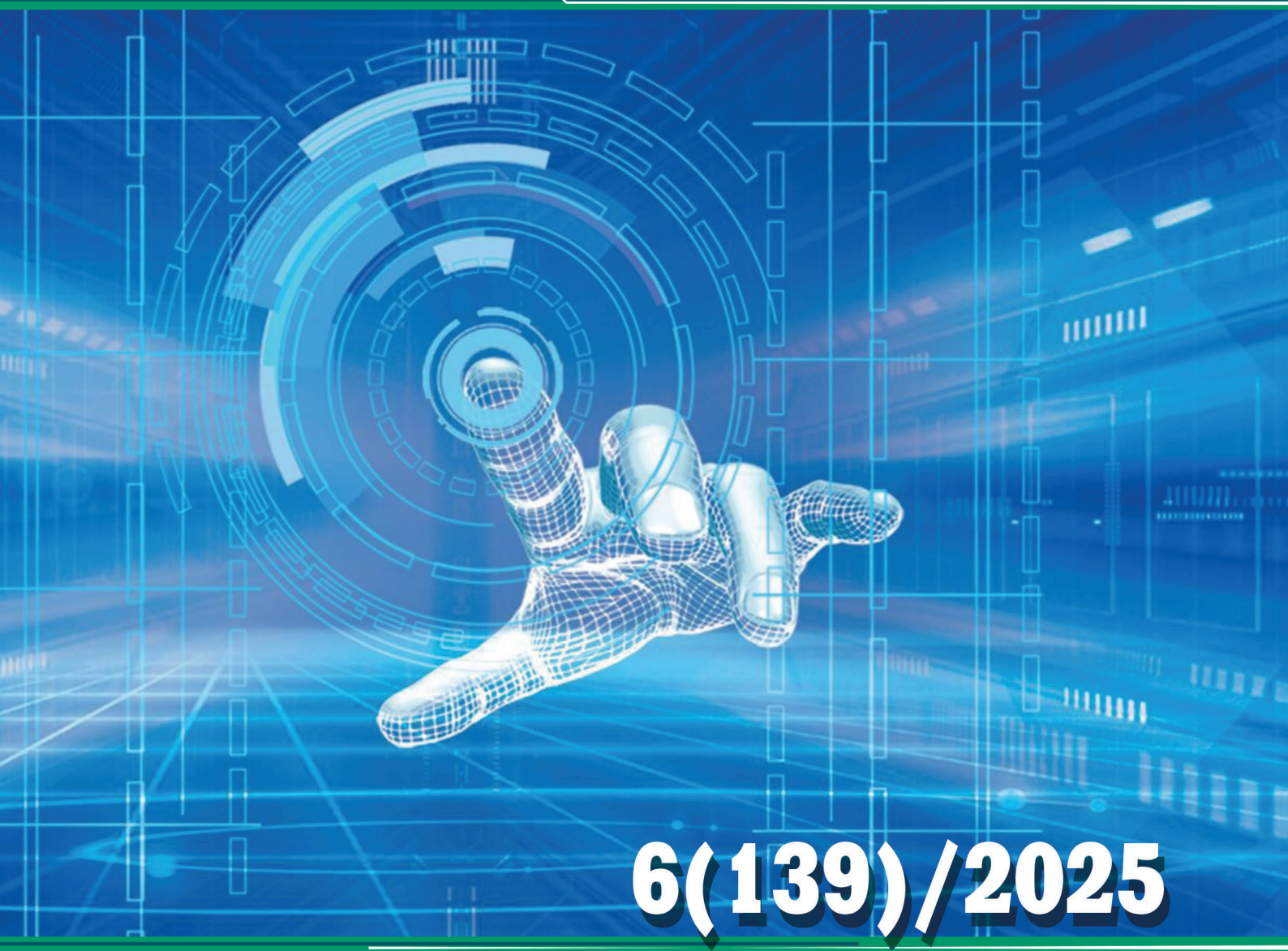
ISSN 1684-8853 (print); ISSN 2541-8610 (online)

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

НАУЧНЫЙ ЖУРНАЛ



6(139)/2025

doi:10.31799/1684-8853-2025-6 ISSN 1684-8853 (print); ISSN 2541-8610 (online)

SCIENTIFIC JOURNAL

6(139)/2025

PEER REVIEWED JOURNAL

INFORMATSIONNO-UPRAVLIAIUSHCHIE SISTEMY (INFORMATION AND CONTROL SYSTEMS)

Contact information
The Editorial and Publishing Center, SUAI
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia
Website: http://i-us.ru/en, e-mail: ius.spb@gmail.com
Tel.: +7 - 812 494 70 02

The Journal was registered in the Ministry of Press, Broadcasting and Mass Media of the Russian Federation.
Registration Certificate JD № 77-12412 from April, 19, 2002.
Re-registration in the Federal Service for Supervision in the Sphere of Telecom, Information Technologies and Mass Communications (ROSKOMNADZOR) due to change of the founder A. Vostrikov,
PI № FS 77-82226 from November, 23, 2021

Submitted for publication 05.11.25. Passed for printing 24.12.25. Signed for printing: 26.12.2025.
Format $60\times84_{1/8}$. Phototype CentSchbkCyrill BT. Digital printing.

Layout original is made at the Editorial and Publishing Center, SUAI.
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia
Printed from slides at the Editorial and Publishing Center, SUAI.
67A, Bol'shaya Morskaia, 190000, Saint Petersburg, Russia

The journal is indexed in Scopus.

Free distribution.

doi:10.31799/1684-8853-2025-6

ISSN 1684-8853 (print); ISSN 2541-8610 (online)

НАУЧНЫЙ ЖУРНАЛ

6(139)/2025

РЕЦЕНЗИРУЕМОЕ ИЗДАНИЕ

ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ

Адрес редакции: 190000, г. Санкт-Петербург,
ул. Большая Морская, д. 67, лит. А, ГУАП, РИЦ
Тел.: (812) 494-70-02, эл. адрес: ius.spb@gmail.com,
сайт: http://i-us.ru

Журнал входит в БД Scopus и в Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук.

Сдано в набор 05.11.25. Подписано в печать 24.12.25. Дата выхода в свет: 26.12.2025.
Формат 60×841/8. Гарнитура CentSchbkCyrill BT. Печать цифровая.
Усл. печ. л. 14,1. Уч.-изд. л. 14,3. Тираж 1000 экз (1-й завод 50 экз.). Заказ № 426.
Оригинал-макет изготовлен в редакционно-издательском центре ГУАП.
190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А.
Отпечатано в редакционно-издательском центре ГУАП.
190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А.
Распространяется бесплатно.

Журнал зарегистрирован в Министерстве РФ по делам печати, телерадиовещания и средств массовых коммуникаций.
Свидетельство о регистрации ПИ № 77-12412 от 19 апреля 2002 г.
Перерегистрирован в Роскомнадзоре.
Свидетельство о регистрации ПИ № ФС77-82226 от 23 ноября 2021 г.

УДК 004.896
doi:10.31799/1684-8853-2025-6-2-14
EDN: DECQWI

Научные статьи
Articles

Гибридный алгоритм глобального планирования и локального взаимодействия для перехвата целей роем БПЛА

А. А. Акимов[а], *канд. физ.-мат. наук, доцент, orcid.org/0000-0003-3387-2959*
Ю. А. Гнатенко[б], *канд. физ.-мат. наук, доцент, orcid.org/0009-0009-9264-3989, y.a.gnatenko@struust.ru*
Р. Г. Болбаков[а], *канд. техн. наук, доцент, orcid.org/0000-0002-4922-7260*
[а]*МИРЭА — Российский технологический университет, Вернадского пр., 78, Москва, 119454, РФ*
[б]*Стерлитамакский филиал Уфимского университета науки и технологий, Ленина пр., 49, Стерлитамак, 453103, РФ*

Введение: *перехват целей роем БПЛА при отсутствии централизованного управления требует сочетания быстрого сближения с целями и строгого контроля междроновых дистанций. При этом алгоритмы глобальной оптимизации не всегда учитывают локальные ограничения, а поведенческие правила ограничены в возможности управления сложными сценариями.* ***Цель:*** *разработать алгоритм децентрализованного управления роем БПЛА, который обеспечит сокращение времени перехвата и одновременно исключит столкновения между агентами в рое.* ***Методы:*** *объединены алгоритм серых волков и модель Boids. Глобальный модуль алгоритма серых волков направляет дроны к целям, локальные правила Boids регулируют относительное движение дронов и предотвращают сближения ниже допустимой дистанции.* ***Результаты:*** *разработан гибридный алгоритм, обеспечивающий устойчивое выполнение задач перехвата в сценариях с неподвижной и движущейся целью. Медианное время захвата составляет 10–15 итераций для стационарной цели и 30–40 итераций для движущейся. Минимальная междроновая дистанция во всех экспериментах оставалась выше допустимой (5 м). Доказано условие безопасности через коэффициент разделения, гарантирующее отсутствие коллизий при сближении агентов. Для работы с несколькими объектами предложена кластеризация, которая позволяет разделять рой на группы и координировать перехват даже при разнесенном расположении целей.* ***Практическая значимость:*** *алгоритм использует малое число параметров, масштабируется по числу агентов и пригоден для бортовой реализации в реальном времени.* ***Обсуждение:*** *результаты показывают, что гибридный алгоритм сочетает преимущества глобальной оптимизации и локальной координации, обеспечивая баланс между скоростью перехвата и безопасностью. Это создает основу для дальнейших исследований в трехмерной динамике и для сценариев со сложными препятствиями.*

Ключевые слова *— рой БПЛА, алгоритм серых волков, Boids, перехват цели, децентрализованное управление, избегание столкновений, метаэвристика.*

Для цитирования: Акимов А. А., Гнатенко Ю. А., Болбаков Р. Г. Гибридный алгоритм глобального планирования и локального взаимодействия для перехвата целей роем БПЛА. *Информационно-управляющие системы*, 2025, № 6, с. 2–14. doi:10.31799/1684-8853-2025-6-2-14, EDN: DECQWI
For citation: Akimov A. A., Gnatenko Y. A., Bolbakov R. G. Hybrid algorithm of global planning and local interaction for target interception by UAV swarms. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 2–14 (In Russian). doi:10.31799/1684-8853-2025-6-2-14, EDN: DECQWI

Введение

Массовое использование беспилотных летательных аппаратов (БПЛА) привело к переходу от задач одиночных вылетов к миссиям роев, выполняемых в условиях ограниченных ресурсов, неполной осведомленности и нестабильных каналов связи [1]. В прикладных сценариях (мониторинг, противодействие БПЛА, охрана периметра, поисково-спасательные операции) важно не только достижение цели, но и удержание контроля над ней, что требует кооперации аппаратов для блокировки маневров, компенсации отказов и работы с несколькими целями при строгих требованиях безопасности [2, 3].

Рост размерности и стохастичности среды обусловил необходимость применения метаэвристик и методов роевой оптимизации, обеспечивающих поиск глобального экстремума без вычисления производных целевой функции и при малом числе настраиваемых параметров [4, 5]. Методы оптимизации дополняются локальными правилами, которые формируют устойчивые построения и безопасные дистанции. Для одновременного достижения быстрого сближения, децентрализации и формальных гарантий безопасности необходимо сочетать глобальное распределение усилий с локальной координацией.

В работе рассматривается задача кооперативного перехвата, в которой рой перехватчиков должен за минимальное время обеспечить геометрический захват одной или нескольких целей, т. е. каждая цель должна оказаться в пределах заданного радиуса хотя бы одного перехватчика.

На всем горизонте в процессе миссий поддерживается минимальная междроновая дистанция и действуют ограничения на скорость и ускорение. Центральный диспетчер отсутствует, обмен данными возможен только с ближайшими соседями. Следовательно, управление должно быть децентрализованным и вычислительно легким.

Актуальность задачи двойная: гражданское применение (наблюдение) и противодействие БПЛА, где одиночный аппарат не гарантирует блокировки и отказоустойчивости. Цель — разработать децентрализованный алгоритм планирования и управления роем, минимизирующий время захвата при сетевых и эксплуатационных ограничениях.

Обзор литературы и методов

Задачи управления роем относятся к классу «черного ящика», где целевая функция неизвестна или недифференцируема, а среда стохастична. Для таких задач применимы методы роевой оптимизации (swarm intelligence), основанные на локальных взаимодействиях [6]. К ним относятся метод роя частиц [7, 8], муравьиный алгоритм, пчелиный алгоритм, алгоритм летучих мышей [9, 10]. Такие характеристики, как простота, малое число параметров, независимость от производных целевой функции и устойчивость к локальным экстремумам, способствуют широкому применению роевых методов в задачах маршрутизации и распределения [11].

Особое место занимает алгоритм серых волков (Grey Wolf Optimizer, GWO), предложенный С. Мирджалили с соавторами в 2014 г. [4]. Он использует трехуровневую иерархию поиска с линейным убывающим коэффициентом сходимости, что при малом числе параметров приводит к быстрой адаптации, включая планирование траекторий БПЛА [12]. Позднее появились модификации: нелинейные и адаптивные законы эволюции коэффициентов (Nonlinear Improved GWO, NI-GWO), многостадийные популяции, гибриды с потенциальными полями и реактивными планировщиками [13]. В версии NI-GWO коэффициент сходимости меняется по заданному закону, что улучшает баланс разведки и атаки [14], снижает риск преждевременной сходимости и ускоряет поиск. Однако такие модификации не гарантируют локальную безопасность; межагентное взаимодействие отсутствует или задается внешними модулями. Это требует интеграции с дополнительными средствами, например потенциальными полями или методом динамического окна, что увеличивает число гиперпараметров и не гарантирует безопасность [11–14].

Параллельно развивались близкие по идее к GWO «волчьи» методы (Wolf Pack Search), использующие поведенческие схемы стаи для глобальной оптимизации без градиентов [15]. Для многоагентных сценариев предложены варианты с принципом «достаточности» (satisficing), ускоряющие поиск в условиях дефицита времени и неполной информации [16, 17].

Локальную безопасность и построение вблизи цели обеспечивает модель Boids, предложенная К. Рейнольдсом в 1987 г. [18]. Три правила (разделение, выравнивание, сцепление) формируют согласованное движение без централизованного управления. Модель вычислительно легкая и хорошо сочетается с глобальными поисковыми методами, сглаживая траектории и поддерживая межагентные расстояния.

Предполагается, что комбинация GWO и Boids осуществит быстрое сближение без диспетчера и устойчивую координацию с контролем дистанций. Благодаря малому числу параметров и простоте реализации она подходит для децентрализованного управления в условиях ограниченных ресурсов. Алгоритм создает баланс между скоростью глобального поиска и безопасностью локальных взаимодействий, отвечая требованиям реального времени для кооперативного перехвата.

Математическая модель управления и перехвата целей для роя БПЛА

Формализация задачи для роя из N БПЛА выполнена в дискретном времени $k = 0, 1, 2, \ldots$ с шагом $\Delta t > 0$.

1. Состояние перехватчиков и кинематика. Для перехватчика $i, i = \overline{1, N}$ вводятся положение $\mathbf{p}_i(k) \in \mathbf{R}^d$, скорость $\mathbf{v}_i(k) \in \mathbf{R}^d$, управляющее ускорение $\mathbf{u}_i(k) \in \mathbf{R}^d$, $d = 2, 3$. Эволюция задается моделью первого порядка

$$\mathbf{v}_i(k+1) = \mathbf{v}_i(k) + \mathbf{u}_i(k)\Delta t;$$
$$\mathbf{p}_i(k+1) = \mathbf{p}_i(k) + \mathbf{v}_i(k+1)\Delta t \qquad (1)$$

с ограничениями

$$\|\mathbf{u}_i(k)\| \le a_{\max}, \ \|\mathbf{v}_i(k)\| \le V_{\max}, \qquad (2)$$

где $a_{\max}$, $V_{\max}$ — максимальные ускорение и скорость соответственно; $\|\cdot\|$ — евклидова норма. Начальные условия $\{\mathbf{p}_i(0), \mathbf{v}_i(0)\}$ заданы. Выбор уравнений (1) согласован с частотами обновления бортовых автопилотов и упрощает интеграцию алгоритмов планирования.

2. Цели и их траектории. Пусть M целей описываются положениями

$$\mathbf{q}_m(k) \in \mathbf{R}^d, \ m = 1..M, \ d = 2, 3.$$

Допускаются стационарные и движущиеся цели; модель движения целей не фиксируется и рассматривается как экзогенная, измеряемая с требуемой частотой. Препятствия не учитываются, что позволяет изолированно оценивать влияние кооперативной координации.

3. Ограничения безопасности и допустимость управления. Безопасность задается минимально допустимой междроновой дистанцией $d_{\min} > 0$:

$$\forall k,\ \forall i \neq j:\ \left\|\mathbf{p}_i(k) - \mathbf{p}_j(k)\right\| \geq d_{\min}.$$

Управляющие воздействия формируются децентрализованно. Допустимые управления имеют вид

$$\mathbf{u}_i(k) = \mu_i(\mathbf{p}_i(k), \mathbf{v}_i(k),$$
$$\{\mathbf{p}_i(0), \mathbf{v}_i(0)\}_{j\in N_i(k)}, \{\mathbf{q}_m(k)\}_{m=1}^{M}),$$

где μ_i — закон управления перехватчика i; $N_i(k)$ — множество ближайших соседей по доступной связи/видимости.

4. Критерии захвата и метрики качества. Момент захвата цели m — это первый момент времени T_m, для которого существует перехватчик i такой, что $\|\mathbf{p}_i(T_m) - \mathbf{q}_i(T_m)\| \leq R_{cap}$, R_{cap} — радиус захвата. Полное время перехвата $T_{cap} = \max\limits_{m=\overline{1,M}} T_m$.

Для мониторинга прогресса вводятся сводные метрики

$$D_{\max}(k) = \max_m \min_i \left\|\mathbf{p}_i(k) - \mathbf{q}_m(k)\right\|;$$
$$D_{\min}(k) = \min_{i\neq j} \left\|\mathbf{p}_i(k) - \mathbf{p}_j(k)\right\|.$$

Условие завершения миссии эквивалентно

$$D_{\max}(T) \leq R_{cap}; \qquad (3)$$

условие безопасности требует для всех k

$$D_{\min}(k) \geq d_{\min}. \qquad (4)$$

5. Постановка оптимизационной задачи. Требуется найти децентрализованные стратегии $\{\mu_i\}_{i=1}^{N}$, минимизирующие время перехвата T_{cap}, при соблюдении для всех перехватчиков ограничений (2)–(4). Альтернативно можно рассматривать эквивалентную задачу минимизации $D_{\max}(k)$ на каждом дискретном шаге k при жестком ограничении (4) и критерии остановки (3).

