного корня `sqrt` и дисперсии `var`. Агрегация может выполняться по всем измерениям либо вдоль заданного измерения, превращая размерность этого измерения в единицу. Извлечение данных реализуется с помощью метода `unfold`, который для заданного тензора возвращает все его подмассивы заданной длины, взятые вдоль заданного измерения.

Алг. 1 TorchMPDE(IN: $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{b \times m \times d}$, α, β ; OUT: MPDE)

```

1:  $\mathbf{X}_{\text{slice}} \leftarrow \text{unfold}(\mathbf{X}, \ell, 1)$ ;  $\mathbf{Y}_{\text{slice}} \leftarrow \text{unfold}(\mathbf{Y}, \ell, 1)$ 
2:  $\mu_x \leftarrow \text{mean}(\mathbf{X}_{\text{slice}}, 2)$ ;  $\mu_y \leftarrow \text{mean}(\mathbf{Y}_{\text{slice}}, 2)$ 
3:  $\sigma_x \leftarrow \text{sqrt}(\text{var}(\mathbf{X}_{\text{slice}}, 2) + \varepsilon)$ ;  $\sigma_y \leftarrow \text{sqrt}(\text{mean}(\mathbf{Y}_{\text{slice}}, 2) + \varepsilon)$ 
4:  $\mathbf{X}_{\text{norm}} \leftarrow (\mathbf{X}_{\text{slice}} - \mu_x) / \sigma_x$ ;  $\mathbf{Y}_{\text{norm}} \leftarrow (\mathbf{Y}_{\text{slice}} - \mu_y) / \sigma_y$ 
5:  $\text{ED2}_{\text{norm}} \leftarrow \text{sum}((\mathbf{X}_{\text{norm}} - \mathbf{Y}_{\text{norm}})^2, 3)$ ;  $\text{ED2} \leftarrow \text{sum}((\mathbf{X}_{\text{slice}} - \mathbf{Y}_{\text{slice}})^2, 3)$ 
6:  $\text{ED}_{\text{norm}} \leftarrow \text{sqrt}(\text{ED2}_{\text{norm}} + \varepsilon)$ ;  $\text{ED} \leftarrow \text{sqrt}(\text{ED2} + \varepsilon)$ 
7:  $\text{MPDE} \leftarrow \text{mean}(\alpha \cdot \text{ED} + \beta \cdot \text{ED}_{\text{norm}}) / 2$ 
8: return MPDE

```

Реализация вычисления функции потерь MPDE представлена в алг. 1. Для встраивания реализации во фреймворк глубокого обучения функция MPDE должна оперировать пакетами (batch), которые состоят из пар с восстановленным и истинным значением. Поэтому, помимо параметров α и β , входными данными алгоритма являются два массива длины b , в которых хранятся восстановленные и соответствующие им истинные многомерные подпоследовательности длины m , обозначенные как трехмерные тензоры $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{b \times m \times d}$ соответственно. На первом шаге алгоритма над пакетом подпоследовательностей выполняется операция `unfold` вдоль измерения времени. В результате каждая подпоследовательность разбивается на окна длины ℓ . Множества всех окон каждой подпоследовательности входных пакетов сохраняются в 4-мерные тензоры $\mathbf{X}_{\text{slice}}, \mathbf{Y}_{\text{slice}} \in \mathbb{R}^{b \times m - \ell \times \ell \times d}$. Далее для каждого окна с помощью операций `mean`, `var` и `sqrt` вычисляются промежуточные данные z-нормализации: средние значения μ_x, μ_y и среднеквадратичные отклонения σ_x, σ_y . Затем каждый тензор, хранящий окна подпоследовательностей, нормализуется с учетом данных, вычисленных на предыдущем шаге. После этого происходит вычисление евклидовых расстояний между двумя типами окон: нормализованными и исходными (см. строки 5 и 6 соответственно). На последнем шаге полученные расстояния умножаются на весовые коэффициенты α и β , а результаты усредняются по пакету. Полученный результат является итоговым значением функции потерь.

4. Вычислительные эксперименты

Для исследования эффективности предложенного метода были проведены вычислительные эксперименты на оборудовании Лаборатории суперкомпьютерного моделирования

Южно-Уральского государственного университета (Челябинск) [30]. В экспериментах сравнивалась точность восстановления пропусков, которую показывают передовые нейросетевые модели, обученные с помощью предложенной функции MPDE и другими общепринятыми функциями потерь. Сравнение осуществлялось с использованием наборов данных из различных предметных областей. Кроме того, в экспериментах исследовалось быстродействие вычисления указанных функций потерь.

4.1. Описание экспериментов

В экспериментах выполнялось сравнение предложенной функции потерь со следующими функциями потерь: средняя абсолютная ошибка (Mean Absolute Error, MAE), среднеквадратичная ошибка (Mean Squared Error, MSE) и квантильная функция потерь (Quantile Loss) [22].

Таблица 1. Наборы данных, используемые в экспериментах

№	Набор	Длина, $n \times 10^3$	Количество, измерений, d	Предметная область
<i>Группа А: Активность субъекта</i>				
1.	Electricity [31]	5	9	Потребление электроэнергии в нескольких домашних хозяйствах
2.	Madrid [32]	25	10	Трафик автомобильных дорог в Мадриде
3.	NREL [33]	8.7	9	Потребление электроэнергии лаборатории в США
4.	PAMAP [34]	50	10	Показания носимого датчика во время активности человека
5.	WalkRun [35]	37	11	
<i>Группа Б: Сезонность и циклы</i>				
6.	BAFU [36]	50	10	Сброс воды в реках Швейцарии
7.	Climate [37]	5	10	Погода в различных локациях Северной Америки
8.	MeteoSwiss [38]	10	10	Погода в городах Швейцарии
9.	Saaleaue [39]	23	14	Погода в городах Германии

Оценка предложенного метода и его сравнения с аналогами проводилась с использованием временных рядов, резюмированных в табл. 1. Временные ряды могут быть разделены на две группы в соответствии с особенностями данных: в группу А входят ряды 1–5, для которых характерно отсутствие сезонности и цикличности компонентов ввиду возможности случайного изменения активности субъекта; группа Б включает в себя ряды 6–9, которые демонстрируют сезонность и цикл (циклические изменения уровня ряда с постоянным и переменным периодом соответственно). Пропуски в многомерном временном ряде формировались в соответствии с наиболее тяжелым для восстановления сценарием Blackout [40], который предполагает пустые значения в каждой из 100 последних точек в каждом измерении многомерного временного ряда.

Для оценки точности восстановления в данной работе используется мера корня из среднеквадратичной ошибки RMSE (Root Mean Square Error) как одна из наиболее часто применяемых для этих целей метрик [41], определяемая следующим образом:

$$RMSE = \sqrt{\frac{1}{h} \sum_{i=0}^n (y_i - \hat{y}_i)^2}, \quad (13)$$

где y_i — фактическое значение, \hat{y}_i — восстановленное значение, h — количество восстановленных точек.