Отсутствие препятствий необходимо для раздельной оценки глобального сближения и локальной координации. Обобщение модели для сложной среды выполняется добавлением множества препятствий и соответствующих ограничений расстояний без изменения структуры пунктов 1–5.

Представленная модель задает единое пространство допусков и критериев для алгоритма планирования и управления роем БПЛА при перехвате целей.

Алгоритм серых волков

Серый волк (Canis lupus) — пример социального хищника с жесткой иерархией. Решения принимает α-волк, поддерживаемый β-волком, готовым заменить лидера. Ниже стоят δ-особи — опытные «старшие», совмещающие функции разведчиков, охотников и стражей. Замыкает пирамиду ω-волк, играющий роль «социального клапана», снижающего внутренние конфликты. Такое распределение обязанностей обеспечивает устойчивость стаи в условиях неопределенности [19, 20].

Алгоритм GWO транслирует следующую стратегию в пространство решений: слежение и сокращение дистанции; окружение и лишение добычи возможности маневра; финальный синхронизированный бросок.

Популяция моделируется «волками»-кандидатами, где α, β и δ задают ориентиры поиска, а ω-особи движутся на основе их опыта. Ранние итерации процесса ставят в приоритет разведку (аналог первому этапу охоты), средние — формирование «кольца» вокруг оптимума, а заключительные сосредоточены на точном «ударе» по глобальному минимуму.

Фаза Encircling (окружение добычи). Математическая схема поведения «волчьей стаи» описывается двумя ключевыми соотношениями [21]:

1) определение вектора смещения $\mathbf{D}_j = |\mathbf{C}_j \times \mathbf{X}_p - \mathbf{X}_j|$, где операция $|\cdot|$ — покомпонентный модуль вектора;

2) обновление координат j-го агента $\mathbf{X}_j(k+1) = \mathbf{X}_q - \mathbf{A}_j \times \mathbf{D}_j$, где $\mathbf{X}_j \in \mathbf{R}^d$, $d = 2, 3$ — координаты j-го «волка» на итерации k; $\mathbf{X}_q$ — позиция добычи; $k_{\max}$ — плановое число итераций; операция × — покомпонентное умножение; $\mathbf{C}_j$, $\mathbf{A}_j \in \mathbf{R}^d$ — стохастические коэффициент-векторы, которые управляют взаимодействием дронов с лидерами и вычисляются как

$$\mathbf{A}_j(k) = 2\mathbf{a}(k) \times \mathbf{r}_j^1(k) - \mathbf{a}(k),\ \mathbf{C}_j = 2\mathbf{r}_j^2(k),$$

$$\mathbf{r}_j^i \sim U(0,1)^d,\ i = 1, 2.$$

Индекс j указывает, для какого дрона рассчитывается коэффициент; $\mathbf{r}_j^i$ — независимые случайные векторные величины в диапазоне $(0, 1)^d$ для каждого j, k, i; $\mathbf{a}(k)$ — «радиус охоты»,

векторный параметр, обеспечивающий баланс поиска и атаки цели, компоненты которого являются монотонно убывающими функциями. Без ограничения общности рассуждений далее положим

$$a_m(k) = 2\left(1 - \frac{k}{k_{\max}}\right),\ m = 1..d.$$

Интерпретация коэффициентов:

1) $\mathbf{D}_j$ — «дистанция» до добычи, масштабированная случайным фактором $\mathbf{C}_j$;

2) при больших значениях $a_m(k)$ $(a_m(k) \approx 2)$ компоненты $\mathbf{A}_j$ принадлежат сегменту [–2, 2] и позволяют «волку» совершать крупные шаги, усиливая разведку пространства;

3) при малых значениях $a_m(k)$ $(a_m(k) \approx 0)$ компоненты $\mathbf{A}_j$ стремятся к нулю, и в этом случае происходит локальная корректировка траектории движения;

4) случайные векторы $\mathbf{r}^1$ и $\mathbf{r}^2$ препятствуют преждевременной сходимости и помогают «волкам» покинуть локальные минимумы.

Фаза Hunting (кооперативная охота). Во время охоты стая ориентируется на три точки притяжения — координаты лучших особей α, β и δ с позициями $\mathbf{X}_\alpha$, $\mathbf{X}_\beta$, $\mathbf{X}_\delta$. Для каждого агента j на шаге k выполняются три вычислительных блока.

1. Формирование векторов напряжения относительно лидеров:

$$\mathbf{A}_{jh}(k) = 2\mathbf{a}(k) \times \mathbf{r}^1_{jh}(k) - \mathbf{a}(k),\ \mathbf{C}_{jh} = 2\mathbf{r}^2_{jh}(k),$$

$$\mathbf{r}^i_{jh} \sim U(0,\ 1)^d,\ i = 1,\ 2,\ d = 2,\ 3,$$

$$\mathbf{D}_{jh} = \left|\mathbf{C}_{jh} \times \mathbf{X}_h - \mathbf{X}_j\right|,\ h \in \{\alpha, \beta, \delta\}.$$

2. Построение «виртуальных» позиций:

$$\mathbf{X}^h_j(k) = \mathbf{X}_h(k) - \mathbf{A}_{jh}(k) \times \mathbf{D}_{j\alpha}(k),\ h \in \{\alpha, \beta, \delta\}.$$

3. Усредненная позиция агента:

$$\mathbf{X}^{\text{new}}_j(k+1) = \frac{\mathbf{X}^\alpha_j(k) + \mathbf{X}^\beta_j(k) + \mathbf{X}^\delta_j(k)}{3}.$$

Каждая пара $(\mathbf{A}_{jh}, \mathbf{C}_{jh})$ генерируется независимо для $h \in \{\alpha, \beta, \delta\}$, что обеспечивает равное влияние всех лидеров и гарантирует расположение агента $\mathbf{X}^{\text{new}}_j(k+1)$ внутри многомерного многоугольника, заданного ими. Это создает устойчивое «сжатие» стаи. Вновь появившееся лучшее решение получает ранг α, а прежние лидеры переходят в β и δ. На каждой итерации выбираются три лучших кандидата: $\alpha = \arg\min_j f(\mathbf{X}_j)$, $\beta = \arg\min_{j \neq \alpha} f(\mathbf{X}_j)$, $\delta = \arg\min_{j \notin \{\alpha,\beta\}} f(\mathbf{X}_j)$, где f — целевая функция («дистанция до добычи»).

Модификация GWO для множественного перехвата на базе кластерного представления целей

В классическом GWO «добычей» считается точка $\mathbf{X}_q$, совпадающая с позицией лучшего агента (α). В задаче перехвата БПЛА целей обычно несколько, поэтому в качестве цели берут геометрический центр, который, как правило, лежит в пустом пространстве и не отражает реальных угроз. Чтобы сохранить иерархию α, β, δ и управлять несколькими мишенями, вводится многошаговый механизм.

Шаг 1. Сбор данных о целях. Пусть имеется M «красных» дронов, текущие координаты которых $\{\mathbf{q}_m\}_{m=1}^M \subset \mathbf{R}^d$, $d = 2, 3$.

Шаг 2. Если цели расположены разреженно, рой делится на несколько групп, например три. С помощью кластеризации K-means ($K = 3$) «красные» дроны разделяются на три кластера и представляются центроидами $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$. Если целей меньше трех, они дублируются для сохранения структуры лидеров. Каждый кластер можно далее делить на три подгруппы, поэтому рассуждения применимы к любой тройке центроидов.

Шаг 3. Назначение лидеров стаи. Среди N перехватчиков («синих» дронов) $\{\mathbf{p}_j\}_{j=1}^N$ выбираем ближайших к каждому центроиду:

$$\alpha = \arg\min_j \left\|\mathbf{p}_j - \mathbf{c}_1\right\|,\ \beta = \arg\min_{j \neq \alpha} \left\|\mathbf{p}_j - \mathbf{c}_2\right\|,$$

$$\delta = \arg\min_{j \notin \{\alpha,\beta\}} \left\|\mathbf{p}_j - \mathbf{c}_3\right\|.$$

Если один перехватчик ближе сразу к двум центроидам, роль второго или третьего лидера получает следующий по расстоянию аппарат, чтобы α, β, δ всегда соответствовали разным дронам.

Шаг 4. Модификация фаз «охоты» и «обхода». Для каждого перехватчика j (включая лидеров) вычисляются три предложения окружения (по одному на каждый центроид):

$$\mathbf{X}^{(i)}_j(k) = \mathbf{c}_i - \mathbf{A}^{(i)}_j(k) \times \mathbf{D}^{(i)}_j(k),\ i = 1, 2, 3;$$

$$\mathbf{A}^{(i)}_j(k) = 2\mathbf{a}(k) \times \mathbf{r}^{1,(i)}_j(k) - \mathbf{a}(k),\ \mathbf{C}^{(i)}_j = 2\mathbf{r}^{2,(i)}_j(k),$$

$$\mathbf{D}^{(i)}_j = \left|\mathbf{C}^{(i)}_j \times \mathbf{c}_i - \mathbf{X}_j\right|,$$

где $\mathbf{r}^{1,(i)}_j, \mathbf{r}^{2,(i)}_j \sim U(0,\ 1)^d$, $d = 2, 3$ генерируются самостоятельно для каждого «волка» и каждого центроида, повышая стохастическое разнообразие поведения.

Шаг 5. Усреднение предложений. Глобально обновленная позиция «волка» определяется как

$$\mathbf{X}^{\text{GWO}}_j(k) = \frac{\mathbf{X}^{(1)}_j(k) + \mathbf{X}^{(2)}_j(k) + \mathbf{X}^{(3)}_j(k)}{3}. \qquad (5)$$

Новая точка всегда лежит внутри выпуклого многогранника, образованного центроидами $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$, и учитывает вклад каждого кластера целей. Такая схема позволяет рою одновременно реагировать на разнесенные угрозы при сохранении вычислительной простоты GWO.

Шаг 6. Динамическое переназначение лидеров. Так как цели подвижны, кластеры и центроиды $\mathbf{c}_i(k)$ пересчитываются на каждой итерации, после чего ближайшие перехватчики вновь получают роли α, β, δ. Это обеспечивает адаптивность, т. е. при сближении групп или появлении новых целей рой автоматически перераспределяет ответственность без внешнего вмешательства и без усложнения вычислений.

В итоге кластеризованное расширение GWO сохраняет простоту, малое число параметров и глобальные свойства исходного метода, одновременно позволяя управлять множеством разнесенных мишеней и направлять дроны в наиболее критические зоны.

Применение модели Boids для локального избегания столкновений

Расширенный GWO задает глобальное движение к центроидам целей. На практике необходим автономный механизм, предотвращающий столкновения и поддерживающий компактное, но разреженное построение на завершающей фазе. Эту задачу решает модель Boids [18], основанная на трех простых правилах.

1. Разделение (Separation). При чрезмерном сближении с соседом j дрон i создает отталкивающее усилие

$$\mathbf{F}_S(i) = -\sum_{j\in N_i} f_s(r_{ij})\frac{\mathbf{p}_j - \mathbf{p}_i}{r_{ij}},\ f_s(r_{ij}) = \frac{k_s}{r_{ij}},$$

где N_i — множество соседей, попавших в радиус обзора дрона i; $r_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\|$ — взаимная дистанция между дронами i и j; $\mathbf{p}_j$, $\mathbf{p}_i$ — их позиции; $k_s > 0$ — коэффициент силы разделения.

2. Выравнивание (Alignment). Дрон стремится согласовать собственную скорость с усредненным вектором скоростей ближайших соседей

$$\mathbf{F}_A(i) = k_a(\bar{\mathbf{v}}_{N_i} - \mathbf{v}_i),\ \bar{\mathbf{v}}_{N_i} = \frac{1}{N_i}\sum_{j\in N_i}\mathbf{v}_j,$$

где $k_a > 0$ — коэффициент выравнивания; $\mathbf{v}_i$ — скорость дрона i.

3. Сцепление (Cohesion). Чтобы удерживать рой в компактной форме, вводится притяжение к центру масс местных соседей

$$\mathbf{F}_C(i) = k_c(\bar{\mathbf{p}}_{N_i} - \mathbf{p}_i),\ \bar{\mathbf{p}}_{N_i} = \frac{1}{N_i}\sum_{j\in N_i}\mathbf{p}_j,$$

где $k_c > 0$ — коэффициент сцепления.

Суммарное локальное ускорение формируется как

$$\mathbf{u}_i^{\text{Boids}} = \mathbf{F}_S(i) + \mathbf{F}_A(i) + \mathbf{F}_C(i). \qquad (6)$$

Разделение действует на малых дистанциях, строго препятствуя нарушению допустимого порога $d_{\min}$. Выравнивание и сцепление работают на большем радиусе, обеспечивая плавную подстройку и сохранение формы кольца.

Главная ценность модели Boids — ее локальность. Каждый аппарат ориентируется лишь на ближайших соседей N_i, что упрощает распределенную реализацию и исключает централизованное управление.

Комбинированный алгоритм GWO + Boids

1. Суперпозиция глобального и локального ускорений. Пусть на k-й итерации у перехватчика i заданы: $\mathbf{X}_i^{\text{GWO}}(k)$ — глобальная «целевая» позиция, рассчитанная по расширенным уравнениям GWO (5); $\mathbf{u}_i^{\text{Boids}}(k)$ — локальное ускорение, полученное на основании правил Boids (6); $\mathbf{p}_i(k)$ — текущий вектор позиции.

Обозначим весовой коэффициент глобального подхода $\mathbf{w}_g(k) = \dfrac{1}{2\sqrt{d}}\mathbf{a}(k)$.

Тогда результирующее ускорение определяется линейной суперпозицией

$$\mathbf{u}_i(k) = \mathbf{w}_g(k)\times(\mathbf{X}_i^{\text{GWO}}(k) - \mathbf{p}_i(k)) + \mathbf{u}_i^{\text{Boids}}(k).$$

На ранних итерациях $a_m(k) \approx 2$, следовательно, $\|w_g\| \approx 1$. Глобальная компонента задает фазу разведки и быстрое сближение, а локальные силы лишь предотвращают коллизии. Ближе к концу итерационного процесса $a_m(k) \approx 0$, значит, $\|w_g\| \approx 0$, поэтому доминирует динамика Boids, обеспечивающая точное окружение и удержание формации, тогда как «охотничье» ускорение становится незначимым.

На практике после вычисления $\mathbf{u}_i(k)$ его модуль нормируют до $\|\mathbf{u}_i\| \le a_{\max}$, обновляют скорость и позицию перехватчика:

$$\mathbf{v}_i \leftarrow \mathbf{v}_i + \mathbf{u}_i\Delta t,\ \|\mathbf{v}_i\| \le V_{\max},\ \mathbf{p}_i \leftarrow \mathbf{p}_i + \mathbf{v}_i\Delta t.$$

2. Коллизионная безопасность. Покажем, что локальная компонента $\mathbf{u}_i^{\text{Boids}}(k)$ гарантирует ненулевую буферную дистанцию между любыми

двумя дронами, даже если глобальная сила стремится их «сжать».

Рассмотрим пару аппаратов i, j и обозначим

$$\mathbf{n}_{ij} = \frac{\mathbf{p}_j - \mathbf{p}_i}{\left\|\mathbf{p}_j - \mathbf{p}_i\right\|},\ d_{ij} = \left\|\mathbf{p}_j - \mathbf{p}_i\right\|.$$

Проекция результирующего ускорения $\mathbf{u}_i$ на направление $\mathbf{n}_{ij}$ равна

$$\langle \mathbf{n}_{ij}, \mathbf{u}_i \rangle = \langle \mathbf{n}_{ij}, \mathbf{u}_i^{\text{Boids}} \rangle + \\ + \langle \mathbf{n}_{ij}, \mathbf{w}_g(k) \times \left(\mathbf{X}_i^{\text{GWO}}(k) - \mathbf{p}_i(k) \right) \rangle.$$

При $d_{ij} \le d_{\min}$ вклад правила Separation оценивается неравенством

$$\langle \mathbf{n}_{ij}, \mathbf{F}_S \rangle \le \frac{k_s}{d_{\min}^2},$$

тогда как глобальная составляющая, проецируемая на ту же ось, не превосходит w_g. Получаем верхнюю границу

$$\langle \mathbf{n}_{ij}, \mathbf{u}_i \rangle \le -\frac{k_s}{d_{\min}^2} + \left\|\mathbf{w}_g\right\| \cos\theta \le -\frac{k_s}{d_{\min}^2} + \left\|\mathbf{w}_g\right\|.$$

При выборе параметра разделения $k_s > d_{\min}^2$ и условии $\|\mathbf{w}_g\| \le 1$ получаем отрицательную проекцию ускорения $\langle \mathbf{n}_{ij}, \mathbf{u}_i \rangle < 0$, т. е. оно направлено на увеличение дистанции, и выход за барьер $d_{\min}$ невозможен. Поскольку рассуждение симметрично по индексам, безопасность обеспечена для всего роя.

Таким образом, сочетание глобального механизма GWO и локальной кинематики Boids формирует согласованную систему, где достигается быстрое перехватывание целей при строгом соблюдении междронового буфера.

Пошаговая схема алгоритма GWO + Boids

Рассмотрим гибридный алгоритм для перехвата целей роем дронов, объединяющий глобальный поиск (GWO) и локальное управление (Boids).

1. Инициализация.

1.1. Считать текущие координаты $\{\mathbf{q}_m\}_{m=1}^{M}$ «красных» дронов (целей).

1.2. Задать для N перехватчиков начальные позиции $\mathbf{p}_j(0)$ и скорости $\mathbf{v}_j(0)$.

1.3. Задать параметры $k_{\max}$, Δt, $a_{\max}$, $V_{\max}$, $d_{\min}$, k_s, k_a, k_c.

1.4. Установить $k = 0$, тогда начальный «радиус охоты» $a_m(0) = 2$, $m = 1..d$.

2. Кластеризация целей.

2.1. Если $M \ge 3$, применить метод кластеризации K-means ($K = 3$) на точках $\mathbf{q}_m$, получить центроиды $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$.

2.2. Если $M < 3$, то дублировать точки целей для получения трех центроидов.

3. Выбор лидеров стаи (α, β, δ).

Определить трех перехватчиков, ближайших к $\mathbf{c}_1$, $\mathbf{c}_2$, $\mathbf{c}_3$:

$$\alpha = \arg\min_j \left\|\mathbf{p}_j - \mathbf{c}_1\right\|,\ \beta = \arg\min_{j \ne \alpha} \left\|\mathbf{p}_j - \mathbf{c}_2\right\|,$$

$$\delta = \arg\min_{j \notin \{\alpha, \beta\}} \left\|\mathbf{p}_j - \mathbf{c}_3\right\|.$$

При совпадении брать следующего по расстоянию, чтобы роли были у трех различных дронов.

4. Глобальное обновление по GWO для каждого перехватчика j.

4.1. Сгенерировать три пары случайных векторов $\mathbf{r}^{(1,(i))}, \mathbf{r}^{(2,(i))} \sim U(0, 1)^d$, $i = 1, 2, 3$, $d = 2, 3$.

4.2. Вычислить коэффициенты $\mathbf{A}_j^{(i)} = 2\mathbf{a}(k) \times$ $\times \mathbf{r}^{(1,(i))} - \mathbf{a}(k)$, $\mathbf{C}_j^{(i)} = 2\mathbf{r}^{(2,(i))}$.

4.3. Для каждого центроида $\mathbf{c}_i$ рассчитать

$$\mathbf{D}_j^{(i)} = \left|\mathbf{C}^{(i)} \times \mathbf{c}_i - \mathbf{p}_j\right|,\ \mathbf{X}_j^{(i)} = \mathbf{c}_i - \mathbf{A}_j^{(i)} \times \mathbf{D}_j^{(i)}.$$

4.4. Усреднить предлагаемые позиции

$$\mathbf{X}_j^{\text{GWO}}(k) = \frac{\mathbf{X}_j^{(1)}(k) + \mathbf{X}_j^{(2)}(k) + \mathbf{X}_j^{(3)}(k)}{3}.$$

5. Локальное управление (Boids).

Для перехватчика i в радиусе обзора или методом кластеризации K-means ($K = 3$) найти множество соседей $N_i(k)$ и рассчитать силы

$$\mathbf{F}_S = -k_s \sum_{\substack{j \in N_i, \\ r_{ij} < d_{\min}}} \frac{\mathbf{p}_j - \mathbf{p}_i}{r_{ij}^2};$$

$$\mathbf{F}_A = k_a \left(\bar{\mathbf{v}}_{N_i} - \mathbf{v}_i \right),\ \bar{\mathbf{v}}_{N_i} = \frac{1}{N_i} \sum_{j \in N_i} \mathbf{v}_j;$$

$$\mathbf{F}_C = k_c \left(\bar{\mathbf{p}}_{N_i} - \mathbf{p}_i \right),\ \bar{\mathbf{p}}_{N_i} = \frac{1}{N_i} \sum_{j \in N_i} \mathbf{p}_j$$

и локальное ускорение

$$\mathbf{u}_i^{\text{Boids}} = \mathbf{F}_S + \mathbf{F}_A + \mathbf{F}_C.$$

6. Вычислить итоговое ускорение и обновление состояния.

6.1. Вес глобальной компоненты

$$\mathbf{w}_g(k) = \frac{1}{2\sqrt{d}} \mathbf{a}(k).$$

6.2. Итоговое ускорение

$$\mathbf{u}_i(k) = \mathbf{w}_g(k) \times \left(\mathbf{X}_i^{\text{GWO}}(k) - \mathbf{p}_i(k)\right) + \mathbf{u}_i^{\text{Boids}}(k),$$
$$\|\mathbf{u}_i\| \le a_{\max}.$$

6.3. Обновить скорость и положение перехватчика, например по схеме Эйлера:

$$\mathbf{v}_i \leftarrow \mathbf{v}_i + \mathbf{u}_i \Delta t, \|\mathbf{v}_i\| \le V_{\max}, \mathbf{p}_i \leftarrow \mathbf{p}_i + \mathbf{v}_i \Delta t.$$

7. Проверка условий останова.

7.1. Увеличить счетчик $k \leftarrow k + 1$.

7.2. Если достигнут $k_{\max}$ или выполнено условие перехвата (каждая цель $\mathbf{q}_m$ оказалась в радиусе R_{cap} хотя бы одного перехватчика), алгоритм остановить; иначе вернуться к шагу 2.

Результаты симуляционных экспериментов и обсуждение

Рассмотрим результаты серии компьютерных симуляций, проведенных для оценки эффективности гибридного алгоритма GWO + Boids в различных сценариях перехвата.

Численные эксперименты выполнены в среде Python 3.11 с использованием библиотек NumPy 1.26 и Matplotlib 3.8. Вычисления проводились на ноутбуке с процессором Intel Core i7-11800H и 16 Гбайт оперативной памяти.

Каждый сценарий моделировался 100 раз. Параметры математической модели для БПЛА соответствуют нормативу Международной организации гражданской авиации (ICAO) [22]: $V_{\max} = 10$ м/с; $a_{\max} = 2$ м/с^2; $\Delta t = 1$ с; $R_{cap} = 5$ м; $d_{\min} = 5$ м. Параметры для алгоритма GWO + + Boids: $k_s = k_a = k_c = 0, 1$; $k_{\max} = 30$.

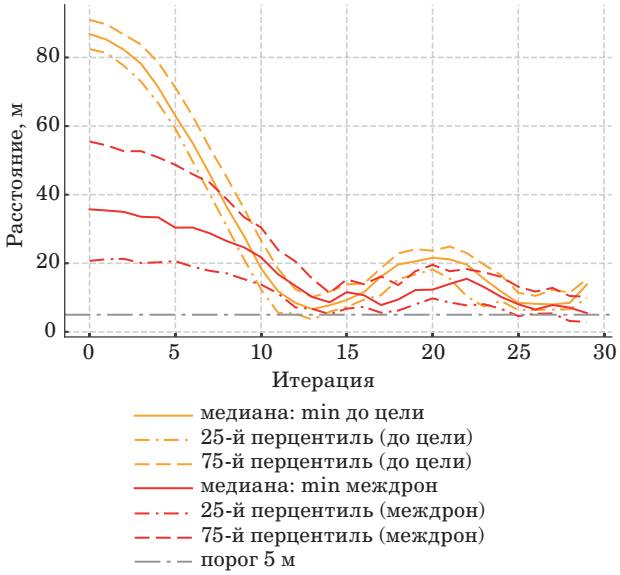
Численные проверки проводились для роя из $N = 4$ дронов на двух простых сценариях: стационарной и равномерно движущейся цели. Все эксперименты выполнялись в плоскости ($d = 2$), что позволило изолированно оценить влияние алгоритма без усложняющих факторов. Эти базовые случаи необходимы для перехода к более сложным испытаниям.

Сценарий № 1 «Одна стационарная цель». Цель фиксирована и не реагирует на действия роя, что является строгим тестом для глобального механизма. Перехватчики должны сократить дистанцию с разных направлений без помощи встречного движения. Сценарий проверяет эффективность GWO в сближении и способность правил Boids формировать устойчивое кольцо окружения.

Медианная минимальная дистанция «перехватчик — цель» убывает монотонно при узком межквартильном размахе (InterQuartile Range, IQR), что указывает на стабильность динамики (рис. 1). Медианная междроновая дистанция также уменьшается при уплотнении построения, но остается выше порогового значения 5 м, подтверждая корректность правил Boids и отсутствие коллизий.

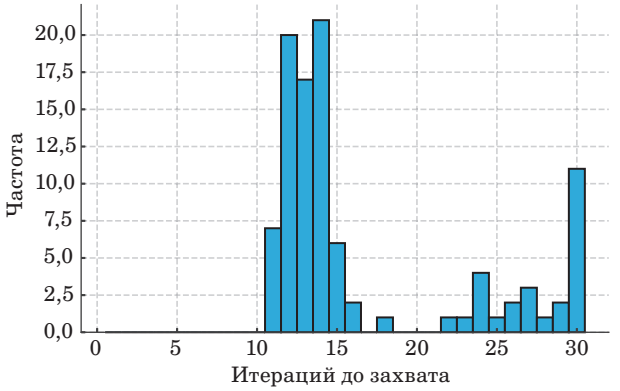
Распределение времени захвата для 100 испытаний (рис. 2) имеет компактный основной кластер в диапазоне 11–16 итераций и удлиненный хвост, соответствующий редким случаям неудачной начальной геометрии, когда рою требуется дополнительное время на перестроение. Медиана составляет около 14 итераций.

В сценарии № 1 наблюдается быстрая и стабильная сходимость к стационарной цели. Панель траекторий (рис. 3) иллюстрирует переход от разреженного построения к организован-



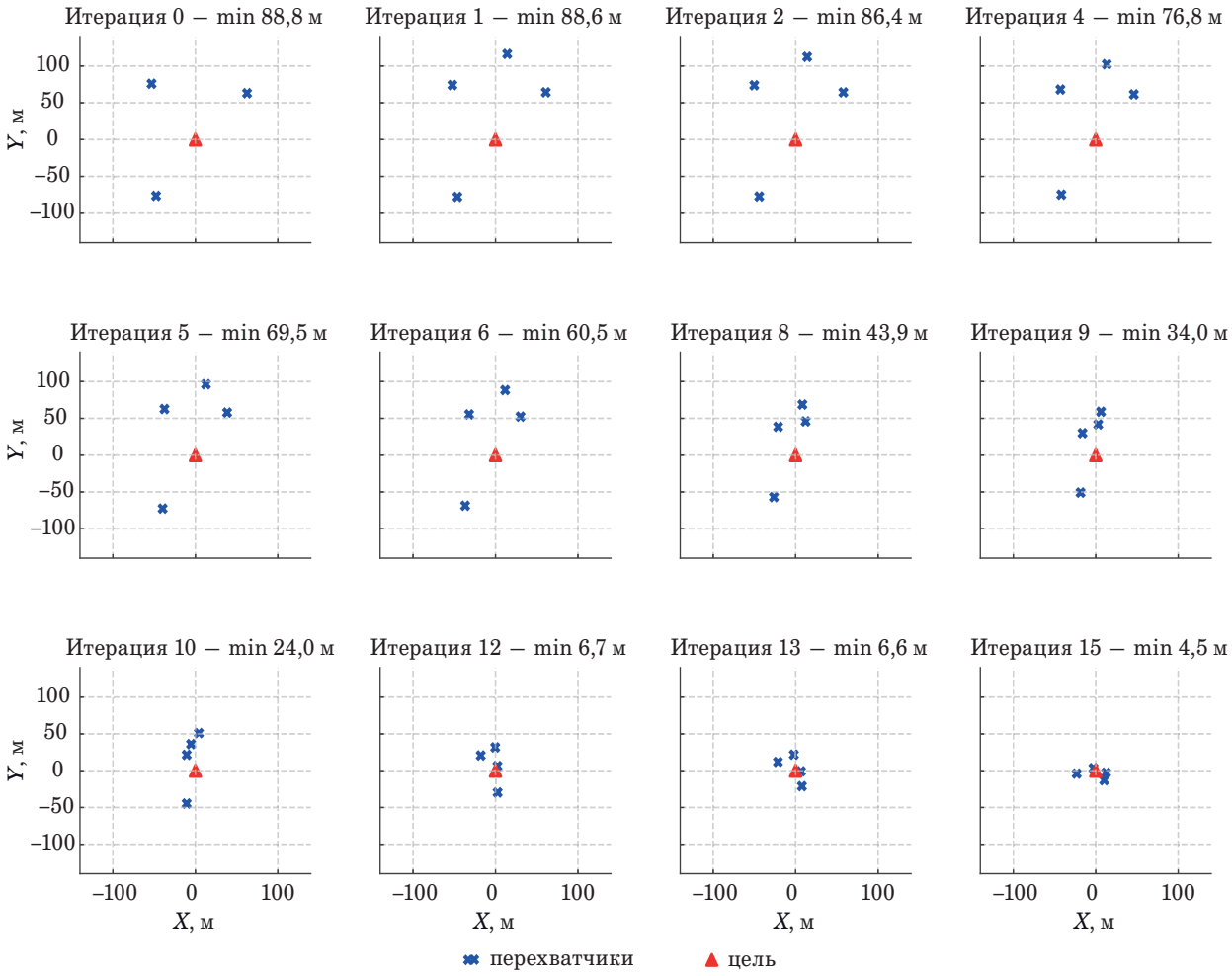
■ ***Рис. 1.*** Сводные кривые для сценария № 1

■ ***Fig. 1.*** Summary curves for scenario No. 1



■ ***Рис. 2.*** Распределение времени захвата для сценария № 1

■ ***Fig. 2.*** Capture time distribution for scenario No. 1

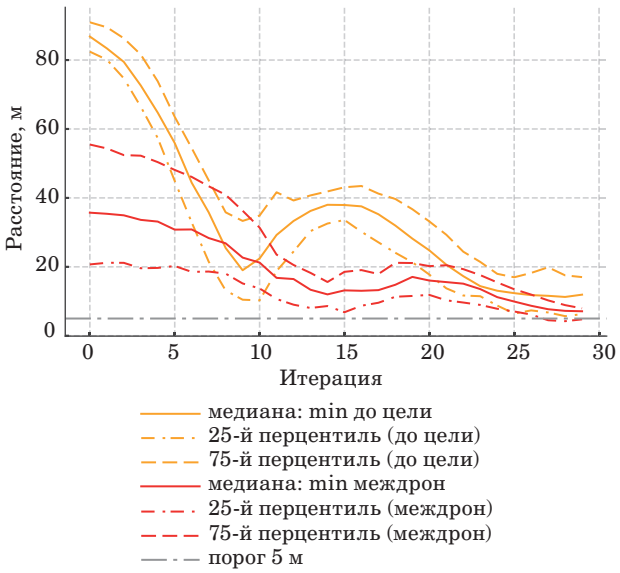


■ ***Рис. 3.*** Позиции перехватчиков и цели по итерациям для сценария № 1
■ ***Fig. 3.*** Iterative positions of interceptors and target for scenario No. 1

ному стягиванию вокруг цели при сохранении безопасных интервалов.

Сценарий № 2 «Одна цель с постоянной скоростью». Цель движется прямолинейно и равномерно со скоростью $V_{red} = 10$ м/с. Постановка динамическая: траектория предсказуема, а отклонения перехватчиков от оптимального курса связаны лишь со стохастикой алгоритма и переназначением лидеров. Сценарий введен для проверки адаптивности схемы, т. е. способности быстро обновлять лидеров и сохранять безопасность при движении цели.

В сценарии № 2 медиана минимальной дистанции до цели сначала быстро убывает (рис. 4), так как при большом абсолютном значении коэффициента сходимости $\mathbf{a}(k)$ доминирует глобальная компонента $\mathbf{w}_g$ метода GWO, направляющая рой к цели. На средних итерациях наблюдается локальный подъем медианы. Во-первых, в расчет начинают входить более сложные траектории. Во-вторых, сказывается динамика цели и



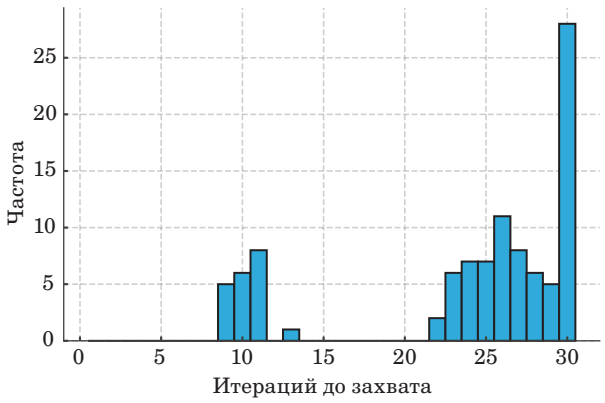
■ ***Рис. 4.*** Сводные кривые для сценария № 2
■ ***Fig. 4.*** Summary curves for scenario No. 2

случайность коэффициент-векторов $\mathbf{A}_j$, $\mathbf{C}_j$, вызывающих временные отклонения от порогового значения. На поздних шагах медиана вновь снижается к порогу 5 м, что показывает успешное завершение даже сложных эпизодов.

Медиана минимальной междроновой дистанции стартует на больших значениях и по мере стягивания роя плавно снижается, но остается выше 5 м как по центру, так и для большинства IQR. Это подтверждает соблюдение безопасных интервалов благодаря правилам Boids. Одновременно IQR сужается, отражая более однотипное поведение при сопровождении цели. По сути, рой надежно сближается с целью без нарушений безопасности, а различия начальных геометрий выражаются в ширине IQR и бимодальной структуре сходимости, что соответствует распределению времени захвата.

На гистограмме времени захвата для сценария № 2 (рис. 5) заметна мультимодальная структура. Первая мода (9–12 итераций) связана с удачной инициализацией, когда перехватчик стартует ближе к цели и GWO быстро приводит его в радиус 5 м. Вторая мода (25–30 итераций) соответствует менее выгодным начальным условиям, когда требуется длительное сокращение дистанции при ослаблении глобальной компоненты и росте влияния правил Boids. Значительная доля таких случаев объясняет повышенное медианное время захвата. Четвертый перехватчик увеличивает шанс «быстрых» исходов, но при неблагоприятной конфигурации сближение требует существенно больше итераций.

Типичный прогон с медианным временем захвата среди 100 испытаний показан на рис. 6. На старте рой сокращает дистанцию от 90–100 м до нескольких десятков метров за счет высокого веса (при больших абсолютных значениях $\mathbf{a}(k)$) глобальной компоненты $\mathbf{w}_g$ метода GWO, интенсивно тянущей перехватчиков к цели. По мере уменьшения абсолютного значения коэффициента $\mathbf{a}(k)$ его вклад снижается и возрастает роль правил Boids. Выравниваются скорости, стабилизируется строй, предотвращается избыточное сжатие. В финале минимальная междроновая дистанция остается выше 5 м, фиксируя успешный захват. Характерно, что перед захватом «синие» дроны движутся почти в одном направлении, что отражает эффект выравнивания (Alignment), обеспечивающий устойчивое сопровождение цели.