Для оценки предложенной в работе функции потерь использовались методы, основанные на различных архитектурах: рекуррентные (BRITS [7] и SANNI [42]), автоэнкодеры (SAETI [35]) и трансформеры (SAITS [10]). В табл. [2] резюмированы параметры обучения нейросетевых моделей, используемых для оценки предложенного метода. Оптимальные параметры функции потерь MPDE для каждой комбинации $\langle \text{Модель}, \text{Набор данных} \rangle$ были подобраны с помощью решетчатого поиска [43] в диапазоне $[0.1, 0.9]$ с шагом 0.1.

Таблица 2. Параметры обучения, используемые в экспериментах

№	Название	Значение
1.	Количество эпох (Epochs)	100
2.	Скорость обучения (Learning rate)	5×10^{-3}
3.	Оптимизатор (Optimizer)	Adam
4.	Размер пакета (Batch size)	256
5.	Длина подпоследовательности (m)	100
6.	Длина окна (ℓ)	50

4.2. Анализ результатов

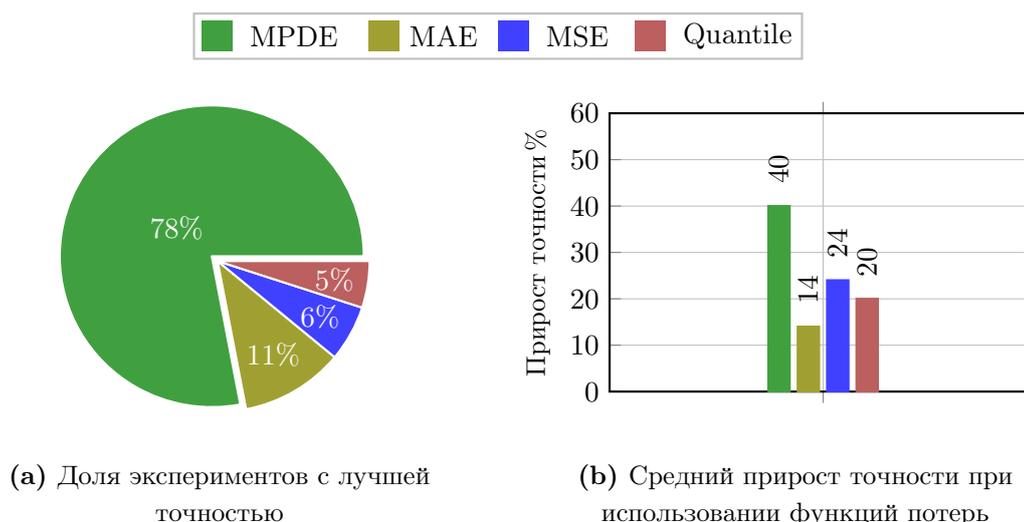


Рис. 2. Точность моделей при использовании различных функций потерь

На рис. [2] представлена доля пар $\langle \text{Модель}, \text{Набор данных} \rangle$ в которых каждая функция потерь продемонстрировала лучший результат. На рис. [2b] показан средний выигрыш (среднее повышение точности в процентах) лучшей функции потерь у прочих функций. Подробные результаты точности восстановления, полученной каждой функцией потерь на каждом наборе данных, разбитые для удобства просмотра по группам данных (А — активность субъекта и Б — сезонность и циклы), представлены на рис. [4] и [5] соответственно.

Можно видеть, что предложенная функция потерь MPDE позволяет минимум в 78% случаев повысить точность восстановления минимум на 40% по сравнению с рассмотренными передовыми функциями потерь. В случае, когда применение MPDE не дает наивысшую точность восстановления, ее отставание от лучшей функции потерь составляет не более 6%.

Отдельно по группам наборов данных MPDE в среднем опережает аналоги по точности на 34% и 46% и является лучшей в 80% и 75% случаев в группах А и Б соответственно.

Таблица 3. Оптимальные параметры MPDE

№	Название	Модель							
		BRITS		SANNI		SAETI		SAITS	
		α	β	α	β	α	β	α	β
<i>Группа А: Активность субъекта</i>									
1.	Electricity	0.7	0.2	0.6	0.1	0.1	0.8	0.5	0.5
2.	Madrid	0.2	0.7	0.5	0.3	0.2	0.5	0.8	0.5
3.	NREL	0.6	0.1	0.5	0.3	0.8	0.2	0.2	0.5
4.	PAMAP	0.4	0.2	0.4	0.1	0.5	0.6	0.4	0.1
5.	WalkRun	0.8	0.4	0.1	0.2	0.3	0.3	0.8	0.2
<i>Группа Б: Сезонность и циклы</i>									
6.	BAFU	0.9	0.1	0.1	0.1	0.2	0.7	0.7	0.2
7.	Climate	0.4	0.9	0.4	0.3	0.5	0.3	0.3	0.8
8.	MeteoSwiss	0.3	0.1	0.2	0.6	0.7	0.8	0.0	0.8
9.	Saaleaue	0.1	0.1	0.6	0.3	0.9	0.4	0.1	0.7

В табл. 3 приведены значения параметров α и β функции MPDE, при которых нейросетевая модель достигает наиболее высокой точности восстановления. Средние значения весовых коэффициентов α и β — по всем наборам данных (0.47 и 0.34) или отдельно для групп данных А (0.4 и 0.45) и Б (0.44 и 0.39) — могут быть использованы как начальные значения для настройки этих гиперпараметров — в случае, когда характеристики восстанавливаемого ряда неизвестны или ряд можно причислить к одной из указанных выше групп соответственно. Автоматизация подбора гиперпараметров α и β может рассматриваться как одно из направлений будущих исследований.

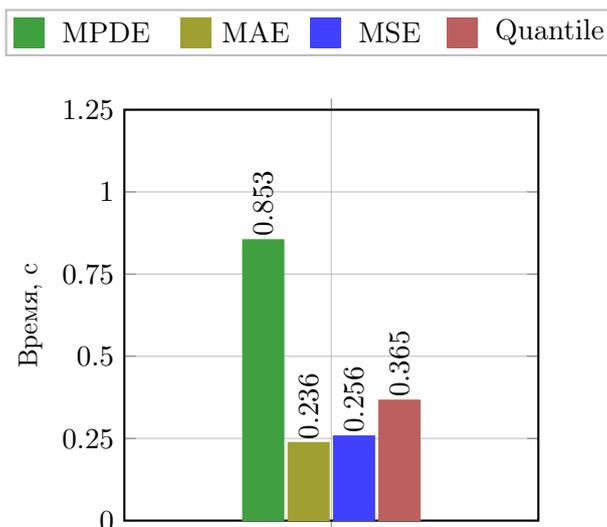


Рис. 3. Быстродействие вычисления функции потерь

На рис. 3 представлено сравнение быстродействия вычисления рассмотренных в описанных выше экспериментах функций потерь. В данном эксперименте сравнивалось время, за которое вычислялись функции потерь для пакета из 128 многомерных подпоследовательностей длиной $m = 100$ (для функции MPDE длина окна $\ell = 50$) и количеством измерений $d = 10$, усредненное по 10 тыс. запусков. Можно видеть, что вычисление MPDE происходит в среднем в 4 раза медленнее, чем у известных функций потерь. Однако это необходимая плата за более высокую точность восстановления, которую обеспечивает функция MPDE, по сравнению с известными функциями потерь.