■ ***Рис. 5.*** Распределение времени захвата для сценария № 2

■ ***Fig. 5.*** Capture time distribution for scenario No. 2

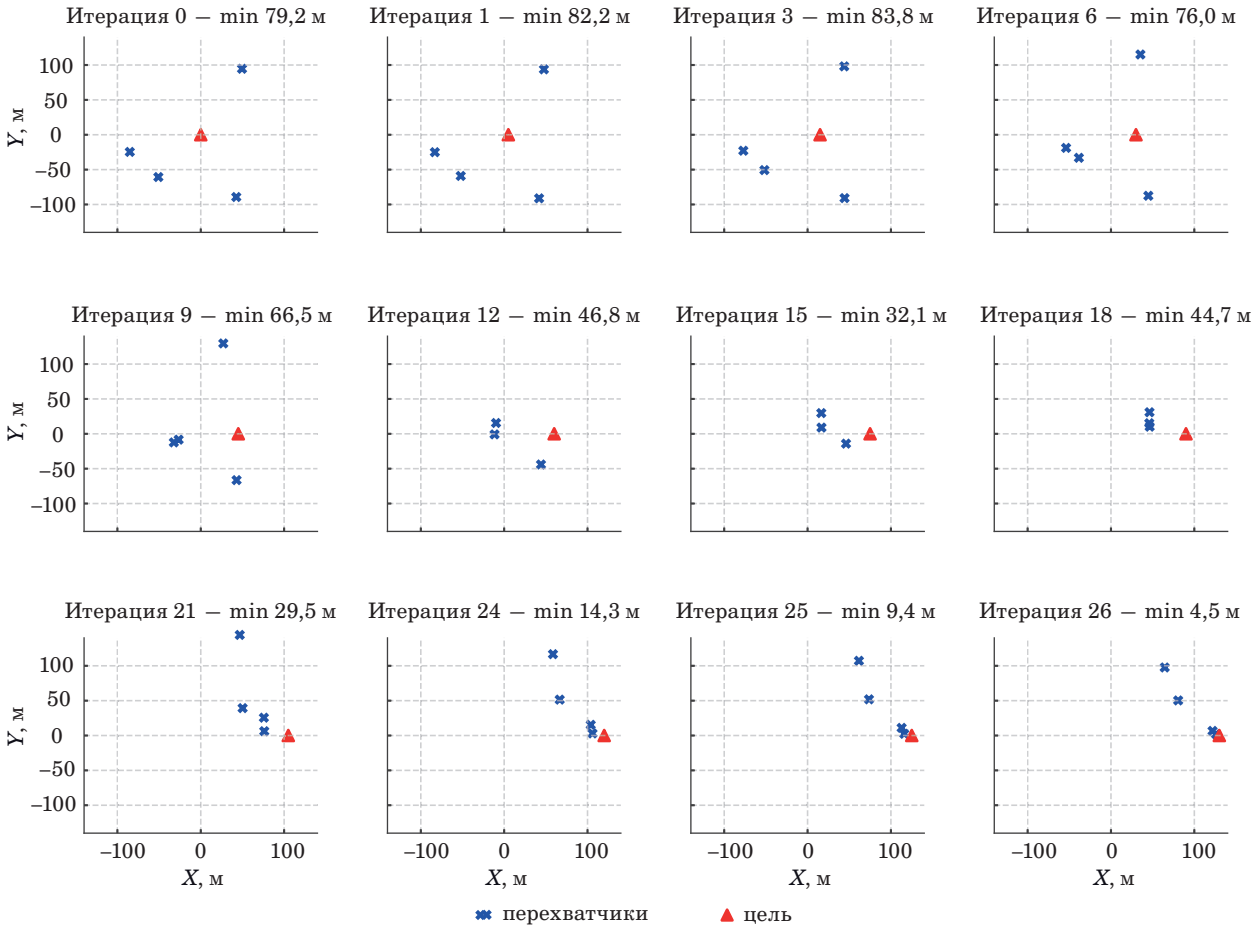
В целом, при случайной инициализации четверка перехватчиков надежно «закрывает» цель, а слой Boids подтверждает соблюдение безопасных интервалов. Увеличение роя до 5–8 аппаратов почти не влияет на время перехвата: численность лишь утолщает кольцо и уменьшает минимальный зазор (до 11–16 м), но быстроту определяет тройка лидеров α, β, δ. Практически рой из четырех дронов обеспечивает «страховку». При удачной геометрии один из них быстро перехватывает цель, но в среднем требуется больше итераций. При этом минимальные дистанции остаются выше 5 м, что подтверждает корректную работу Boids. На финальной фазе кольцо стягивается, сохраняя допустимые интервалы.

Для количественной оценки вклада гибридизации методов GWO+Boids проведем сравнительные испытания при тех же условиях сценариев № 1 и 2 для 100 испытаний (таблица). Применим два базовых алгоритма:

1) классический алгоритм GWO [4] — популяционный стохастический оптимизатор управляющих векторов перехватчиков, минимизирующий глобальную метрику сближения с целью без локальных правил безопасности;

2) прямолинейное наведение перехватчиков с упреждением — движение перехватчиков по лучу к прогнозной точке встречи цели при постоянной скорости или для стационарной цели (в этом случае, очевидно, прогнозная точка совпадает с положением цели).

По медианному времени базовые алгоритмы сопоставимы с гибридным (колебания в пределах IQR объясняются случайностью коэффициент-векторов $\mathbf{A}_j$, $\mathbf{C}_j$ и стартовой геометрией). Ключевым преимуществом GWO+Boids является безопасность, так как при всех 100 запусках коллизии не наблюдаются, в то время как в GWO и при прямолинейном наведении фиксируется существенное число столкновений. Причина носит структурный характер: GWO+Boids реализует режим «охват+сближение», формируя кольцевой фронт с управляемыми зазорами (разделение/выравнивание/сцепление), тогда как при равномерном наведении траектории сходятся к общей точке, повышая вероятность опасных сближений. В прикладных условиях реального



■ ***Рис. 6.*** Позиции перехватчиков и цели по итерациям для сценария № 2
■ ***Fig. 6.*** Iterative positions of interceptors and target for scenario No. 2

времени гибридный алгоритм снижает потребность в подключении внешних модулей предотвращения столкновений и уменьшает число гиперпараметров при сохранении сопоставимой скорости перехвата.

Испытания показали, что комбинация GWO (глобальное распределение) и Boids (локальная координация) обеспечивает эффективный, безопасный и адаптивный перехват одиночных и множественных целей. Применение роя вместо одного быстрого дрона оправдано. Кольцо блокирует маневры, повышает отказоустойчивость, позволяет работать с несколькими целями и поддерживает давление после контакта.

Гибридная схема разделяет функции планировщика: GWO распределяет дроны по перспективным направлениям, а Boids удерживает строй без диспетчера. Во всех 100 прогонах каждого сценария алгоритм сходился, нарушений безопасности не зафиксировано. Тесты подтвердили вклад подсистем и заложили основу для дальнейших испытаний с множественными целями и препятствиями.

■ Сравнительный анализ различных алгоритмов
■ Comparative analysis of various algorithms

Алгоритм	Доля перехватов, %	Медианное время, итерации	IQR	Частота столкновений (на 100 испытаний)
Сценарий № 1				
GWO+Boids	100	14	12–16	0
GWO	100	13	11–17	18
Прямолинейное наведение	99	13	11–18	27
Сценарий № 2				
GWO+Boids	100	35	28–42	0
GWO	97	33	26–41	23
Прямолинейное наведение	95	32	24–45	31

Заключение

Работа показала, что децентрализованная гибридная схема на основе иерархии лидеров GWO и локальных правил Boids позволяет одновременно достигать быстрого глобального сближения и соблюдения междроновых дистанций без диспетчера и при умеренной сложности. Модель использует дискретную кинематику с ограничениями по скорости и монотонно убывающими компонентами коэффициента «охоты» **a**, которые обеспечивают переход от разведки к окружению. В многокритериальных задачах применяется кластеризация целей и динамическое переназначение ролей α, β, δ.

Получено достаточное условие коллизионной безопасности через настройку коэффициента разделения. Медианное время захвата составило 10–15 итераций для неподвижной цели и 30–40 для движущейся при сохранении междроновой дистанции не менее 5 м. Дополнительные дроны (сверх четырех) не влияют на время перехвата цели, лишь повышают устойчивость кольца, что указывает на рациональные пределы численности дронов в рое.

Исследование ограничено плоской кинематикой без препятствий и простейшими целями. Ближайшие направления развития исследований включают переход к трехмерной динамике и ограниченным каналам связи, учет препятствий и сложных целей, адаптивное изменение параметра сходимости **a** и локальных коэффициентов Boids, а также анализ многоцелевых конфигураций с переменной топологией кластеров и критериями миссии. Такая программа исследований позволит распространить полученные гарантии и характеристики сходимости на более реалистичные сценарии и уточнить границы применимости алгоритма для задач кооперативного перехвата в сложной среде.

Литература

1. **Meng Q., Qu Q., Chen K., Yi T.** Multi UAV path planning based on cooperative co-evolutionary algorithms with adaptive decision variable selection. *Drones*, 2024, vol. 8, art. 435. doi:10.3390/drones8090435
2. **Горшков И. Ф., Акимов А. А.** Динамическая маршрутизация дронов для доставки заказов. *Научно-технический вестник Поволжья*, 2025, № 6, с. 202–205. EDN: KOXOXI
3. **Крестовников К. Д.** Математическая модель и алгоритмы управления группой наземных роботов с перераспределением энергетических ресурсов. *Информационно-управляющие системы*, 2023, № 6, с. 20–34. doi:10.31799/1684-8853-2023-6-20-34, EDN: QFJGNR
4. **Mirjalili S., Mirjalili S. M., Lewis A.** Grey Wolf Optimizer. *Advances in Engineering Software*, 2014, vol. 69, pp. 46–61. doi:10.1016/j.advengsoft.2013.12.007
5. **Ходашинский И. А.** Методы повышения эффективности роевых алгоритмов оптимизации. *Автоматика и телемеханика*, 2021, № 6, с. 3–45. doi:10.31857/S0005231021060015, EDN: WKONWW
6. **Адонин Л. С., Владыко А. Г.** Алгоритмы роевого интеллекта для решения задач оптимизации в системах телекоммуникаций. *Труды учебных заведений связи*, 2025, т. 11, № 3, с. 7–24. doi:10.31854/1813-324X-2025-11-3-7-24, EDN: JUAAMB
7. **Akimov A. A., Sapozhnikova K. A., Gnatenko Y. A.** A discrete swarm optimization modification for the multi agent traveling salesman problem. *2025 International Russian Smart Industry Conference (SmartIndustryCon)*, Sochi, Russian Federation, 2025, pp. 418–424. doi:10.1109/SmartIndustryCon65166.2025.10986083
8. **Phung M. D., Ha Q. P.** Safety enhanced UAV path planning with spherical vector based particle swarm optimization. *Applied Soft Computing*, 2021, vol. 107, art. 107376. doi:10.1016/j.asoc.2021.107376
9. **Tang J., Liu G., Pan Q. T.** A Review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA Journal of Automatica Sinica*, 2021, vol. 8, no. 10, pp. 1627–1643. doi:10.1109/JAS.2021.1004129
10. **Kaya E., Görkemli B., Akay B., Karaboga D.** A Review on the studies employing Artificial Bee Colony algorithm to solve combinatorial optimization problems. *Engineering Applications of Artificial Intelligence*, 2022, vol. 115, art. 105311. doi10.1016/j.engappai.2022.105311
11. **Shehab M., Abu Hashem M. A., Shambour M. K. Y., Alsalibi A. I., Alomari O. A., Gupta J. N. D., Alsoud A. R., Abualigah L., Abuhaija B.** A comprehensive Review of Bat Inspired algorithm: Variants, applications, and hybridization. *Archives of Computational Methods in Engineering*, 2023, vol. 30, pp. 765–797. doi:10.1007/s11831-022-09817-5
12. **Дивеев А. И., Константинов С. В.** Задача оптимального управления и ее решение эволюционным алгоритмом «серого волка». *Вестник Российского университета дружбы народов. Серия: Инженерные исследования*, 2018, т. 19, № 1, с. 67–79. doi:10.22363/2312-8143-2018-19-1-67-79, EDN: XPUJXN
13. **Han D., Yu Q., Jiang H.** Three dimensional path planning for post disaster rescue UAV by integrating improved Grey Wolf Optimizer and Artificial Potential Field method. *Applied Sciences*, 2024, vol. 14, art. 4461. doi:10.3390/app14114461
14. **Cao Y., Mohamad Nor N.** An improved dynamic window approach algorithm for dynamic obstacle avoidance in mobile robot formation. *Decision Analytics Journal*, 2024, vol. 11, art. 100471. doi:10.1016/j.dajour.2024.100471

15. **Zhou T., Chen M., Wang Y., Zhu R., Yang C.** Cooperative attack defense decision making of multi UAV using satisficing decision enhanced Wolf Pack search algorithm. *Preprint*, 2021. doi:10.21203/rs.3.rs-1116258
16. **Zhou X., Shi G., Zhang J.** Improved Grey Wolf algorithm: A method for UAV path planning. *Drones*, 2024, vol. 8, no. 11, art. 675. doi:10.3390/drones8110675
17. **Shi J., Tan L., Zhang H., Lian X., Xu T.** Adaptive multi UAV path planning method based on improved Gray Wolf Optimizer (AP GWO). *Computers & Electrical Engineering*, 2022, vol. 104, art. 108377. doi:10.1016/j.compeleceng.2022.108377
18. **Sun Y., Lv B., Yang H., Li X.** Multi UAV trajectory planning based on improved multi population Grey Wolf Optimizer. *Proceedings of the 36th Chinese Control and Decision Conference (CCDC)*, IEEE, 2024, pp. 6142–6148. doi:10.1109/CCDC62350.2024.10587624
19. **Reynolds C. W.** Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 1987, vol. 21, no. 4, pp. 25–34. doi:10.1145/37402.37406
20. **Rao C., Wang Z., Shao P.** A multi strategy collaborative Grey Wolf Optimization algorithm for UAV path planning. *Electronics*, 2024, vol. 13, no. 13, art. 2532. doi:10.3390/electronics13132532
21. **Zhao D., Cai G., Wang Y., Li X.** Path planning of obstacle crossing robot based on golden sine Grey Wolf Optimizer. *Applied Sciences*, 2024, vol. 14, art. 1129. doi:10.3390/app14031129
22. **Riedel M. A.** Review of detect and avoid standards for unmanned aircraft systems. *Aerospace*, 2025, vol. 12, no. 4, art. 344. doi:10.3390/aerospace12040344

UDC 004.896
doi:10.31799/1684-8853-2025-6-2-14
EDN: DECQWI

Hybrid algorithm of global planning and local interaction for target interception by UAV swarms

A. A. Akimov[a], PhD, Phys.-Math., Associate Professor, orcid.org/0000-0003-3387-2959
Y. A. Gnatenko[b], PhD, Phys.-Math., Associate Professor, orcid.org/0009-0009-9264-3989, y.a.gnatenko@struust.ru
R. G. Bolbakov[a], PhD, Tech., Associate Professor, orcid.org/0000-0002-4922-7260
[a]MIREA — Russian Technological University, 78, Vernadsky Av., 119454, Moscow, Russian Federation
[b]Branch of the Ufa University of Science and Technology, 49, Lenin Av., 453103, Sterlitamak, Russian Federation

Introduction: Target interception by UAV swarms requires both rapid convergence to the target and strict maintenance of inter-UAV distances without centralized control. Global optimization algorithms provide efficient convergence but do not always account for local safety constraints, while behavioral rules maintain spacing but have limited applicability in complex scenarios. **Purpose:** To develop a decentralized swarm control algorithm that shortens interception time while preventing collisions among agents. **Methods:** The proposed approach integrates the Grey Wolf Optimizer with the Boids model. The global Grey Wolf Optimizer module directs drones toward targets, while the Boids rules regulate relative motion and prevent spacing violations. **Results:** We develop a hybrid algorithm that ensures reliable interception in scenarios with both stationary and moving targets. The median capture time is 10–15 iterations for a stationary target and 30–40 iterations for a moving one. In all experiments, the minimum inter-UAV distance has remained above the safety threshold of 5 m. A safety condition has been established through the separation coefficient, demonstrating the absence of collisions during convergence. To handle multiple targets, we apply clustering, which allows splitting the swarm into subgroups and coordinating interception even when targets are spatially separated. **Practical relevance:** The algorithm requires few parameters, scales with the swarm size, and is suitable for real-time onboard implementation. **Discussion:** The results indicate that the hybrid algorithm successfully combines the advantages of global optimization and local coordination, providing a balance between interception speed and safety. This forms a basis for future research on three-dimensional dynamics and scenarios with complex obstacles.

Keywords — UAV swarm, Grey Wolf Optimizer, Boids, target interception, decentralized control, collision avoidance, metaheuristics.