Заключение

В данной статье затронута проблема повышения точности нейросетевых моделей восстановления пропущенных значений в многомерных временных рядах, которая является актуальной в широком спектре предметных областей. Предложена новая функция потерь для обучения нейросетевых моделей восстановления временных рядов, которая получила название MPDE (Mean Profile Distance Error) и основана на средней ошибке профиля расстояния. MPDE для истинной и восстановленной подпоследовательностей ряда вычисляется как среднее значение расстояний между всеми парами окон (непрерывных промежутков) этих подпоследовательностей, имеющими меньшую длину, где окна имеют одинаковые начальные индексы. Расстояние между двумя окнами определяется как взвешенная сумма евклидова и z-нормированного евклидова расстояний между данными окнами. Веса слагаемых берутся из отрезка $[0,1]$ и являются гиперпараметрами функции потерь, равно как и длины подпоследовательности и окна. При обучении нейросетевой модели и многократном сравнении истинных и восстановленных подпоследовательностей ряда MPDE учитывает поведенческое сходство, сходство по форме и абсолютным значениям, повышая тем самым точность восстановления. Поведенческое сходство предполагает, что при сравнении подпоследовательностей учитывается наличие в них сходных окон независимо от мест взаимного расположения этих окон. Другими словами, MPDE учитывает, что в сравниваемых подпоследовательностях окна могут отражать сходные активности субъекта, не совпадающие по времени.

Функция MPDE имеет существенно более высокую вычислительную сложность, чем известные аналоги, поскольку в ней необходимо вычисление евклидовых расстояний между всеми исходными и z-нормализованными окнами входных подпоследовательностей. Для эффективного использования предложенной функции потерь в обучении нейросетевых моделей реализован параллельный алгоритм вычисления MPDE, работа которого кратко может быть описана следующим образом. Сначала каждый входной пакет подпоследовательностей разбивается на окна. Далее для каждого окна вычисляются параметры z-нормализации (среднее и стандартное отклонение), после чего выполняется нормализация всех окон. Затем алгоритм вычисляет евклидовы расстояния между нормализованными и исходными окнами. Наконец, полученные расстояния с учетом весов усредняются по пакету. Алгоритм реализован на базе фреймворка PyTorch [19], который инкапсулирует параллелизм вычислений на графическом процессоре. В вычислительных экспериментах сравнивалась точность восстановления, которую демонстрируют передовые нейросетевые модели на многомерных временных рядах из различных предметных областей, при применении MPDE и других известных функций потерь. Результаты экспериментов показывают, что в 78% случаев нейросетевые модели достигают наиболее высокой точности восстановления (по метрике

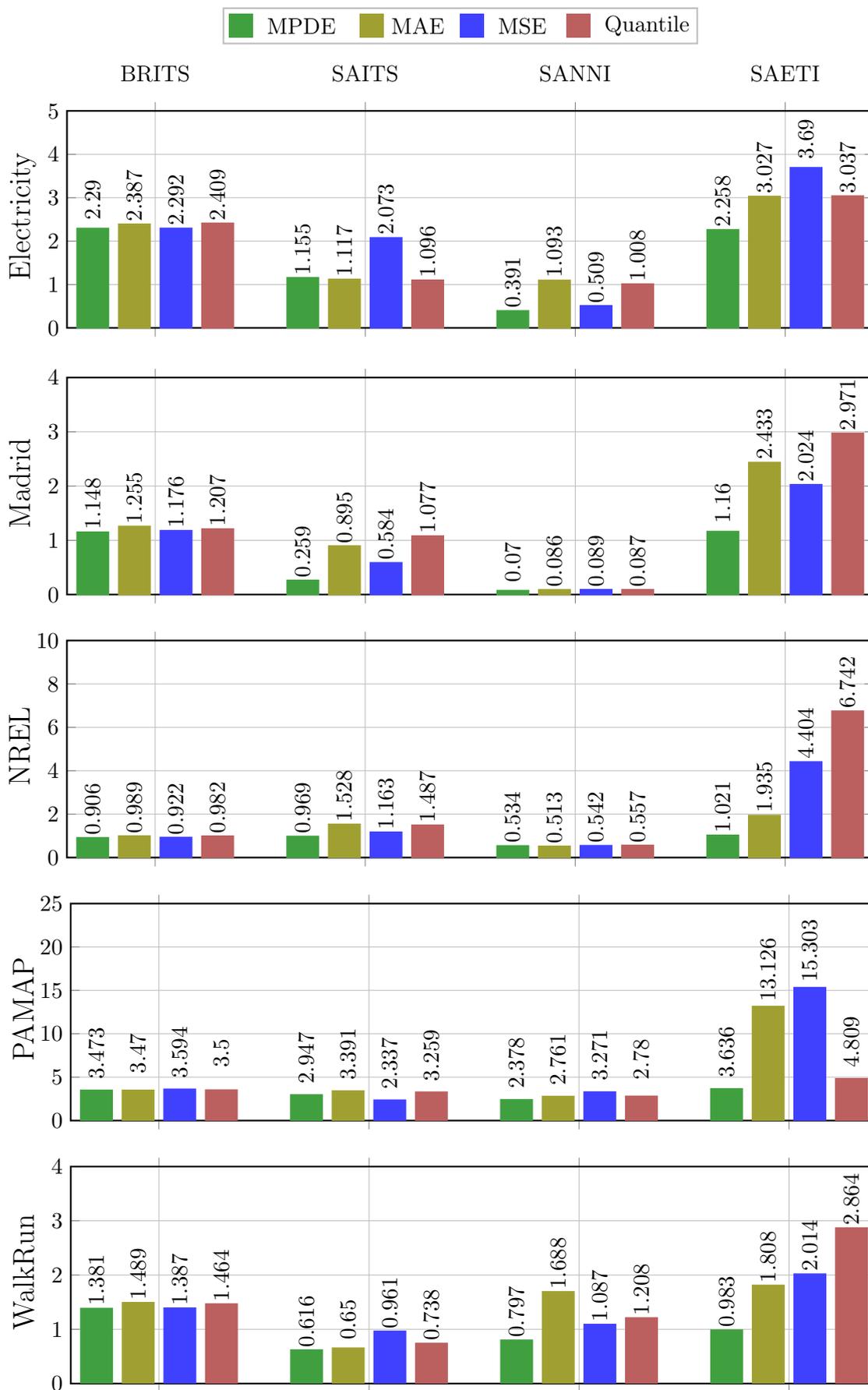


Рис. 4. Результаты экспериментов для данных группы А (RMSE)

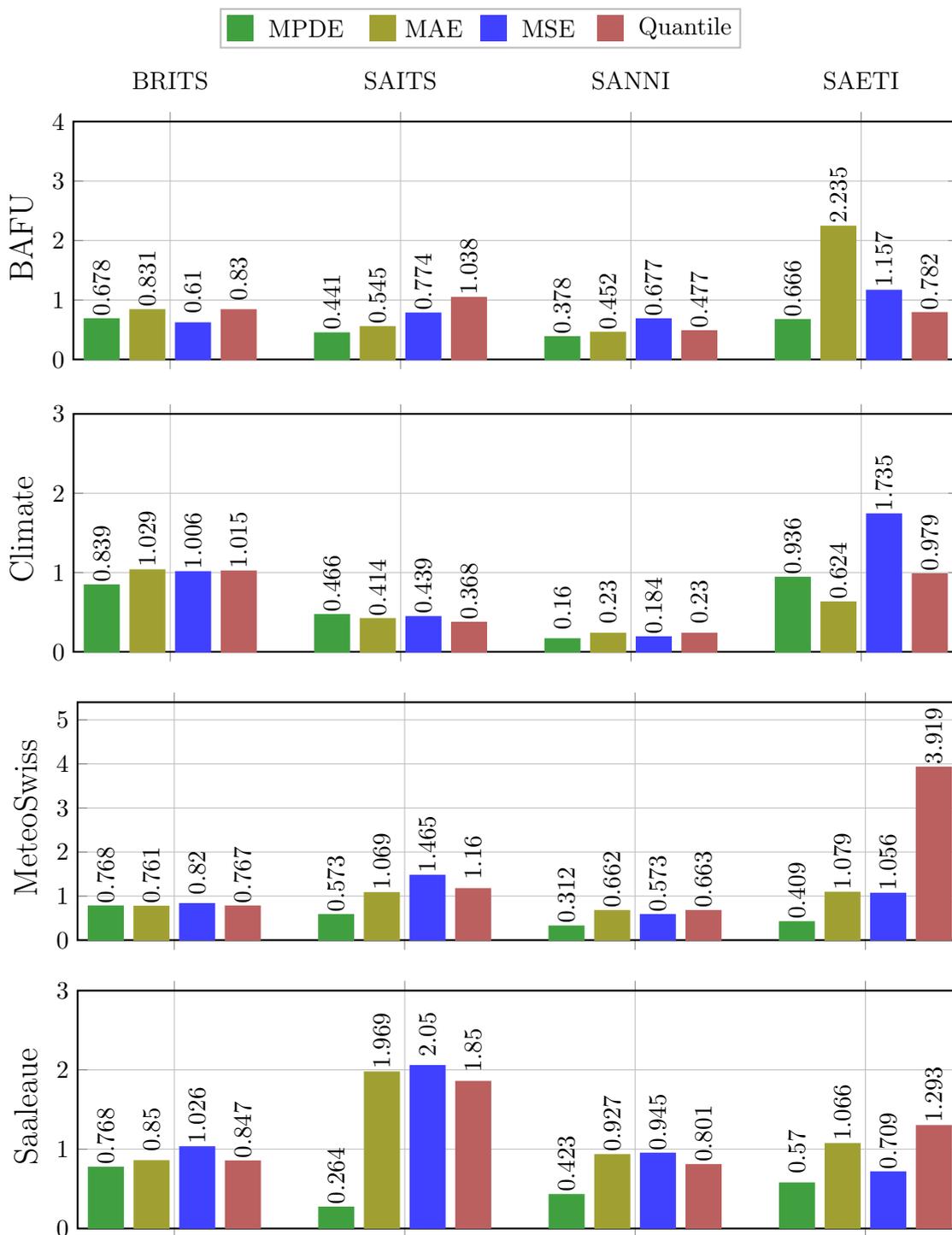


Рис. 5. Результаты экспериментов для данных группы Б (RMSE)

RMSE) при использовании предложенной функции потерь; при этом точность восстановления на 40% выше среднего значения, достигнутого при использовании других функций потерь. Однако, необходимой платой за более высокую точность восстановления является более низкое быстродействие вычисления MPDE по сравнению с известными функциями потерь (в среднем в 4 раза медленнее).

В будущих исследованиях планируется рассмотреть возможность автоматизации процесса подбора гиперпараметров функции потерь.

Работа выполнена при финансовой поддержке Российского научного фонда (грант № 23-21-00465).

Литература

1. Majumdar S., Laha A.K. Clustering and classification of time series using topological data analysis with applications to finance // *Expert Syst. Appl.* 2020. Vol. 162. P. 113868. DOI: [10.1016/j.eswa.2020.113868](https://doi.org/10.1016/j.eswa.2020.113868).
2. Lara-Benitez P., Carranza-Garcia M., Luna-Romera J.M., Riquelme J.C. Temporal convolutional networks applied to energy-related time series forecasting // *applied sciences*. 2020. Vol. 10, no. 7. P. 2322. DOI: [10.3390/app10072322](https://doi.org/10.3390/app10072322).
3. Gratius N., Wang Z., Hwang M.Y., *et al.* Digital Twin Technologies for Autonomous Environmental Control and Life Support Systems // *J. Aerosp. Inf. Syst.* 2024. Vol. 21, no. 4. P. 332–347. DOI: [10.2514/1.I011320](https://doi.org/10.2514/1.I011320).
4. Zhou Z., Tang W., Li M., *et al.* A Novel Hybrid Intelligent SOPDEL Model with Comprehensive Data Preprocessing for Long-Time-Series Climate Prediction // *Remote. Sens.* 2023. Vol. 15, no. 7. P. 1951. DOI: [10.3390/RS15071951](https://doi.org/10.3390/RS15071951).
5. Kazijevs M., Samad M.D. Deep imputation of missing values in time series health data: A review with benchmarking // *J. Biomed. Informatics*. 2023. Vol. 144. P. 104440. DOI: [10.1016/J.JBI.2023.104440](https://doi.org/10.1016/J.JBI.2023.104440).
6. Fang C., Wang C. Time Series Data Imputation: A Survey on Deep Learning Approaches // *CoRR*. 2020. Vol. abs/2011.11347. arXiv: [2011.11347](https://arxiv.org/abs/2011.11347). URL: <https://arxiv.org/abs/2011.11347>.
7. Cao W., Wang D., Li J., *et al.* BRITS: Bidirectional Recurrent Imputation for Time Series // *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada* / ed. by S. Bengio, H.M. Wallach, H. Larochelle, *et al.* 2018. P. 6776–6786. URL: <https://proceedings.neurips.cc/paper/2018/hash/734e6bfcd358e25ac1db0a4241b95651-Abstract.html>.
8. Yoon J., Zame W.R., Schaar M. van der Estimating Missing Data in Temporal Data Streams Using Multi-Directional Recurrent Neural Networks // *IEEE Trans. Biomed. Eng.* 2019. Vol. 66, no. 5. P. 1477–1490. DOI: [10.1109/TBME.2018.2874712](https://doi.org/10.1109/TBME.2018.2874712).
9. Fortuin V., Baranchuk D., Rätsch G., Mandt S. GP-VAE: Deep Probabilistic Time Series Imputation // *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*. Vol. 108 / ed. by S. Chappa, R. Calandra. PMLR, 2020. P. 1651–1661. *Proceedings of Machine Learning Research*. URL: <http://proceedings.mlr.press/v108/fortuin20a.html>.
10. Du W., Côté D., Liu Y. SAITS: Self-attention-based imputation for time series // *Expert Syst. Appl.* 2023. Vol. 219. P. 119619. DOI: [10.1016/J.ESWA.2023.119619](https://doi.org/10.1016/J.ESWA.2023.119619).
11. Oh E., Kim T., Ji Y., Khyalia S. STING: Self-attention based Time-series Imputation Networks using GAN. 2021. DOI: [10.1109/ICDM51629.2021.00155](https://doi.org/10.1109/ICDM51629.2021.00155).
12. Kaya M., Bilge H.S. Deep Metric Learning: A Survey // *Symmetry*. 2019. Vol. 11, no. 9. P. 1066. DOI: [10.3390/SYM11091066](https://doi.org/10.3390/SYM11091066).