For citation: Akimov A. A., Gnatenko Y. A., Bolbakov R. G. Hybrid algorithm of global planning and local interaction for target interception by UAV swarms. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 2–14 (In Russian). doi:10.31799/1684-8853-2025-6-2-14, EDN: DECQWI

References

1. Meng Q., Qu Q., Chen K., Yi T. Multi UAV path planning based on cooperative co-evolutionary algorithms with adaptive decision variable selection. *Drones*, 2024, vol. 8, art. 435. doi:10.3390/drones8090435
2. Gorshkov I. F., Akimov A. A. Dynamic routing of drones for order delivery. *Scientific and Technical Volga Region Bulletin*, 2025, no. 6, pp. 202–205 (In Russian). EDN: KOXOXI
3. Krestovnikov K. D. Mathematical model and control algorithms for a group of ground robots with energy resource redistribution. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2023, no. 6, pp. 20–34 (In Russian). doi:10.31799/1684-8853-2023-6-20-34, EDN: QFJGNR
4. Mirjalili S., Mirjalili S. M., Lewis A. Grey Wolf Optimizer. *Advances in Engineering Software*, 2014, vol. 69, pp. 46–61. doi:10.1016/j.advengsoft.2013.12.007
5. Hodashinsky I. A. Methods for improving the efficiency of swarm optimization algorithms. A survey. *Automation and Remote Control*, 2021, vol. 82, no. 6, pp. 935–967. doi:10.1134/S0005117921060011, EDN: YBFNCY
6. Adonin L. S., Vladyko A. G. Swarm intelligence algorithms for solving optimization problems in telecommunication systems. *Proceedings of Telecommunication Universities*, 2025, vol. 11, no. 3, pp. 7–24 (In Russian). doi:10.31854/1813-324X-2025-11-3-7-24, EDN: JUAAMB
7. Akimov A. A., Sapozhnikova K. A., Gnatenko Y. A. A discrete swarm optimization modification for the multi agent traveling salesman problem. *2025 International Russian Smart Industry Conference (SmartIndustryCon)*, Sochi, Russian Federation, 2025, pp. 418–424. doi:10.1109/SmartIndustryCon65166.2025.10986083

8. Phung M. D., Ha Q. P. Safety enhanced UAV path planning with spherical vector based particle swarm optimization. *Applied Soft Computing*, 2021, vol. 107, art. 107376. doi:10.1016/j.asoc.2021.107376
9. Tang J., Liu G., Pan Q. T. A Review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. *IEEE/CAA Journal of Automatica Sinica*, 2021, vol. 8, no. 10, pp. 1627–1643. doi:10.1109/JAS.2021.1004129
10. Kaya E., Görkemli B., Akay B., Karaboga D. A Review on the studies employing Artificial Bee Colony algorithm to solve combinatorial optimization problems. *Engineering Applications of Artificial Intelligence*, 2022. vol. 115, art. 105311. doi:10.1016/j.engappai.2022.105311
11. Shehab M., Abu Hashem M. A., Shambour M. K. Y., Alsalibi A. I., Alomari O. A., Gupta J. N. D., Alsoud A. R., Abualigah L., Abuhaija B. A comprehensive Review of Bat Inspired algorithm: Variants, applications, and hybridization. *Archives of Computational Methods in Engineering*, 2023, vol. 30, pp. 765–797. doi:10.1007/s11831-022-09817-5
12. Diveev A. I., Konstantinov S. V. Optimal control problem and its solution by Grey Wolf Optimizer algorithm. *RUDN Journal of Engineering Researches*, 2018, vol. 19, no. 1, pp. 67–79 (In Russian). doi:10.22363/2312-8143-2018-19-1-67-79, EDN: XPUJXN
13. Han D., Yu Q., Jiang H. Three dimensional path planning for post disaster rescue UAV by integrating improved Grey Wolf Optimizer and Artificial Potential Field method. *Applied Sciences*, 2024, vol. 14, art. 4461. doi:10.3390/app14114461
14. Cao Y., Mohamad Nor N. An improved dynamic window approach algorithm for dynamic obstacle avoidance in mobile robot formation. *Decision Analytics Journal*, 2024, vol. 11, art. 100471. doi:10.1016/j.dajour.2024.100471
15. Zhou T., Chen M., Wang Y., Zhu R., Yang C. Cooperative attack defense decision making of multi UAV using satisficing decision enhanced Wolf Pack search algorithm. *Preprint*, 2021. doi:10.21203/rs.3.rs-1116258
16. Zhou X., Shi G., Zhang J. Improved Grey Wolf algorithm: A method for UAV path planning. *Drones*, 2024, vol. 8, no. 11, art. 675. doi:10.3390/drones8110675
17. Shi J., Tan L., Zhang H., Lian X., Xu T. Adaptive multi UAV path planning method based on improved Gray Wolf Optimizer (AP GWO). *Computers & Electrical Engineering*, 2022, vol. 104, art. 108377. doi:10.1016/j.compeleceng.2022.108377
18. Sun Y., Lv B., Yang H., Li X. Multi UAV trajectory planning based on improved multi population Grey Wolf Optimizer. *Proceedings of the 36th Chinese Control and Decision Conference (CCDC)*, IEEE, 2024, pp. 6142–6148. doi:10.1109/CCDC62350.2024.10587624
19. Reynolds C. W. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 1987, vol. 21, no. 4, pp. 25–34. doi:10.1145/37402.37406
20. Rao C., Wang Z., Shao P. A multi strategy collaborative Grey Wolf Optimization algorithm for UAV path planning. *Electronics*, 2024, vol. 13, no. 13, art. 2532. doi:10.3390/electronics13132532
21. Zhao D., Cai G., Wang Y., Li X. Path planning of obstacle crossing robot based on Golden Sine Grey Wolf Optimizer. *Applied Sciences*, 2024, vol. 14, art. 1129. doi:10.3390/app14031129
22. Riedel M. A. Review of detect and avoid standards for unmanned aircraft systems. *Aerospace*, 2025, vol. 12, no. 4, art. 344. doi:10.3390/aerospace12040344

УДК 004.6
doi:10.31799/1684-8853-2025-6-15-27
EDN: ERTCQY

Научные статьи
Articles

Метод фильтрации признаков по критериям стабильности и значимости

О. С. Исаева[a]***,*** *доктор техн. наук, ведущий научный сотрудник, orcid.org/0000-0002-5061-6765, isaeva@icm.krasn.ru*
[a]*Институт вычислительного моделирования СО РАН — обособленное подразделение ФИЦ КНЦ СО РАН, Академгородок, 50/44, Красноярск, 660036, РФ*

Введение: *анализ сетевого трафика интернета вещей осложнен высокой размерностью, избыточностью и нестабильностью признаков. Наблюдается сильная корреляция, мультиколлинеарность и шум, что снижает качество кластеризации и затрудняет интерпретацию. Кроме того, легитимный и аномальный трафик часто перекрываются, что осложняет формализацию границ между классами. В этой связи требуется метод отбора признаков, обеспечивающий устойчивость, компактность и семантическую интерпретируемость.* ***Цель:*** *разработать и экспериментально оценить новый метод для построения устойчивого и интерпретируемого признакового пространства в задачах кластеризации сетевого трафика — Progressive Feature Filtering with Stability and Significance (PFF-SS, PF^{2}S).* ***Методы:*** *описан пошаговый алгоритм PF^{2}S, сочетающий анализ линейных (корреляция, VIF) и нелинейных (взаимная информация) зависимостей с оценкой стабильности и информативности. На каждом этапе исключаются избыточные, слабо значимые или нестабильные признаки.* ***Результаты:*** *применение PF^{2}S к датасету сетевого трафика интернета вещей позволило сократить число признаков с более чем 300 до 17, сохранив высокую информативность. Сравнение с пространствами, редуцированными методом главных компонент и методом рекурсивного исключения признаков, показало, что PF^{2}S обеспечивает более высокие метрики стабильности, интерпретируемости и качестве кластеризации. Метод не преобразует признаки, как метод главных компонент, а сохраняет их исходную семантику. По сравнению с методом рекурсивного исключения признаков PF^{2}S обеспечил отсутствие мультиколлинеарности, более низкую сложность модели и на 17,6 % более высокий силуэтный коэффициент. Кластеры, построенные на основе PF^{2}S-пространства, оказались устойчивыми (высокий скорректированный индекс Рэнда) и семантически интерпретируемыми.* ***Практическая значимость:*** *PF^{2}S формирует компактное и устойчивое признаковое пространство, пригодное для систем обнаружения аномалий в сетевом трафике интернета вещей.* ***Обсуждение:*** *перспективным направлением является адаптация PF^{2}S для потоковой обработки данных и интеграция с сигнатурными методами выявления аномалий и онтологиями сетевого трафика.*

Ключевые слова *— интернет вещей, устойчивость признаков, информативность признаков, кластеризация K-средних, агломеративная кластеризация, спектральная кластеризация, модель гауссовых смесей, метод главных компонент, метод рекурсивного исключения признаков, анализ сетевого трафика, обнаружение аномалий.*

Для цитирования: Исаева О. С. Метод фильтрации признаков по критериям стабильности и значимости. *Информационно-управляющие системы*, 2025, № 6, с. 15–27. doi:10.31799/1684-8853-2025-6-15-27, EDN: ERTCQY
For citation: Isaeva O. S**.** Feature filtering method based on stability and significance criteria. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 15–27 (In Russian). doi:10.31799/1684-8853-2025-6-15-27, EDN: ERTCQY

Введение

Широкое применение систем искусственного интеллекта сопровождается ростом числа задач, в которых признаковое пространство характеризуется высокой размерностью, разнотипностью и избыточностью. Это особенно актуально в предметных областях, где данные генерируются в реальном времени и содержат сотни, а иногда и тысячи признаков, описывающих поведение устройств, пользователей и правила их взаимодействия [1]. В таких условиях эффективность моделей машинного обучения снижается из-за увеличения вычислительной сложности, риска переобучения и потери интерпретируемости. Сокращение признакового пространства становится ключевым условием построения устойчивых, интерпретируемых и масштабируемых моделей.

К таким задачам относится анализ сетевой активности в системах интернета вещей (Internet of Things, IoT), где применение традиционных протоколов безопасности затруднено требованиями к облегченности и энергоэффективности решений из-за ограничений в вычислительной мощности и времени автономной работы устройств [2]. Архитектура IoT объединяет физические и виртуальные объекты: датчики, исполнительные механизмы, облачные сервисы, специальные сетевые протоколы, транспортные средства коммуникации и пользователей [3]. Основными целями безопасности IoT являются обеспечение конфиденциальности, целостности и доступности предлагаемых услуг [4]. Аномалии в таких сетях могут быть вызваны как техническими сбоями (например, выходом из строя IoT-устройства), так и целенаправленными кибератаками (например, DoS — «отказ в обслуживании», MITM — «атака посредника», spoofing — «подмена доверенного лица»). Эффективное выявление таких угроз требует

построения моделей, способных различать нормальное поведение и деструктивные воздействия на основе анализа многомерных данных [5]. IoT-сети характеризуются динамичностью процессов, гетерогенностью устройств и постоянным изменением признаков нормального поведения, что затрудняет формализацию границ между «нормой» и «аномалией» [6].

В рамках исследований по обеспечению безопасности сети IoT в Красноярском научном центре СО РАН создана и внедрена инфраструктура для сбора данных и имитации угроз, позволяющая генерировать реалистичные сценарии сетевой активности [7]. Схема IoT-сети построена по шаблону «Издатель — Подписчик» с использованием протокола MQTT. Различные сценарии атак для этого протокола рассмотрены в работе [8]. Обязательным элементом архитектуры сети является брокер, отвечающий за прием и маршрутизацию сообщений. Исследования проводятся для брокеров, развернутых в нескольких популярных платформах (Eclipse Mosquitto, EMQX, NanoMQ, VerneMQ) с различными конфигурациями политик безопасности. Настройки брокеров осуществляются адаптивно [9], но вопросы безопасности для такой разнообразной сети остаются. С помощью программных агентов фиксируется весь сетевой трафик, поступающий на стандартные и шифрованные порты как во внутренней сети, так и извне [10]. В настоящее время собраны датасеты, описывающие временные, статистические, протокольные и поведенческие характеристики трафика.

Полученные датасеты характеризуются высокой размерностью как по числу объектов (пакетов, накопленных за длительный период), так и по числу признаков (которых более 300). При этом легитимные и аномальные сетевые сессии могут быть как долгосрочными, так и кратковременными, что затрудняет их исследование на основе временных характеристик. Наблюдается высокий уровень шума, многие признаки являются избыточными, сильно коррелированными и чувствительными к вариативности данных. Особую сложность представляет перекрытие классов на подмножествах признаков. В работе [11] показано, что на качество классификации значительно влияет степень такого пересечения. В этих условиях возникает необходимость в применении надежных и интерпретируемых методов снижения размерности, способных выделить устойчивое ядро информативных признаков.

Существующие методы снижения размерности можно разделить на фильтрующие, встраиваемые, обертывающие и методы преобразования [12]. Несмотря на их различия, большинство подходов не обеспечивают высокой стабильности отбора признаков. В работе [13] показано, что в реальных задачах эффективность сокращения пространства признаков целесообразно оценивать по критерию его устойчивости (стабильности), т. е. способности метода воспроизводить схожие наборы признаков при вариации обучающих выборок. Это свойство особенно важно в условиях IoT, где нормальное поведение динамично, а данные подвержены изменчивости из-за обновлений устройств, сбоев или изменений в сетевой нагрузке.

Фильтрующие методы (например, на основе корреляции, взаимной информации или условной энтропии) просты и вычислительно эффективны, что делает их популярными для предварительного анализа в задачах с большим числом признаков [14]. Однако фильтрующие методы не учитывают взаимодействия между признаками и не зависят от целевой модели. Это может привести к отбору избыточных признаков (например, нескольких коррелирующих переменных) или пропуску информативных комбинаций.

Встраиваемые методы интегрируют отбор признаков в процесс обучения, что позволяет учитывать скрытую структуру данных. К таким методам относится LASSO (Least Absolute Shrinkage and Selection Operator), который применяет L_1-регуляризацию для обнуления коэффициентов при слабых признаках, деревья решений и градиентный бустинг (XGBoost, LightGBM), где признаки ранжируются по важности [15]. Эти методы эффективно учитывают нелинейности и взаимодействия, но в условиях шумных и разреженных данных, типичных для сетевых сессий интернета вещей, даже небольшие изменения выборки приводят к существенно разным результатам, что снижает доверие к модели.

Методы преобразования признаков, такие как метод анализа главных компонент (Principal Component Analysis, PCA), t-SNE, UMAP и autoencoders [16], подразделяются на глобальные и локальные в зависимости от того, какие структурные свойства данных они стремятся сохранить. Глобальные методы ориентированы на сохранение расстояний между всеми парами точек в пространстве, в то время как локальные методы фокусируются на сохранении структуры локальных окрестностей. В результате локальные подходы могут лучше передавать внутреннюю структуру кластеров, но при этом искажать глобальную топологию данных; напротив, глобальные методы воспроизводят общую структуру распределения, но могут терять детали локальной кластеризации. Современные методы стремятся сбалансировать эти аспекты. Например, t-SNE моделирует распределение попарных сходств в исходном многомерном пространстве и воспроизводит его в низкоразмерном представ-

лении, что позволяет сохранить как локальные, так и частично глобальные структуры. Тем не менее такие методы остаются чувствительными к выбору гиперпараметров и теряют интерпретируемость, поскольку представляют данные в виде линейных или нелинейных комбинаций исходных признаков, а не в терминах самих признаков [17]. Это делает их неприменимыми для задач, где важно понимать, какие именно исходные признаки вносят вклад в модель.

Обертывающие методы, такие как метод рекурсивного исключения признаков (Recursive Feature Elimination, RFE) и метод исключения предсказуемых признаков (Predictable Feature Elimination, PFE), обеспечивают высокую точность, но требуют значительных вычислительных ресурсов. RFE итеративно исключает наименее важные признаки на основе обученной модели, но требует многократного переобучения, что делает его неэффективным для больших данных [18]. PFE оценивает предсказуемость каждого признака по остальным с помощью вспомогательной модели машинного обучения и последовательно исключает признаки, которые могут быть восстановлены через остальные [19]. В отличие от PCA, PFE сохраняет интерпретируемость, так как работает с исходными признаками, но не учитывает целевую переменную и не оценивает стабильность отбора. Сравнение эффективности методов для разных типов задач приводится, например, в [20, 21].

На практике для выбора адекватного метода требуется компромисс между статистической значимостью признаков и воспроизводимостью их отбора. Многие подходы фокусируются только на одном из этих аспектов, что снижает надежность итогового решения, особенно в условиях шумных, разреженных или несбалансированных данных.

Целью данной работы является разработка нового метода отбора признаков, сочетающего пошаговое сокращение признакового пространства с одновременной оценкой значимости и стабильности признаков. Результатом работы стал оригинальный метод прогрессивной фильтрации признаков на основе стабильности и значимости (Progressive Feature Filtering with Stability and Significance, PFF-SS, или PF^{2}S), основанный на итеративной фильтрации. Метод позволяет на каждом шаге выявлять зависимые признаки (коррелирующие, мультикоррелирующие или информационно-связанные), которые ранжируются по комбинированному критерию значимости и стабильности (последняя оценивается через бутстрэп-выборки), выполнять оценку сложности модели при исключении признака и в результате исключать наименее устойчивые и слабые признаки, не увеличивающие обобщающую способность модели. Метод предназначен для данных, собранных в инфраструктуре интернета вещей, развернутой в рамках корпоративной сети научного центра.

Постановка задачи исследования данных IoT

Набор данных, содержащий сетевой трафик интернета вещей, охватывает шесть ключевых категорий: временные характеристики, флаги протоколов TCP и MQTT, параметры скорости соединений, статистические данные по заголовкам пакетов, свойства полезной нагрузки и объемные характеристики при массовой передаче данных [22]. Требуется построить устойчивые классы сетевой активности, которые можно использовать для разметки данных и последующего анализа новых наблюдений в целях выявления сетевых аномалий. Эта задача сводится к задаче кластеризации — автоматического разбиения объектов на группы на основе схожести их признаковых описаний.

Пусть $\mathbf{X} \in \mathbb{R}^{n \times m}$ — матрица наблюдений, где каждая строка соответствует объекту, а каждый столбец — признаку. Обозначим $\mathbf{I} = \{1, 2, \ldots, n\}$ — множество индексов объектов, $\mathbf{J} = \{1, 2, \ldots, m\}$ — множество индексов признаков, x_{ij} — значение j-го признака для i-го объекта, где $i \in \mathbf{I}$, $j \in \mathbf{J}$. Каждый объект $i \in \mathbf{I}$ описывается вектором $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{im}) \in \mathbb{R}^m$. Каждый признак $j \in \mathbf{J}$ описывается вектором $\mathbf{p}_i = (x_{1j}, x_{2j}, \ldots, x_{mj}) \in \mathbb{R}^n$. Матрица наблюдений может быть представлена через множество объектов или множество признаков:

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{bmatrix} = \begin{bmatrix} p_1 & p_2 & \cdots & p_m \end{bmatrix}. \quad (1)$$

Обозначим множество объектов как $\chi = (x_i | i \in \mathbf{I})$, множество признаков как $\mathbf{P} = \{p_j \mid j \in \mathbf{J}\}$. Задача кластеризации заключается в разбиении множества объектов χ на K непересекающихся подмножеств:

$$\chi = \bigcup_{k=1}^{K} \chi_k, \quad (2)$$

где $\chi_k \subseteq \chi$, $\chi_k \neq \varnothing$, $\chi_{k1} \cap \chi_{k2} = \varnothing$ для $\forall\, k1 \neq k2$. χ_k — множество объектов, отнесенных к k-му кластеру, $k = [1, K]$.