13. Wang Q., Ma Y., Zhao K., Tian Y. A comprehensive survey of loss functions in machine learning // *Annals of Data Science*. 2022. Vol. 9. P. 187–212. DOI: [10.1007/s40745-020-00253-5](https://doi.org/10.1007/s40745-020-00253-5).
14. Ciampiconi L., Elwood A., Leonardi M., *et al.* A survey and taxonomy of loss functions in machine learning // *CoRR*. 2023. Vol. abs/2301.05579. DOI: [10.48550/ARXIV.2301.05579](https://doi.org/10.48550/ARXIV.2301.05579), arXiv: [2301.05579](https://arxiv.org/abs/2301.05579).
15. Netrapalli P. Stochastic gradient descent and its variants in machine learning // *Journal of the Indian Institute of Science*. 2019. Vol. 99, no. 2. P. 201–213. DOI: [10.1007/s41745-019-0098-4](https://doi.org/10.1007/s41745-019-0098-4).
16. Xu J., Ren X., Lin J., Sun X. Diversity-Promoting GAN: A Cross-Entropy Based Generative Adversarial Network for Diversified Text Generation // *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018* / ed. by E. Riloff, D. Chiang, J. Hockenmaier, J. Tsujii. Association for Computational Linguistics, 2018. P. 3940–3949. DOI: [10.18653/v1/D18-1428](https://doi.org/10.18653/v1/D18-1428).
17. Chang Y., Wang X., Wang J., *et al.* A Survey on Evaluation of Large Language Models // *ACM Trans. Intell. Syst. Technol.* 2024. Vol. 15, no. 39. P. 1–45. DOI: [10.1145/3641289](https://doi.org/10.1145/3641289).
18. Chen W., Huang H., Peng S., *et al.* YOLO-face: a real-time face detector // *Vis. Comput.* 2021. Vol. 37, no. 4. P. 805–813. DOI: [10.1007/S00371-020-01831-7](https://doi.org/10.1007/S00371-020-01831-7).
19. Paszke A., Gross S., Massa F., *et al.* PyTorch: An Imperative Style, High-Performance Deep Learning Library // *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* / ed. by H.M. Wallach, H. Larochelle, A. Beygelzimer, *et al.* 2019. P. 8024–8035. URL: <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>.
20. Jadon A., Patil A., Jadon S. A Comprehensive Survey of Regression-Based Loss Functions for Time Series Forecasting // *International Conference on Data Management, Analytics & Innovation*. Springer. 2024. P. 117–147. DOI: [10.1007/978-981-97-3245-6_9](https://doi.org/10.1007/978-981-97-3245-6_9).
21. Qi J., Du J., Siniscalchi S.M., *et al.* On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression // *IEEE Signal Process. Lett.* 2020. Vol. 27. P. 1485–1489. DOI: [10.1109/LSP.2020.3016837](https://doi.org/10.1109/LSP.2020.3016837).
22. Chen X., Liu W., Mao X., Yang Z. Distributed High-dimensional Regression Under a Quantile Loss Function // *J. Mach. Learn. Res.* 2020. Vol. 21, no. 182. P. 1–43. URL: <http://jmlr.org/papers/v21/20-297.html>.
23. Saleh R.A., Saleh A.K.M.E. Statistical Properties of the log-cosh Loss Function Used in Machine Learning // *CoRR*. 2022. Vol. abs/2208.04564. DOI: [10.48550/ARXIV.2208.04564](https://doi.org/10.48550/ARXIV.2208.04564), arXiv: [2208.04564](https://arxiv.org/abs/2208.04564).
24. Sun Q., Zhou W.-X., Fan J. Adaptive Huber regression // *Journal of the American Statistical Association*. 2020. Vol. 115, no. 529. P. 254–265. DOI: [10.1080/01621459.2018.1543124](https://doi.org/10.1080/01621459.2018.1543124).
25. Cuturi M., Blondel M. Soft-DTW: a Differentiable Loss Function for Time-Series // *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70 / ed. by D. Precup, Y.W. Teh. PMLR, June 2017. P. 894–903. *Proceedings of Machine Learning Research*. URL: <https://proceedings.mlr.press/v70/cuturi17a.html>.