Такое разбиение выполняет соответствующее разбиение множества индексов I на подмножества

$$\mathbf{C}_k = \{i \in \mathbf{I} \mid x_i \in \chi_k\} \quad (3)$$

такие, что

$$\mathbf{I} = \bigcup_{k=1}^{K} \mathbf{C}_k, \quad (4)$$

где $\mathbf{C}_k \neq \varnothing$, $\mathbf{C}_{k1} \cap \mathbf{C}_{k2} = \varnothing$ для $\forall\, k1 \neq k2$.

Для построения такого разбиения было рассмотрено несколько методов кластеризации, представляющих разные подходы к группировке данных [23]:

— метод K-средних эффективен для компактных, сферических кластеров, но чувствителен к выбросам и может не находить сложные структуры;

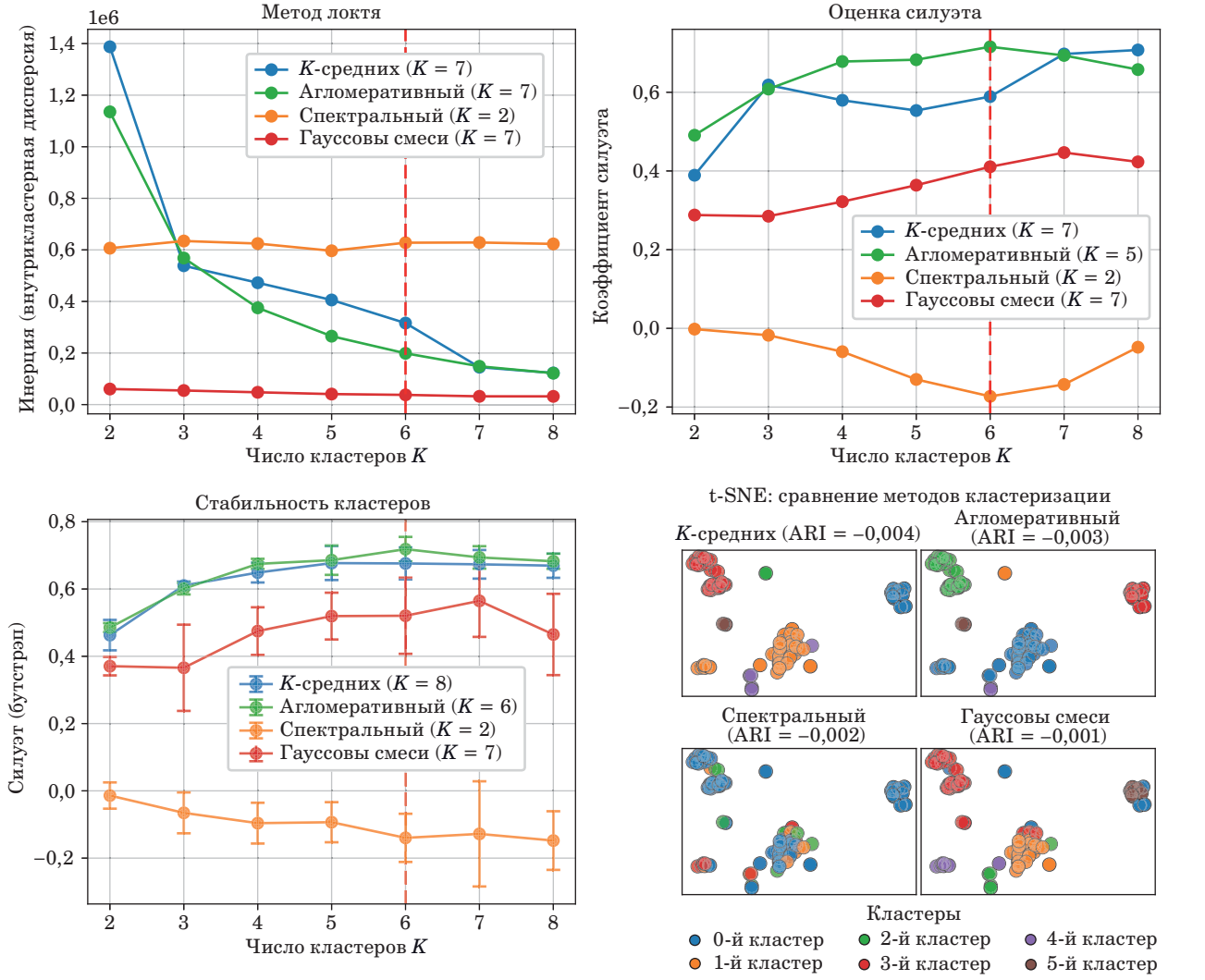
— метод агломеративной иерархической кластеризации (Hierarchical Agglomerative Clustering) позволяет выделять кластеры произвольной формы и анализировать иерархию группировок;

— метод спектральной кластеризации (Spectral Clustering) основан на спектральном разложении матрицы сходства, эффективен для кластеров с нелинейной структурой, но чувствителен к начальным параметрам;

— модель гауссовых смесей (Gaussian Mixture Model) позволяет представить данные как смесь многомерных распределений, за счет чего применима при наличии перекрытий и сложной внутренней структуры.

Оптимальное число кластеров определяется на основе вычисления внутрикластерной дисперсии, коэффициентов силуэта и оценки стабильности кластеризации на бутстрэп-выборках по скорректированному индексу Ренда (Adjusted Rand Index, ARI [24]) при различных значениях K. Результаты и визуализация кластеров через t-SNE [25] приведены на рис. 1.

Как видно из рисунка, рекомендованное количество кластеров различается для разных мето-



■ ***Рис. 1.*** Выбор оптимального количества кластеров

■ ***Fig. 1.*** Determining the optimal number of clusters

дов. Применение оценок компактности и разделенности кластеров тоже не позволяет однозначно сделать выбор K. Все методы демонстрируют низкую стабильность при увеличении числа кластеров. Коэффициент устойчивости ARI близок к нулю или отрицательный для всех методов. Это указывает на то, что признаковое пространство избыточно, структура данных не выражена явно, кластеры не компактны и не разделены.

Для уменьшения размерности признакового пространства применен метод PCA. Построена система обобщенных признаков в виде линейных комбинаций исходных переменных, объясняющих заданную долю общей дисперсии данных (не менее 95 %). На основе полученного сжатого представления выполнена кластеризация с использованием описанных выше алгоритмов (рис. 2).

Анализ результатов кластеризации (после обработки данных методом PCA) показал пересечение кластеров и их низкую стабильность (оценки ARI на бутстрэп-подвыборках близки к нулю). Разные алгоритмы кластеризации демонстрировали значительное расхождение в результатах: то, что один метод выделял как отдельный кластер и формировал компактные и изолированные группы, другой объединял с соседними группами, выделяя более протяженные структуры. Такая несогласованность затрудняет выбор единого, предпочтительного разбиения. Для оценки качества кластеризации выделенные группы проецировались обратно в исходное признаковое пространство, и их содержательная однородность анализировалась экспертами предметной области. Семантическая оценка осмысленности кластеров показала, что схожие с точки зрения предметной области сессии нередко оказывались в разных кластерах, в то время как существенно различающиеся сетевые события объединялись в один класс. Причиной этого эффекта является высокая степень зависимости между признаками, включая корреляцию, мультиколлинеарность и функциональную взаимозависимость. Такие признаки вносят избыточный вклад в отдельные направления признакового пространства, что искажает его геометрию и приводит к формированию некорректных кластеров.

Для устранения этого эффекта автором предложен новый метод отбора признаков PFF-SS (PF^{2}S), основанный на оценке стабильности и значимости (вклада) признаков в структуру данных. В методе введены метрики, которые позволяют последовательно исключать избыточные признаки при минимальной потере информативности, обеспечивая сохранение ключевых свойств признакового пространства на каждом этапе преобразования.

PF^{2}S — метод сокращения размерности признакового пространства

Цель метода PF^{2}S — пошагово сократить множество признаков $\mathbf{P} = \{p_j \mid j \in \mathbf{J}\}$ до подмножества $\mathbf{P}^H \subseteq \mathbf{P}$, удовлетворяющего критериям информативности, устойчивости и независимости. Для его применения необходимо обеспечить выполнение в матрице наблюдений $\mathbf{X}$ (1) следующих условий.

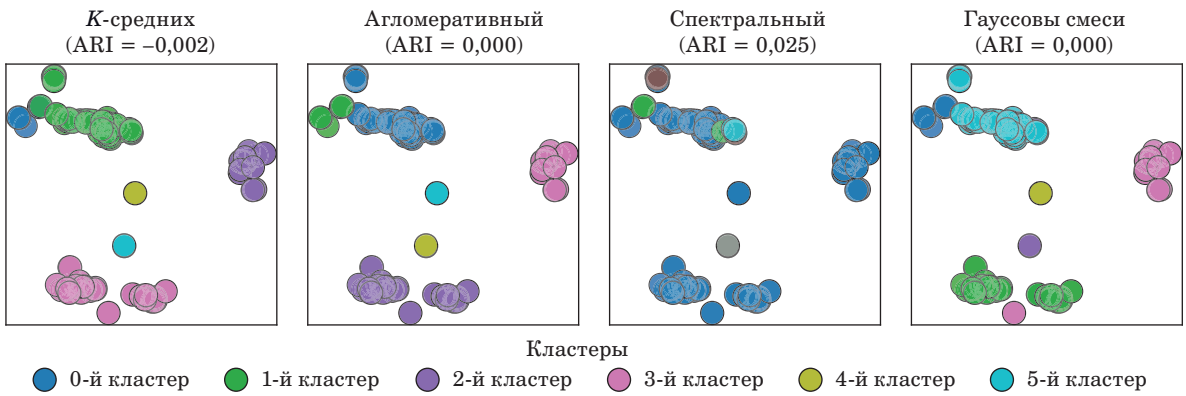
1. Все элементы матрицы $\mathbf{X}$ являются числовыми и определенными:

$$\sum_{i=1}^{n}\sum_{j=1}^{m} \mathrm{I}\left(x_{ij} = \varnothing\right) = 0, \tag{5}$$

где $\mathrm{I}(\cdot)$ — индикаторная функция; x_{ij} — (i, j) значение в матрице $\mathbf{X}$.

2. Все признаки имеют дисперсию, превышающую заданный порог:

$$\forall j \ \sigma_j(p_j) \geq \tau_\sigma, \tag{6}$$



■ ***Рис. 2.*** Результат кластеризации в пространстве главных компонент

■ ***Fig. 2.*** Clustering results in the principal component space

где σ_j — стандартное отклонение; τ_σ — порог дисперсии.

3. Матрица является центрированной. Для этого построим матрицу $\tilde{\mathbf{X}} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_n)$, $\tilde{x}_i = \left(\tilde{x}_{i1}, \tilde{x}_{i2}, \ldots, \tilde{x}_{im}\right)$, $\tilde{x}_{ij} = \dfrac{x_{ij} - \bar{x}_j}{\sigma_j}$. Это требование позволит построить ковариационную матрицу $\mathbf{\Sigma}_{\mathbf{X}} = \dfrac{1}{n-1}\tilde{\mathbf{X}}^{\mathrm{T}} \cdot \tilde{\mathbf{X}}$ и вычислить собственные числа $\lambda_1, \lambda_2, \ldots, \lambda_m$, собственные векторы $v_1, v_2, \ldots, v_m$ и сингулярные числа $\delta_1, \delta_2, \ldots, \delta_m$ ($\delta_i = \sqrt{\lambda_i}$).

Введем целевую функцию оценки признакового пространства

$$L(\tilde{\mathbf{X}}) = \alpha_1 \cdot K\left(\tilde{\mathbf{X}}\right) + \alpha_2 \cdot I\left(\tilde{\mathbf{X}}\right) + \sum_{l=1}^{L} \alpha_{3,l} \cdot R\left(\mathbf{F}, \tilde{\mathbf{X}}\right), \quad (7)$$

где K — число обусловленности матрицы $\tilde{\mathbf{X}}$, отражающее степень мультиколлинеарности; I — средняя взаимная информация между признаками, оценивающая нелинейную зависимость; R — мера сложности для семейства функций $\mathbf{F}$; $\alpha_1, \alpha_2, \alpha_3, \ldots, \alpha_l$ — весовые коэффициенты, позволяющие настраивать приоритеты между компонентами (по умолчанию $\alpha_i = 1$).

Минимизация L соответствует улучшению структуры признакового пространства за счет снижения зависимости между признаками и повышения устойчивости результата. Распишем каждое из слагаемых в L. Число обусловленности матрицы $\tilde{\mathbf{X}}$ вычисляется как $K = \delta_{\max}/\delta_{\min}$, где $\delta_{\max}$ — наибольшее сингулярное число, $\delta_{\max} \neq 0$ — наименьшее сингулярное число. K показывает, насколько данные устойчивы к малым изменениям (для линейной зависимости). Средняя взаимная информация I отвечает за оценку меры нелинейной зависимости и избыточности признаков:

$$I\left(\tilde{\mathbf{X}}\right) = \frac{2}{m(m-1)} \sum_{j=1}^{m} \sum_{k=j+1}^{m} I_{jk}, \quad (8)$$

где элемент I_{jk} определяет величину взаимной информации между P_j и P_k:

$$I_{jk} = \sum_{\tilde{x}_{ij} \in p_j} \sum_{\tilde{x}_{ik} \in p_k} f(\tilde{x}_{ij}, \tilde{x}_{ik}) \cdot \ln\left(\frac{f(\tilde{x}_{ij}, \tilde{x}_{ik})}{f(\tilde{x}_{ij}) \cdot f(\tilde{x}_{ik})}\right), \quad (9)$$

где $f(\tilde{x}_{ij}, \tilde{x}_{ik})$ – частота совместного появления значений признаков p_j и p_k для i-го объекта; $f(\tilde{x}_{ij})$ и $f(\tilde{x}_{ik})$ – частота появления каждого значения признака в отдельности. Данная формула определена для эмпирических частот [26].

В качестве меры сложности R будет применяться оценка Радемахера

$$R\left(F, \tilde{\mathbf{X}}\right) = E_{\varepsilon_i, \tilde{\mathbf{X}}}\left[\sup_{F \in \mathbf{F}} \left| \frac{1}{n} \sum_{i=1}^{n} \varepsilon_i F\left(\tilde{\mathbf{X}}_i\right) \right|\right], \quad (10)$$

где ε_i — случайная величина (переменная Радемахера, принимающая значения +1 и –1 с вероятностью 1/2); $F \in \mathbf{F}$ — функция, $\mathbf{F}_{\tilde{l}}$ — семейство функций; E — среднее по всем ε_i и $\tilde{\mathbf{X}}$. R — теоретическая мера, которая показывает способность аппроксимировать данные и избегать переобучения. В работе [27] обосновано применение меры сложности Радемахера (Rademacher complexity) для оценки обобщающей способности моделей, обученных на немаркированных данных.

Для метода PF^{2}S необходимо определить, какие признаки рассматривать в качестве кандидатов на удаление и по каким критериям делать выбор. Для каждого признака $p_j \in \mathbf{P}$, где $j \in \mathbf{J}$, определим три типа подмножеств, включающих признаки, связанные с p_j различными видами зависимости: $\mathbf{C}_j \subseteq \mathbf{P}$ — коррелирующих с p_j признаков, $\mathbf{V}_j \subseteq \mathbf{P}$ — мультиколлинеарных с p_j признаков и $\mathbf{I}_j \subseteq \mathbf{P}$ — взаимозависимых с p_j признаков. Для каждого признака p_j формируется подмножество: $\mathbf{D}_j = (\mathbf{C}_j, \mathbf{V}_j, \mathbf{I}_j)$, включающее все признаки, находящиеся в зависимости с p_j. Объединенное множество всех признаков используется для последующей фильтрации:

$$\mathbf{D} = \bigcup_{j=1}^{m} \mathbf{D}_j. \quad (11)$$

Ниже описаны правила построения каждого из этих подмножеств. Коррелирующими с p_j считаются признаки, коэффициент корреляции с которыми превышает заданный порог:

$$\mathbf{C}_j = \left\{ p_k \in \mathbf{P} \setminus \{p_j\} \,\middle|\, |\rho_{jk}| \geq \tau_\rho \right\}, \quad (12)$$

где ρ_{jk} — коэффициент корреляции между признаком p_j и p_k; $\tau_\rho \in [0, 1]$ — пороговое значение корреляции.

Для оценки мультиколлинеарности признака p_j выполняется построение линейной модели его восстановления по всем остальным признакам: $p_j = \sum_{k \neq j} \beta_k p_k + \zeta_j$, где p_j — вектор значений j-го признака; β_k — коэффициенты, найденные методом наименьших квадратов; p_k — векторы значений остальных признаков; ζ_j — вектор остатков. Поскольку данные предварительно центрированы, свободный член модели β_0 отсутствует.

Признак $p_j \in \mathbf{P}$ мультиколлинеарный, если коэффициент инфляции дисперсии (Variance Inflation Factor, VIF [28]) Vif_j превышает пороговое значение τ_V, т. е.

$$Vif_j = \frac{1}{1-\mathcal{R}_j^2} \geq \tau_V, \quad (13)$$

где $\mathcal{R}_j^2$ — коэффициент детерминации признака p_j, который вычисляется по формуле

$$\mathcal{R}_j^2 = \frac{\sum_{i=1}^{n}(\hat{x}_{ij} - \bar{x}_j)^2}{\sum_{i=1}^{n}(\tilde{x}_{ij} - \bar{x}_j)^2}, \quad (14)$$

где $\tilde{x}_{ij}$ – истинное значение j-го признака у i-го объекта; $\hat{x}_{ij} = \sum_{k \neq j} \beta_k \tilde{x}_{ik}$ – предсказанное значение, $\tilde{x}_{ik}$ – значение k-го признака у i-го объекта; $\bar{x}_j = \frac{1}{n}\sum_{i=1}^{n} \tilde{x}_{ij}$ – среднее значение.

Для каждого признака, удовлетворяющего условиям (13), (14), анализируется структура зависимости. Выбираются признаки p_k, входящие в модель (13) с коэффициентами β_k, превышающими заданный порог. Формируется подмножество таких признаков

$$\mathbf{V}_j = \{p_k \in \mathbf{P} \setminus \{p_k\} \mid \beta_k > \tau_\beta\}, \quad (15)$$

где $\tau_\beta \geq 0$ — порог учета признака.