26. Berndt D.J., Clifford J. Using Dynamic Time Warping to find patterns in time series // KDD Workshop. 1994. P. 359–370. URL: <https://cdn.aaai.org/Workshops/1994/WS-94-03/WS94-03-031.pdf>.
27. Gharghabi S., Imani S., Bagnall A.J., *et al.* Matrix Profile XII: MPdist: A Novel Time Series Distance Measure to Allow Data Mining in More Challenging Scenarios // IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018. IEEE Computer Society, 2018. P. 965–970. DOI: [10.1109/ICDM.2018.00119](https://doi.org/10.1109/ICDM.2018.00119).
28. Zhuang J., Tang T., Ding Y., *et al.* AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients // Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual / ed. by H. Larochelle, M. Ranzato, R. Hadsell, *et al.* 2020. URL: <https://proceedings.neurips.cc/paper/2020/hash/d9d4f495e875a2e075a1a4a6e1b9770f-Abstract.html>.
29. Baydin A.G., Pearlmutter B.A., Radul A.A., Siskind J.M. Automatic Differentiation in Machine Learning: a Survey // J. Mach. Learn. Res. 2017. Vol. 18, no. 153. P. 1–43. URL: <https://jmlr.org/papers/v18/17-468.html>.
30. Биленко Р.В., Долганина Н.Ю., Иванова Е.В., Рекачинский А.И. Высокопроизводительные вычислительные ресурсы Южно-Уральского государственного университет // Вычислительные методы и программирование. 2022. Т. 11, № 1. С. 15–30. DOI: [10.14529/cmse220102](https://doi.org/10.14529/cmse220102).
31. Trindade A. Electricity Load Diagrams 2011–2014. 2015. DOI: [10.24432/C58C86](https://doi.org/10.24432/C58C86). UCI Machine Learning Repository.
32. Laña I., Olabarrieta I., Vélez M., Del Ser J. On the imputation of missing data for road traffic forecasting: New insights and novel techniques // Transportation Research Part C: Emerging Technologies. 2018. Vol. 90. P. 18–33. DOI: [10.1016/j.trc.2018.02.021](https://doi.org/10.1016/j.trc.2018.02.021).
33. Sheppy M., Beach A., Pless S. NREL RSF Measured Data 2011. Nov. 2014. Accessed: 2023-09-03 DOI: [10.25984/1845288](https://doi.org/10.25984/1845288).
34. Reiss A. PAMAP2 Physical Activity Monitoring. 2012. Accessed: 2023-09-03 DOI: [10.24432/C5NW2H](https://doi.org/10.24432/C5NW2H). UCI Machine Learning Repository.
35. Юртин А.А. Восстановление многомерных временных рядов на основе выявления поведенческих шаблонов и применения автоэнкодеров // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. 2024. Т. 13, № 2. С. 39–55. DOI: [10.14529/cmse240203](https://doi.org/10.14529/cmse240203).
36. Bundesamt Für Umwelt – Swiss Federal Office for the Environment. Accessed: 2023-09-03. <https://www.hydrodaten.admin.ch/>.
37. Lozano A.C., Li H., Niculescu-Mizil A., *et al.* Spatial-temporal causal modeling for climate change attribution // Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009 / ed. by J.F. Elder, F. Fogelman-Soulié, P.A. Flach, M.J. Zaki. ACM, 2009. P. 587–596. DOI: [10.1145/1557019.1557086](https://doi.org/10.1145/1557019.1557086).
38. MeteoSwiss: Federal Office of Meteorology and Climatology. 2023. Accessed: 2023-09-03. <https://www.meteoswiss.admin.ch/services-and-publications/service/open-government-data.html>.

39. Weather Station Saaleaue, Max Planck Institute for Biogeochemistry, Germany. Accessed: 2023-09-03. https://www.bgc-jena.mpg.de/wetter/weather_data.html.
40. Khayati M., Lerner A., Tymchenko Z., Cudré-Mauroux P. Mind the Gap: An Experimental Evaluation of Imputation of Missing Values Techniques in Time Series // Proc. VLDB Endow. 2020. Vol. 13, no. 5. P. 768–782. DOI: [10.14778/3377369.3377383](https://doi.org/10.14778/3377369.3377383).
41. Minor B.D., Doppa J.R., Cook D.J. Learning Activity Predictors from Sensor Data: Algorithms, Evaluation, and Applications // IEEE Trans. Knowl. Data Eng. 2017. Vol. 29, no. 12. P. 2744–2757. DOI: [10.1109/TKDE.2017.2750669](https://doi.org/10.1109/TKDE.2017.2750669).
42. Цымблер М.Л., Полонский В.А., Юртин А.А. Об одном методе восстановления пропущенных значений потокового временного ряда в режиме реального времени // Вестник Южно-Уральского государственного университета. Серия: Вычислительная математика и информатика. 2021. Т. 10, № 4. С. 5–25. DOI: [10.14529/cmse210401](https://doi.org/10.14529/cmse210401).
43. Pontes F.J., F. de Amorim G. da, Balestrassi P.P., *et al.* Design of experiments and focused grid search for neural network parameter optimization // Neurocomputing. 2016. Vol. 186. P. 22–34. DOI: [10.1016/J.NEUCOM.2015.12.061](https://doi.org/10.1016/J.NEUCOM.2015.12.061).

Юртин Алексей Артемьевич, программист, Лаборатория больших данных и машинного обучения, аспирант кафедры системного программирования, Южно-Уральский государственный университет (национальный исследовательский университет) (Челябинск, Российская Федерация)

DOI: [10.14529/cmse240404](https://doi.org/10.14529/cmse240404)

TOWARDS A LOSS FUNCTION FOR TRAINING NEURAL NETWORK MODELS OF TIME SERIES IMPUTATION

© 2024 A.A. Yurtin

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

E-mail: iurtinaa@susu.ru

Received: 01.09.2024

In the article, we touch upon the problem of choosing a loss function for training neural network models for imputation of missing values of multidimensional time series and introduce a novel loss function called MPDE (Mean Profile Distance Error). The MPDE function for real and reconstructed m -length subsequences is calculated as the average of the distances between all pairs of ℓ -length sliding windows of these subsequences, where $\ell \leq m$ and above windows have the same starting points. The distance between two windows is a modification of the MPdist (matrix profile distance) similarity measure and is defined as the weighted sum of the Euclidean and z -normalized Euclidean distances between these windows. The above weights are taken from the range $[0,1]$ and are the hyperparameters of the loss function. When training a neural network model, MPDE allows taking into account the behavioral similarity of the compared subsequences through the presence of similar windows in them, regardless of the relative locations of these windows. Since MPDE has a high computational complexity, we implement a parallel algorithm for its calculation on a GPU to incorporate MPDE into deep learning frameworks. The algorithm is implemented using the PyTorch framework, where MPDE is represented as a sequence of automatically parallelizable operations with multidimensional tensors. Experiments over multidimensional time series from various subject domains showed that in 78% of cases state-of-the-art neural network models achieve their highest imputation accuracy (in terms of the RMSE metric) when using the proposed loss function; at the same time, the above models demonstrate imputation accuracy 40% higher than the average value achieved when using other loss functions.

Keywords: time series, imputation of missing values, neural networks, loss function, PyTorch.

FOR CITATION

Yurtin A.A. Towards a Loss Function for Training Neural Network Models of Time Series Imputation. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2024. Vol. 13, no. 4. P. 53–73. (in Russian) DOI: 10.14529/cmse240404.

This paper is distributed under the terms of the Creative Commons Attribution-Non Commercial 4.0 License which permits non-commercial use, reproduction and distribution of the work without further permission provided the original work is properly cited.