Взаимозависимыми считаются признаки, между которыми коэффициент взаимной информации превышает заданный порог. Для каждого p_j определим множество, отражающее нелинейные зависимости между признаками:

$$\mathbf{I}_j = \{p_k \in \mathbf{P} \setminus \{p_k\} \mid I_{jk} > \tau_I\}, \quad (16)$$

где I_{jk} — величина взаимной информации между p_j и p_k, вычисляемая по (8); τ_I — порог сильной зависимости.

Для принятия решения об исключении признака, входящего в множество $\mathbf{D}$, для каждого множества $\mathbf{D}_j \subseteq \mathbf{D}$ сформируем расширенное множество $\mathbf{D}'_j = \mathbf{D}_j \cup \{p_j\}$ и введем метрики стабильности и значимости.

Стабильность признака $p_k \in \mathbf{D}'_j$ оценивается через его воспроизводимость на случайных подвыборках:

$$S(p_k) = \frac{1}{T-1}\sum_{t=1}^{T-1} M(p_k^{(t)}, p_k^{(t+1)}), \quad (17)$$

где T — количество случайных выборок, полученных из $\tilde{\mathbf{X}}$; $M(p_k^{(t)}, p_k^{(t+1)})$ – взаимная информация, вычисленная по (9) для признака p_k на t-й и $(t+1)$-й подвыборках. Чем выше $S(p_k)$, тем стабильнее признак.

Значимость признака $p_k \in \mathbf{D}'_j$ вычисляется как вклад в объясненную дисперсию:

$$W(p_k) = \sum_{i=1}^{d} \lambda_i \cdot v_i(p_k)^2, \quad (18)$$

где $d = \min\left\{k \middle| \frac{\sum_{i=1}^{k}\lambda_i}{\sum_{i=1}^{m}\lambda_j} \geq \varsigma\right\}$ – глубина редуцированного пространства, содержащего ς дисперсии исходных данных, $\varsigma = (0, 1]$ — порог; $\lambda_1, \lambda_2, \ldots, \lambda_d$ — собственные значения ковариационной матрицы; $v_i(p_k)$ — компонента собственного вектора v_i, соответствующая признаку p_k. Чем выше $W(p_k)$, тем важнее признак с точки зрения структуры данных.

Для сложных структур данных предлагается дополнительно ввести метрику нелинейной значимости, которая включает веса, формируемые с использованием моделей машинного обучения (например, Lasso), оценивающих предсказательную способность признака относительно других признаков, рассматриваемых в роли целевых переменных по всем остальным признакам.

Построим множество кандидатов на удаление:

$$\mathbf{G}_j = \{p_k \in \mathbf{D}'_j \mid (S(p_k) < \tau_S) \wedge (W(p_k) < \tau_W)\}, \quad (19)$$

где τ_S, τ_W — пороги стабильности и значимости соответственно.

Если $\mathbf{G}_j = \varnothing$, группа $\mathbf{D}_j$ считается устойчивой и информативной — удаление признаков не производится.

Если $\mathbf{G}_j \neq \varnothing$ и существует $p^* \in \mathbf{D}'_j$ такой, что

$$p^* = \arg\min_{p_k \in \mathbf{G}_j}\left(\alpha\frac{S(p_k)}{S_{\max}} + \beta\frac{W(p_k)}{W_{\max}}\right), \quad (20)$$

где $S_{\max} = \max_{p_k \in \mathbf{G}_j} S(p_k)$, $W_{\max} = \max_{p_k \in \mathbf{G}_j} W(p_k)$, α, $\beta \geq 0$, $\alpha + \beta = 1$ — веса, позволяющие настраивать приоритет, то $\mathbf{P} = \mathbf{P} \setminus p^*$ и $\mathbf{D} = \mathbf{D} \setminus \mathbf{D}_j$.

Алгоритм итеративной фильтрации признаков заключается в пошаговом построении множества $\mathbf{D}$, выборе признаков для удаления из признакового пространства $\mathbf{P}$ и его исключении при условии, что это не ухудшает целевую функцию L. Пусть $\mathbf{A}$ — множество операций фильтрации признакового пространства. На каждом шаге $h = \{0, 1, \ldots, H\}$ из множества $\mathbf{A}$ выбирается операция $A^h \in \mathbf{A}$ исключения признака, удовлетворяющая условию

$$A^h = \arg\min_{A' \in \mathbf{A}} L(A'(\mathbf{P}^{s-1})), \quad (21)$$

где A' — операция, для которой $L(A'(\mathbf{P}^{h-1})) \leq L(\mathbf{P}^{h-1})$.

Если такая A^h существует, она применяется к текущему множеству признаков:

$$\mathbf{P}^h = A^h(\mathbf{P}^{h-1}). \quad (22)$$

Пусть A^h выполняет исключение признака $p \in \mathbf{P}^{h-1}$ из признакового пространства, тогда $\mathbf{P}^h = \mathbf{P}^{h-1} \backslash p$, $\mathbf{A} = \mathbf{A} \backslash A^h$. Итоговое признаковое пространство на шаге H определяется композицией всех преобразований

$$\mathbf{P}^H = A^H \circ A^{H-1} \circ \ \dots \ \circ A^1(\mathbf{P}), \qquad (23)$$

где $\mathbf{P}$ — исходное признаковое пространство; «$\circ$» — операция композиции.

Фильтрация признакового пространства завершается при выполнении хотя бы одного из условий

$$|\mathbf{P}^h| \le m_{\min},$$

$$\min_{A' \in \mathbf{A}} L(A'(\mathbf{P}^h)) > L(\mathbf{P}^h), \ h = H. \qquad (24)$$

Условия (24) определяют, что число признаков достигло заданного минимума, или ни одна операция из $\mathbf{A}$ не приводит к улучшению L, или достигнуто максимальное число итераций H. Применение многофакторной целевой функции, включающей меры мультиколлинеарности, нелинейной зависимости и сложности модели, позволяет выявлять как явные, так и скрытые избыточности в данных. Благодаря гибкости в выборе порогов и весов, PF^{2}S может быть адаптирован под различные типы данных и задачи.

Применение метода PF^{2}S для признакового пространства IoT

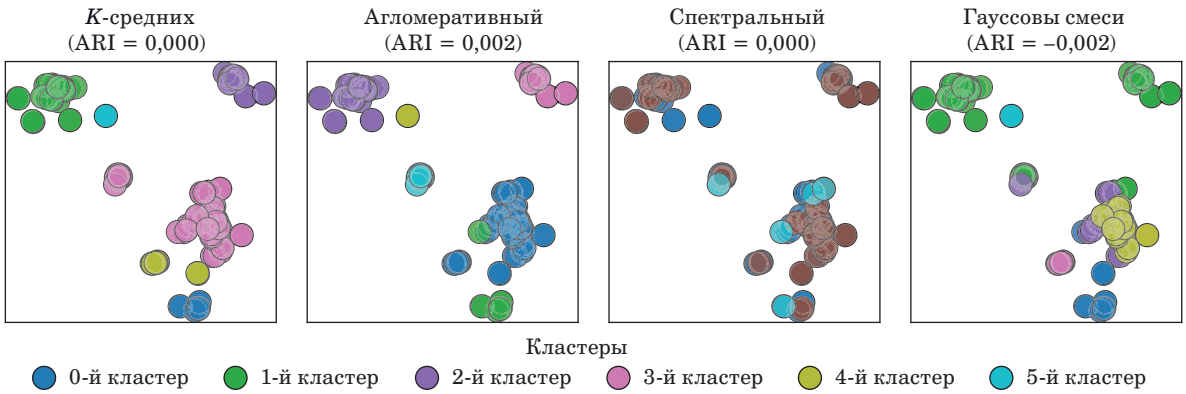
Для оценки эффективности метода PF^{2}S проведено сравнение с двумя базовыми подходами: методом PCA и методом RFE. Метод рекурсивного исключения признаков применялся в двух конфигурациях: с фиксированным числом отбираемых признаков и с порогом объясненной дисперсии 95 %. После сокращения признакового пространства выполнялась кластеризация с использованием четырех методов: K-средних, агломеративного, спектрального и модели гауссовых смесей. Для визуализации результатов применялся метод t-SNE (рис. 3). Стабильность кластеризации оценивалась с помощью скорректированного индекса Рэнда при бутстрэп-повторениях.

Значения ARI, полученные для методов кластеризации после преобразования признакового пространства методами PCA и RFE, не показывают высокой стабильности, что характеризует низкую воспроизводимость кластеров и свидетельствует о неустойчивости результатов при небольших изменениях в данных.

Для сравнения методов PF^{2}S и RFE были рассчитаны ключевые характеристики: время выполнения, итоговая размерность признакового пространства, число обусловленности, наличие коррелирующих и взаимозависимых признаков, среднее значение силуэтного коэффициента и сложность по Рademахеру. Расчет сложности выполнялся для нескольких моделей, для сравнения выбраны значения, полученные на линейной модели со случайными весами, которые позволяют оценить склонность метода к переобучению на шум.

Результаты, приведенные в таблице, показывают, что PF^{2}S обеспечивает более высокое качество кластеризации (силуэтный коэффициент — 0,82) по сравнению с RFE (силуэтный коэффициент — 0,68). Это объясняется тем, что PF^{2}S выполняет поэтапное удаление признаков с контролем стабильности и значимости на каждом шаге, что снижает риск переобучения и повышает воспроизводимость результатов.

Полученное с помощью PF^{2}S признаковое пространство является компактным (17 признаков) и обладает высокой численной устойчивостью (число обусловленности — 2,83), что указывает на отсутствие мультиколлинеарности. В отличие от



■ ***Рис. 3.*** Кластеризация в пространстве RFE
■ ***Fig. 3.*** Clustering in the RFE feature space

- Сравнение PF²S и RFE
- Comparison of PF²S and RFE

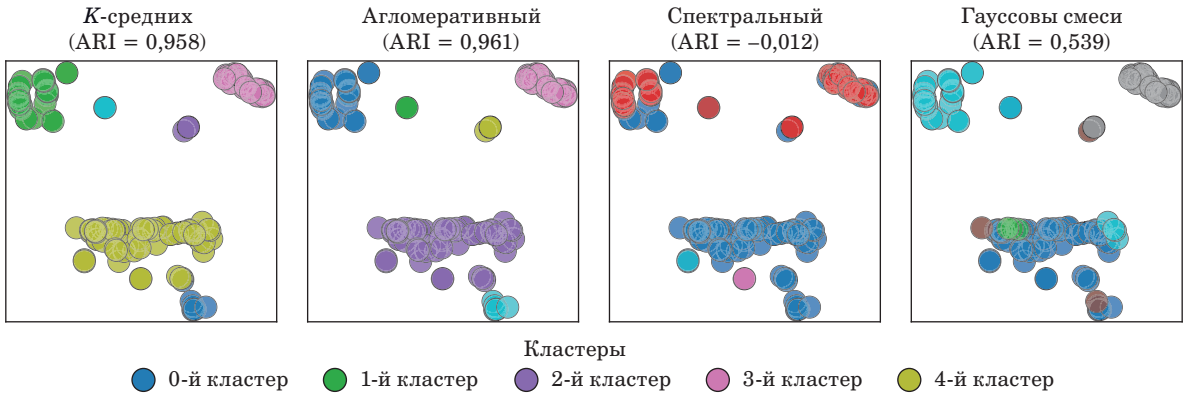
Выполненные действия	Число признаков	Сложность по Радемахеру	Силуэтный коэффициент
Сбор, парсер, загрузка	347	–	–
Предобработка, очистка	275	0,417	0,62
Метод прогрессивной фильтрации признаков Progressive Feature Filtering with Stability and Significance (PF²S)			
1. Коррелирующие, нестабильные	223	0,415	0,62
2. Коррелирующие, незначительные	79	0,233	0,64
3. Мультиколлинеарные, нестабильные	25	0,145	0,72
4. Взаимные, нестабильные	22	0,117	0,77
5. Взаимные, незначительные	**17**	**0,087**	**0,82**
Время выполнения: **6,72 с** (~1000 строк), **58,46 с** (~10 200 строк) Коррелирующие: **0** Взаимозависимые: **0** Число обусловленности: **2,83**			
Метод рекурсивного сокращения размерности (95 % дисперсии) Recursive Feature Elimination (RFE)			
Выбор подмножества признаков RFE	**19**	**0,083**	**0,68**
Время выполнения: **41,42 с** (~1000 строк), **147,42 с** (~10 200 строк) Коррелирующие: **8** Взаимозависимые: **5675,17** Число обусловленности: **75,17**			

него, признаковое пространство, сформированное с помощью RFE, характеризуется высоким числом обусловленности (75,17), свидетельствующим о сильной мультиколлинеарности и потенциальной неустойчивости модели.

К построенному PF²S признаковому пространству были применены методы кластеризации, выполнена визуализация t-SNE и расчет ARI, аналогично предыдущим подходам. Анализ количества кластеров и их структуры показал, что для нового признакового пространства эффективным является разделение данных на пять кластеров (рис. 4).

Для оценки качества кластеризации была выполнена семантическая интерпретация выделенных групп. В признаковом пространстве PF²S остались информативные и устойчивые признаки: минимальный размер пакета, интервалы между пакетами, скорости передачи данных, количество пакетов с флагами и другие, релевантные для анализа сетевого трафика.

Сравнение разбиений на пять и шесть кластеров показало, что 5-кластерная структура является более интерпретируемой: каждый кластер четко соответствует определенному типу сетевой активности. В случае шести кластеров один из них оказывается малочисленным и дублирует другие, что указывает на избыточность разбиения. Выделенные пять кластеров интерпретируются следующим образом: 0-й кластер содержит



- ***Рис. 4.*** Кластеризация в пространстве PF²S
- ***Fig. 4.*** Clustering in the PF²S feature space

одиночные пакеты (SYN, ACK, RST) — сканирование портов и фоновый трафик, в 1-й кластер вошли средние сессии с двусторонним обменом, 2-й кластер объединил высокоскоростные сессии (потоковые передачи), в 3-й кластер выделился трафик с признаками сетевой перегрузки (потенциальные DDoS-атаки), 4-й кластер собрал очень короткие сессии.

Методы *K*-средних и агломеративной кластеризации показали схожие результаты: они эффективно разделили трафик по объему данных, длительности сессий и наличию специфических флагов. Спектральный метод, чувствительный к глобальной структуре данных, выделил редкие и слабо выраженные события (например, сессии с флагом ECE). Метод гауссовых смесей продемонстрировал распределения данных по тем же типам трафика, что и в других методах. Его кластеры имеют четкую структуру и учитывают разделение по вероятностным признакам.

Заключение

Описанный в работе метод PFF-SS (PF^2S) представляет собой систематический подход к сокращению признакового пространства, сочетающий анализ линейных и нелинейных зависимостей с оценкой стабильности и информативности признаков. В отличие от традиционных методов, например метода PCA, новый метод не преобразует исходные признаки, а последовательно исключает избыточные, коррелирующие, мультиколлинеарные и нестабильные компоненты, сохраняя семантическую интерпретируемость оставшегося набора. Это особенно важно в прикладных задачах, таких как анализ сетевого трафика, где физический смысл признаков критичен для интерпретации выделенных паттернов.

В сравнении с методом RFE предложенный подход продемонстрировал существенно лучшие характеристики результирующего признакового пространства. PF^2S обеспечивает более высокое качество кластеризации (видно из расчета силуэтного коэффициента), значительно меньшую сложность модели (по Рademахеру) и формирует численно устойчивое пространство с низким числом обусловленности. Благодаря поэтапному контролю над стабильностью и значимостью признаков на каждом этапе отбора PF^2S снижает риск переобучения и повышает воспроизводимость результатов анализа.

Полученное признаковое пространство компактно и позволяет четко интерпретировать кластеры в соответствии с типами сетевого трафика: фоновые сессии, сканирование портов, веб-запросы, DNS-трафик и признаки сетевой перегрузки. Применение методов *K*-средних и агломеративной кластеризации показало схожие результаты, что подтверждается высокой стабильностью разбиений на бутстрэп-подвыборках (ARI близок к 1,0).

Таким образом, PF^2S представляет собой эффективный, быстрый и интерпретируемый инструмент для подготовки данных в задачах, для которых важны как качество кластеризации, так и понимание природы выделенных событий. Дальнейшее исследование будет направлено на выбор оптимального метода кластеризации для полученного признакового пространства и использование построенных меток для классификации трафика в сетях интернета вещей, а также на адаптацию PF^2S для потоковой обработки данных в реальном времени.

Финансовая поддержка

Работа поддержана Красноярским математическим центром, финансируемым Министерством науки и высшего образования Российской Федерации в рамках мероприятий по созданию и развитию региональных НОМЦ (Соглашение № 075-02-2025-1606).