References

1. Majumdar S., Laha A.K. Clustering and classification of time series using topological data analysis with applications to finance. Expert Syst. Appl. 2020. Vol. 162. P. 113868. DOI: [10.1016/j.eswa.2020.113868](https://doi.org/10.1016/j.eswa.2020.113868).
2. Lara-Benitez P., Carranza-Garcia M., Luna-Romera J.M., Riquelme J.C. Temporal convolutional networks applied to energy-related time series forecasting. applied sciences. 2020. Vol. 10, no. 7. P. 2322. DOI: [10.3390/app10072322](https://doi.org/10.3390/app10072322).
3. Gratius N., Wang Z., Hwang M.Y., et al. Digital Twin Technologies for Autonomous Environmental Control and Life Support Systems. J. Aerosp. Inf. Syst. 2024. Vol. 21, no. 4. P. 332–347. DOI: [10.2514/1.I011320](https://doi.org/10.2514/1.I011320).
4. Zhou Z., Tang W., Li M., et al. A Novel Hybrid Intelligent SOPDEL Model with Comprehensive Data Preprocessing for Long-Time-Series Climate Prediction. Remote. Sens. 2023. Vol. 15, no. 7. P. 1951. DOI: [10.3390/RS15071951](https://doi.org/10.3390/RS15071951).
5. Kazijevs M., Samad M.D. Deep imputation of missing values in time series health data: A review with benchmarking. J. Biomed. Informatics. 2023. Vol. 144. P. 104440. DOI: [10.1016/J.JBI.2023.104440](https://doi.org/10.1016/J.JBI.2023.104440).
6. Fang C., Wang C. Time Series Data Imputation: A Survey on Deep Learning Approaches. CoRR. 2020. Vol. abs/2011.11347. arXiv: [2011.11347](https://arxiv.org/abs/2011.11347). URL: <https://arxiv.org/abs/2011.11347>.
7. Cao W., Wang D., Li J., et al. BRITS: Bidirectional Recurrent Imputation for Time Series. Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada / ed. by S. Bengio, H.M. Wallach, H. Larochelle, et al. 2018. P. 6776–6786. URL: <https://proceedings.neurips.cc/paper/2018/hash/734e6bfcd358e25ac1db0a4241b95651-Abstract.html>.
8. Yoon J., Zame W.R., Schaar M. van der Estimating Missing Data in Temporal Data Streams Using Multi-Directional Recurrent Neural Networks. IEEE Trans. Biomed. Eng. 2019. Vol. 66, no. 5. P. 1477–1490. DOI: [10.1109/TBME.2018.2874712](https://doi.org/10.1109/TBME.2018.2874712).
9. Fortuin V., Baranchuk D., Rätsch G., Mandt S. GP-VAE: Deep Probabilistic Time Series Imputation. The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]. Vol. 108 / ed. by S. Chiappa, R. Calandra. PMLR, 2020. P. 1651–1661. Proceedings of Machine Learning Research. URL: <http://proceedings.mlr.press/v108/fortuin20a.html>.

10. Du W., Côté D., Liu Y. SAITS: Self-attention-based imputation for time series. *Expert Syst. Appl.* 2023. Vol. 219. P. 119619. DOI: [10.1016/J.ESWA.2023.119619](https://doi.org/10.1016/J.ESWA.2023.119619).
11. Oh E., Kim T., Ji Y., Khyalia S. STING: Self-attention based Time-series Imputation Networks using GAN. 2021. DOI: [10.1109/ICDM51629.2021.00155](https://doi.org/10.1109/ICDM51629.2021.00155).
12. Kaya M., Bilge H.S. Deep Metric Learning: A Survey. *Symmetry*. 2019. Vol. 11, no. 9. P. 1066. DOI: [10.3390/SYM11091066](https://doi.org/10.3390/SYM11091066).
13. Wang Q., Ma Y., Zhao K., Tian Y. A comprehensive survey of loss functions in machine learning. *Annals of Data Science*. 2022. Vol. 9. P. 187–212. DOI: [10.1007/s40745-020-00253-5](https://doi.org/10.1007/s40745-020-00253-5).
14. Ciampiconi L., Elwood A., Leonardi M., *et al.* A survey and taxonomy of loss functions in machine learning. *CoRR*. 2023. Vol. abs/2301.05579. DOI: [10.48550/ARXIV.2301.05579](https://doi.org/10.48550/ARXIV.2301.05579). arXiv: [2301.05579](https://arxiv.org/abs/2301.05579).
15. Netrapalli P. Stochastic gradient descent and its variants in machine learning. *Journal of the Indian Institute of Science*. 2019. Vol. 99, no. 2. P. 201–213. DOI: [10.1007/s41745-019-0098-4](https://doi.org/10.1007/s41745-019-0098-4).
16. Xu J., Ren X., Lin J., Sun X. Diversity-Promoting GAN: A Cross-Entropy Based Generative Adversarial Network for Diversified Text Generation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018* / ed. by E. Riloff, D. Chiang, J. Hockenmaier, J. Tsujii. Association for Computational Linguistics, 2018. P. 3940–3949. DOI: [10.18653/V1/D18-1428](https://doi.org/10.18653/V1/D18-1428).
17. Chang Y., Wang X., Wang J., *et al.* A Survey on Evaluation of Large Language Models. *ACM Trans. Intell. Syst. Technol.* 2024. Vol. 15, no. 39. P. 1–45. DOI: [10.1145/3641289](https://doi.org/10.1145/3641289).
18. Chen W., Huang H., Peng S., *et al.* YOLO-face: a real-time face detector. *Vis. Comput.* 2021. Vol. 37, no. 4. P. 805–813. DOI: [10.1007/S00371-020-01831-7](https://doi.org/10.1007/S00371-020-01831-7).
19. Paszke A., Gross S., Massa F., *et al.* PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* / ed. by H.M. Wallach, H. Larochelle, A. Beygelzimer, *et al.* 2019. P. 8024–8035. URL: <https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html>.
20. Jadon A., Patil A., Jadon S. A Comprehensive Survey of Regression Based Loss Functions for Time Series Forecasting. *International Conference on Data Management, Analytics & Innovation*. Springer. 2024. P. 117–147. DOI: [10.1007/978-981-97-3245-6_9](https://doi.org/10.1007/978-981-97-3245-6_9).
21. Qi J., Du J., Siniscalchi S.M., *et al.* On Mean Absolute Error for Deep Neural Network Based Vector-to-Vector Regression. *IEEE Signal Process. Lett.* 2020. Vol. 27. P. 1485–1489. DOI: [10.1109/LSP.2020.3016837](https://doi.org/10.1109/LSP.2020.3016837).
22. Chen X., Liu W., Mao X., Yang Z. Distributed High-dimensional Regression Under a Quantile Loss Function. *J. Mach. Learn. Res.* 2020. Vol. 21, no. 182. P. 1–43. URL: <http://jmlr.org/papers/v21/20-297.html>.
23. Saleh R.A., Saleh A.K.M.E. Statistical Properties of the log-cosh Loss Function Used in Machine Learning. *CoRR*. 2022. Vol. abs/2208.04564. DOI: [10.48550/ARXIV.2208.04564](https://doi.org/10.48550/ARXIV.2208.04564). arXiv: [2208.04564](https://arxiv.org/abs/2208.04564).

24. Sun Q., Zhou W.-X., Fan J. Adaptive Huber regression. *Journal of the American Statistical Association*. 2020. Vol. 115, no. 529. P. 254–265. DOI: [10.1080/01621459.2018.1543124](https://doi.org/10.1080/01621459.2018.1543124).
25. Cuturi M., Blondel M. Soft-DTW: a Differentiable Loss Function for Time-Series. *Proceedings of the 34th International Conference on Machine Learning*. Vol. 70 / ed. by D. Precup, Y.W. Teh. PMLR, June 2017. P. 894–903. *Proceedings of Machine Learning Research*. URL: <https://proceedings.mlr.press/v70/cuturi17a.html>.
26. Berndt D.J., Clifford J. Using Dynamic Time Warping to find patterns in time series. *KDD Workshop*. 1994. P. 359–370. URL: <https://cdn.aaai.org/Workshops/1994/WS-94-03/WS94-03-031.pdf>.
27. Gharghabi S., Imani S., Bagnall A.J., *et al.* Matrix Profile XII: MPdist: A Novel Time Series Distance Measure to Allow Data Mining in More Challenging Scenarios. *IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018*. IEEE Computer Society, 2018. P. 965–970. DOI: [10.1109/ICDM.2018.00119](https://doi.org/10.1109/ICDM.2018.00119).
28. Zhuang J., Tang T., Ding Y., *et al.* AdaBelief Optimizer: Adapting Stepsizes by the Belief in Observed Gradients. *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* / ed. by H. Larochelle, M. Ranzato, R. Hadsell, *et al.* 2020. URL: <https://proceedings.neurips.cc/paper/2020/hash/d9d4f495e875a2e075a1a4a6e1b9770f-Abstract.html>.
29. Baydin A.G., Pearlmutter B.A., Radul A.A., Siskind J.M. Automatic Differentiation in Machine Learning: a Survey. *J. Mach. Learn. Res.* 2017. Vol. 18, no. 153. P. 1–43. URL: <https://jmlr.org/papers/v18/17-468.html>.
30. Bilenko R.V., Dolganina N.Y., Ivanova E.V., Rekachinsky A.I. High-performance Computing Resources of South Ural State University. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2022. Vol. 11, no. 1. P. 15–30. DOI: [10.14529/cmse220102](https://doi.org/10.14529/cmse220102).
31. Trindade A. Electricity Load Diagrams 2011–2014. 2015. DOI: [10.24432/C58C86](https://doi.org/10.24432/C58C86). UCI Machine Learning Repository.
32. Laña I., Olabarrieta I., Vélez M., Del Ser J. On the imputation of missing data for road traffic forecasting: New insights and novel techniques. *Transportation Research Part C: Emerging Technologies*. 2018. Vol. 90. P. 18–33. DOI: [10.1016/j.trc.2018.02.021](https://doi.org/10.1016/j.trc.2018.02.021).
33. Sheppy M., Beach A., Pless S. NREL RSF Measured Data 2011. Nov. 2014. Accessed: 2023-09-03 DOI: [10.25984/1845288](https://doi.org/10.25984/1845288).
34. Reiss A. PAMAP2 Physical Activity Monitoring. 2012. Accessed: 2023-09-03 DOI: [10.24432/C5NW2H](https://doi.org/10.24432/C5NW2H). UCI Machine Learning Repository.
35. Yurtin A.A. Imputation of Multivariate Time Series Based on the Behavioral Patterns and Autoencoders. *Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering*. 2024. Vol. 13, no. 2. P. 39–55. DOI: [10.14529/cmse240203](https://doi.org/10.14529/cmse240203).
36. Bundesamt Für Umwelt – Swiss Federal Office for the Environment. Accessed: 2023-09-03. <https://www.hydrodaten.admin.ch/>.

37. Lozano A.C., Li H., Niculescu-Mizil A., *et al.* Spatial-temporal causal modeling for climate change attribution. Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009 / ed. by J.F. Elder, F. Fogelman-Soulié, P.A. Flach, M.J. Zaki. ACM, 2009. P. 587–596. DOI: [10.1145/1557019.1557086](https://doi.org/10.1145/1557019.1557086).
38. MeteoSwiss: Federal Office of Meteorology and Climatology. 2023. Accessed: 2023-09-03. <https://www.meteoswiss.admin.ch/services-and-publications/service/open-government-data.html>.
39. Weather Station Saaleaue, Max Planck Institute for Biogeochemistry, Germany. Accessed: 2023-09-03. https://www.bgc-jena.mpg.de/wetter/weather_data.html.
40. Khayati M., Lerner A., Tymchenko Z., Cudré-Mauroux P. Mind the Gap: An Experimental Evaluation of Imputation of Missing Values Techniques in Time Series. Proc. VLDB Endow. 2020. Vol. 13, no. 5. P. 768–782. DOI: [10.14778/3377369.3377383](https://doi.org/10.14778/3377369.3377383).
41. Minor B.D., Doppa J.R., Cook D.J. Learning Activity Predictors from Sensor Data: Algorithms, Evaluation, and Applications. IEEE Trans. Knowl. Data Eng. 2017. Vol. 29, no. 12. P. 2744–2757. DOI: [10.1109/TKDE.2017.2750669](https://doi.org/10.1109/TKDE.2017.2750669).
42. Zymbler M.L., Polonsky V.A., Yurtin A.A. On One Method of Imputation Missing Values of a Streaming Time Series in Real Time. Bulletin of the South Ural State University. Series: Computational Mathematics and Software Engineering. 2021. Vol. 10, no. 4. P. 5–25. DOI: [10.14529/cmse210401](https://doi.org/10.14529/cmse210401).
43. Pontes F.J., F. de Amorim G. da, Balestrassi P.P., *et al.* Design of experiments and focused grid search for neural network parameter optimization. Neurocomputing. 2016. Vol. 186. P. 22–34. DOI: [10.1016/J.NEUCOM.2015.12.061](https://doi.org/10.1016/J.NEUCOM.2015.12.061).

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 ПИИ 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: 1.2.3 – Теоретическая информатика, кибернетика (физико-математические науки), 2.3.5 – Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей (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 <https://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,
зам. главного редактора 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»
2024 Том 13, № 4

16+

Техн. редактор А.В. Миних

Издательский центр Южно-Уральского государственного университета

Подписано в печать 29.11.2024. Дата выхода в свет 24.12.2024. Формат 60×84 1/8. Печать цифровая.
Усл. печ. л. 8,83. Тираж 500 экз. Заказ 307/446. Цена свободная.

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