Литература

1. **Gandomi A., Haider M.** Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 2015, vol. 35, iss. 2, pp. 137–144. doi:10.1016/j.ijinfomgt.2014.10.007
2. **Choudhary A.** Internet of Things: A comprehensive overview, architectures, applications, simulation tools, challenges and future directions. *Discov Internet Thing*, 2024, vol. 4, iss. 31. doi:10.1007/s43926-024-00084-3
3. **Ray P. P.** A survey on Internet of Things architectures. *Journal of King Saud University — Computer and Information Sciences*, 2018, vol. 30, iss. 3, pp. 291–319. doi:10.1016/j.jksuci.2016.10.003
4. **Schiller E., Aidoo A., Fuhrer J., Stahl J., Ziörjen M., Stiller B.** Landscape of IoT security. *Computer Science Review*, 2022, vol. 44, pp. 100467. doi:10.1016/j.cosrev.2022.100467
5. **Татарникова Т. М., Богданов П. Ю.** Обнаружение атак в сетях интернета вещей методами машинного обучения. *Информационно-управляющие системы*, 2021, № 6, с. 42–52. doi:10.31799/1684-8853-2021-6-42-52
6. **Ado A., Hamayadji A., Arouna N. N., Moussa A., Asside D., Ousmane T., Alidou M.** Data collection in IoT networks: Architecture, solutions, protocols and challenges. *IET Wireless Sensor Systems*, 2024, vol. 14, iss. 4, pp. 85–110. doi:14.10.1049/wss2.12080

7. **Исаева О. С., Кулясов Н. В., Исаев С. В.** Инфраструктура сбора данных и имитации угроз безопасности сети интернета вещей. *Сибирский аэрокосмический журнал*, 2025, т. 26, № 1, с. 8–20. doi:10.31772/2712-8970-2025-26-1-8-20, EDN: OPICJJ
8. **Andy S., Rahardjo B., Hanindhito B.** Attack scenarios and security analysis of MQTT communication protocol in IoT system. *4th International Conference on Electrical Engineering, Computer Science and Informatics*, 2017, pp. 1–6. doi:10.1109/EECSI.2017.8239179
9. **Исаева О. С., Исаев С. В., Кулясов Н. В.** Формирование адаптивных рассылок брокера данных интернета вещей. *Информационно-управляющие системы*, 2022, № 5, с. 23–31. doi:10.31799/1684-8853-2022-5-23-31, EDN: DNOSCW
10. **Isaeva O. S., Kulyasov N. V., Isaev S. V.** Semantic modeling of the scheme “Publisher-Subscriber” for data analysis of the Internet of Things. *AIP Conference Proceeding*, 2025, no. 3268, pp. 070025. doi:10.1063/5.0257199
11. **Алексеев А. А., Попова Ю. Б., Шестопалов М. Ю.** Алгоритмы нечеткой кластеризации в задачах диагностики технических систем. *Известия вузов. Северо-Кавказский регион. Серия: Технические науки*, 2012, № 3, с. 3–7. EDN: OYZUBP
12. **Xueyi C.** A comprehensive study of feature selection techniques in machine learning models. *Insights in Computer, Signals and Systems*, 2024, no. 1, pp. 65–78. doi:10.70088/xpf2b276
13. **Omaimah A. H., Andrew S.** Assesing the stability and selection performance of feature selection methods under different data complexity. *The International Arab Journal of Information Technology*, 2022, vol. 19, no. 3A, pp. 442–455. doi:10.34028/iajit/19/3A/4
14. **Liu Y., Mu Y., Chen K., Li Y., Guo J.** Daily activity feature selection in smart homes based on pearson correlation coefficient. *Neural Process Lett*, 2020, vol. 51, pp. 1771–1787. doi:10.1007/s11063-019-10185-8
15. **Czajkowski M., Jurczuk K., Kretowski M.** Steering the interpretability of decision trees using lasso regression — an evolutionary perspective. *Information Sciences*, 2023, vol. 638, pp. 118944. doi:10.1016/j.ins.2023.118944
16. **Kamalov F., Sulieman H., Alzaatreh A., Emarly M., Chamlal H., Safaraliev M.** Mathematical methods in feature selection: A review. *Mathematics*, 2025, no. 13, pp. 996. doi:10.3390/math13060996
17. **Cynthia R., Chaofan C., Zhi C., Haiyang H., Semenova L., Zhong C.** Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 2022, no. 16. doi:10.1214/21-SS133
18. **Priyatno A., Widiyaningtyas T.** A systematic literature review: recursive feature elimination algorithms. *Jurnal Ilmu Pengetahuan dan Teknologi Komputer*, 2024, no. 9. pp. 196–207. doi:10.33480/jitk.v9i2.5015
19. **Barbiero P., Squillero G., Tonda A.** Predictable features elimination: an unsupervised approach to feature selection. *Lecture Notes in Computer Science*, 2022, no. 13163, pp. 399–412. doi:10.1007/978-3-030-95467-3_29
20. **Aker Y.** Comparison of PCA and RFE-RF algorithm in bankruptcy prediction. *Gümüşhane Üniversitesi Sosyal Bilimler Dergisi*, 2022, vol. 13, iss. 3, pp. 1001–1008.
21. **Гайнетдинова А. А., Воробьев А. В.** Сравнение методов отбора значимых признаков для классификации геомагнитных данных. *Прикладная математика и вопросы управления*, 2023, № 4, с. 46–54. doi:10.15593/2499-9873/2023.4.02, EDN: LGWAOR
22. **Исаева О. С., Кулясов Н. В., Исаев С. В.** Создание инструментов сбора данных для анализа аспектов безопасности Интернета вещей. *Информационные и математические технологии в науке и управлении*, 2022, № 3(27), с. 113–125. doi:10.38028/ESI.2022.27.3.011, EDN: UKFFWD
23. **Булыга Ф. С., Курейчик В. М.** Алгоритмы агломеративной кластеризации применительно к задачам анализа лингвистической экспертной информации. *Известия ЮФУ. Технические науки*, 2021, № 6(223), с. 73–88. doi:10.18522/2311-3103-2021-6-73-88, EDN: UVKNNZ
24. **Santos J., Embrechts M.** On the use of the Adjusted Rand Index as a metric for evaluating supervised classification. *Lecture Notes in Computer Science*, 2009, no. 5769, pp. 175–184. doi:10.1007/978-3-642-04277-5_18
25. **Бодров А. О.** Применение метода t-SNE для визуализации и кластеризации многомерных данных. *Сборник трудов IV Международного научно-технического форума «Современные технологии в науке и образовании»*, 2021, т. 6, с. 66–69.
26. **Guoping Z.** A unified definition of mutual information with applications in machine learning. *Mathematical Problems in Engineering*, 2015, no. 201874, pp. 1–12. doi:10.1155/2015/201874
27. **Oneto L., Ghio A., Ridella S., Anguita D.** Local Rademacher Complexity: Sharper risk bounds with and without unlabeled samples. *Neural Networks*, 2015, vol. 65, pp. 115–125. doi:10.1016/j.neunet.2015.02.006
28. **Моисеев Н. А.** Сравнительный анализ эффективности методов устранения мультиколлинеарности. *Учет и статистика*, 2017, № 2 (46), с. 62–77. EDN: ZELDIN

UDC 004.6
doi:10.31799/1684-8853-2025-6-15-27
EDN: ERTCQY

Feature filtering method based on stability and significance criteria

O. S. Isaeva[a], Dr. Sc. Tech., Senior Researcher, orcid.org/0000-0002-5061-6765, isaeva@icm.krasn.ru
[a]Institute of Computational Modelling SB RAS, 50/44, Akademgorodok St., 660036, Krasnoyarsk, Russian Federation

Introduction: Network traffic analysis in the Internet of Things (IoT) is complicated by high dimensionality, feature redundancy, and instability. Strong correlation, multicollinearity, and noise degrade clustering quality and hinder interpretation. Moreover, legitimate and anomalous traffic often overlap, making it difficult to formalize class boundaries. Therefore, a feature selection method that ensures stability, compactness, and semantic interpretability is required. **Purpose:** To develop and experimentally evaluate a new method for constructing a stable and interpretable feature space in network traffic clustering tasks — Progressive Feature Filtering with Stability and Significance (PFF-SS, PF^{2}S). **Methods:** We describe a step-by-step PF^{2}S algorithm that combines analysis of linear dependencies (correlation, VIF) and nonlinear dependencies (mutual information) with assessment of feature stability and significance. At each stage, redundant, weakly significant, or unstable features are removed. **Results:** Applying PF^{2}S to an IoT network traffic dataset has reduced the number of features from over 300 to 17 while preserving high informativeness. The comparison with feature spaces reduced by Principal Component Analysis (PCA) and Recursive Feature Elimination shows that PF^{2}S achieves higher metrics in stability, interpretability, and clustering quality. Unlike Principal Component Analysis, PF^{2}S does not transform features but preserves their original semantics. Compared to Recursive Feature Elimination, PF^{2}S eliminates multicollinearity, reduces model complexity, and achieves a silhouette coefficient 17.6% higher. Clusters built on the PF2S-derived feature space are stable (high Adjusted Rand Index) and semantically interpretable. **Practical relevance:** PF^{2}S produces a compact and robust feature space suitable for anomaly detection systems in IoT network traffic. **Discussion:** Promising directions include adapting PF^{2}S for streaming data processing and integrating it with signature-based anomaly detection methods and network traffic ontologies.

Keywords — Internet of Things, feature stability, feature significance, *K*-means clustering, agglomerative clustering, spectral clustering, Gaussian Mixture Model, Principal Component Analysis, Recursive Feature Elimination, network traffic analysis, anomaly detection.

For citation: Isaeva O. S**.** Feature filtering method based on stability and significance criteria. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 15–27 (In Russian). doi:10.31799/1684-8853-2025-6-15-27, EDN: ERTCQY

Financial support

This work is supported by the Krasnoyarsk Mathematical Center and financed by the Ministry of Science and Higher Education of the Russian Federation in the framework of the establishment and development of regional Centers for Mathematics Research and Education (Agreement No. 075-02-2025-1606).

References

1. Gandomi A., Haider M. Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management*, 2015, vol. 35, iss. 2, pp. 137–144. doi:10.1016/j.ijinfomgt.2014.10.007
2. Choudhary A. Internet of Things: A comprehensive overview, architectures, applications, simulation tools, challenges and future directions. *Discov Internet Thing*, 2024, vol. 4, iss. 31. doi:10.1007/s43926-024-00084-3
3. Ray P. P. A survey on Internet of Things architectures. *Journal of King Saud University — Computer and Information Sciences*, 2018, vol. 30, iss. 3, pp. 291–319. doi:10.1016/j.jksuci.2016.10.003
4. Schiller E., Aidoo A., Fuhrer J., Stahl J., Ziörjen M., Stiller B. Landscape of IoT security. *Computer Science Review*, 2022, vol. 44, pp. 100467. doi:10.1016/j.cosrev.2022.100467
5. Tatarnikova T. M., Bogdanov P. Yu. Intrusion detection in internet of things networks based on machine learning methods. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2021, no. 6, pp. 42–52 (In Russian). doi:10.31799/1684- 8853-2021-6-42-52
6. Ado A., Hamayadji A., Arouna N. N., Moussa A., Asside D., Ousmane T., Alidou M. Data collection in IoT networks: Architecture, solutions, protocols and challenges. *IET Wireless Sensor Systems*, 2024, vol. 14, iss. 4, pp. 85–110. doi:14.10.1049/wss2.12080
7. Isaeva O. S., Kulyasov N. V., Isaev S. V. Infrastructure for collecting data and simulating security threats in the Internet of Things network. *Siberian Aerospace Journal*, 2025, vol. 26, no. 1, pp. 8–20 (In Russian). doi:10.31772/2712-8970-2025-26-1-8-20, EDN: OPICJJ
8. Andy S., Rahardjo B., Hanindhito B. Attack scenarios and security analysis of MQTT communication protocol in IoT system. *4th International Conference on Electrical Engineering, Computer Science and Informatics*, 2017, pp. 1–6. doi:10.1109/EECSI. 2017.8239179
9. Isaeva O. S., Isaev S. V., Kulyasov N. V. Formation of adaptive publications from the Internet of Things data broker. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2022, no. 5, pp. 23–31 (In Russian). doi:10.31799/1684-8853-2022-5-23-31, EDN: DNOSCW
10. Isaeva O. S., Kulyasov N. V., Isaev S. V. Semantic modeling of the scheme “Publisher-Subscriber” for data analysis of the Internet of Things. *AIP Conference Proceeding*, 2025, no. 3268, pp. 070025. doi:10.1063/5.0257199
11. Alekseev A. A., Popova Yu. B., Shestopalov M. Yu. Algorithms fuzzy clustering algorithms in technical systems diagnostics problems. *Bulletin of Higher Educational Institutions. North Caucasus Region. Technical Sciences*, 2012, no. 3, pp. 3–7 (In Russian). EDN: OYZUBP
12. Xueyi C. A comprehensive study of feature selection techniques in machine learning models. *Insights in Computer, Signals and Systems*, 2024, no. 1, pp. 65–78. doi:10.70088/xpf2b276
13. Omaimah A. H., Andrew S. Assesing the stability and selection performance of feature selection methods under different data complexity. *The International Arab Journal of Information Technology*, 2022, vol. 19, no. 3A, pp. 442–455. doi:10.34028/iajit/19/3A/4
14. Liu Y., Mu Y., Chen K., Li Y., Guo J. Daily activity feature selection in smart homes based on pearson correlation coefficient. *Neural Process Lett*, 2020, vol. 51, pp. 1771–1787. doi:10.1007/s11063-019-10185-8
15. Czajkowski M., Jurczuk K., Kretowski M. Steering the interpretability of decision trees using lasso regression — an evolutionary perspective. *Information Sciences*, 2023, vol. 638, pp. 118944. doi:10.1016/j.ins.2023.118944
16. Kamalov F., Sulieman H., Alzaatreh A., Emarly M., Chamlal H., Safaraliev M. Mathematical methods in feature selection: A review. *Mathematics*, 2025, no. 13, pp. 996. doi:10.3390/math13060996
17. Cynthia R., Chaofan C., Zhi C., Haiyang H., Semenova L., Zhong C. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 2022, no. 16. doi:10.1214/21-SS133
18. Priyatno A., Widiyaningtyas T. A systematic literature review: recursive feature elimination algorithms. *Jurnal Ilmu*

Pengetahuan dan Teknologi Komputer, 2024, no. 9. pp. 196–207. doi:10.33480/jitk.v9i2.5015
19. Barbiero P., Squillero G., Tonda A. Predictable features elimination: an unsupervised approach to feature selection. *Lecture Notes in Computer Science*, 2022, no. 13163, pp. 399–412. doi:10.1007/978-3-030-95467-3_29
20. Aker Y. Comparison of PCA and RFE-RF algorithm in bankruptcy prediction. *Gümüşhane Üniversitesi Sosyal Bilimler Dergisi*, 2022, vol. 13, iss. 3, pp. 1001–1008.
21. Gainetdinova A. A., Vorobev A. V. Comparison of features elimination methods for geomagnetic data classification. *Applied Mathematics and Control Sciences*, 2023, no. 4, pp. 46–54 (In Russian). doi:10.15593/2499-9873/2023.4.02, EDN: LGWAOR
22. Isaeva O. S., Kulyasov N. V., Isaev S. V. Creating data collection tools to analyze security aspects Internet of Things. *Information and Mathematical Technologies in Science and Management*, 2022, no. 3(27), pp. 113–125 (In Russian). doi:10.38028/ESI.2022.27.3.011, EDN: UKFFWD
23. Bulyga Ph. S., Kureichik V. M. Agglomerative clusterization algorithms for the problems of analysis of linguistic expert information. *Izvestiya SFedU. Engineering Sciences*, 2021, no. 6(223), pp. 73–88 (In Russian). doi:10.18522/2311-3103-2021-6-73-88, EDN: UVKNNZ
24. Santos J., Embrechts M. On the use of the Adjusted Rand Index as a metric for evaluating supervised classification. *Lecture Notes in Computer Science*, 2009, no. 5769, pp. 175–184. doi:10.1007/978-3-642-04277-5_18
25. Bodrov A. O. Application of the t-SNE method for visualization and clustering of multidimensional data. *Sbornik trudov IV Mezhdunarodnogo nauchno-tekhnicheskogo foruma "Sovremennye tekhnologii v nauke i obrazovanii"*. [Proceedings of the IV International Scientific and Technical Forum "Modern Technologies in Science and Education"], 2021, no. 6, pp. 66–69 (In Russian).
26. Guoping Z. A unified definition of mutual information with applications in machine learning. *Mathematical Problems in Engineering*, 2015, no. 201874, pp. 1–12. doi:10.1155/2015/201874
27. Oneto L., Ghio A., Ridella S., Anguita D. Local Rademacher Complexity: Sharper risk bounds with and without unlabeled samples. *Neural Networks*, 2015, vol. 65, pp. 115–125. doi:10.1016/j.neunet.2015.02.006
28. Moiseev N. A. Comparative analysis of the effectiveness of methods for eliminating multicollinearity. *Uchet i statistika*, 2017, no. 2 (46), pp. 62–77 (In Russian). EDN: ZELDIN

УВАЖАЕМЫЕ АВТОРЫ!

Научные базы данных, включая Scopus и Web of Science, обрабатывают данные автоматически. С одной стороны, это ускоряет процесс обработки данных, с другой — различия в транслитерации ФИО, неточные данные о месте работы, области научного знания и т. д. приводят к тому, что в базах оказывается несколько авторских страниц для одного и того же человека. В результате для всех по отдельности считаются индексы цитирования, что снижает рейтинг ученого.

Для идентификации авторов в сетях Thomson Reuters проводит регистрацию с присвоением уникального индекса (ID) для каждого из авторов научных публикаций.

Процедура получения ID бесплатна и очень проста, есть возможность провести регистрацию на 12 языках, включая русский (чтобы выбрать язык, кликните на зеленое поле вверху справа на стартовой странице): https://orcid.org

УДК 004.852
doi:10.31799/1684-8853-2025-6-28-41
EDN: KKSXFO

Научные статьи
Articles

Метаалгоритм управления процессами синтеза моделей машинного обучения

Н. А. Жукова[a]***,*** *доктор техн. наук, доцент, orcid.org/0000-0001-5877-4461, nazhukova@mail.ru*
В. Э. Ковалевский[a]***,*** *младший научный сотрудник , orcid.org/0000-0002-0414-906X*
[a]*Санкт-Петербургский Федеральный исследовательский центр Российской академии наук, 14-я линия В. О., 39, Санкт-Петербург, 199178, РФ*

Введение: *методы автоматизированного машинного обучения позволяют автоматизировать синтез моделей машинного обучения, адаптированных к обработке конкретных данных. Однако эти методы требуют значительных временных и вычислительных затрат.* ***Цель:*** *разработать метаалгоритм управления процессами синтеза моделей машинного обучения, позволяющий снизить вычислительную сложность автоматического построения моделей машинного обучения.* ***Результаты:*** *предложен общий метаалгоритм управления процессами синтеза моделей машинного обучения и частный алгоритм, предусматривающий ограничение пространства поиска за счет использования метаобучения. Предлагаемый частный алгоритм основан на использовании метасвойств данных и онтологии, содержащей правила выбора алгоритмов машинного обучения в зависимости от метасвойств обрабатываемых данных. Наполнение онтологии выполняется за счет предварительной обработки результатов ранее синтезированных моделей машинного обучения. Для частного алгоритма разработан алгоритм формирования обучающей выборки и алгоритм построения онтологии для сокращения пространства поиска. Проведенные экспериментальные исследования показали, что использование предложенного частного алгоритма позволило снизить время синтеза моделей машинного обучения на 41,12 %. Кроме того, у полученных моделей повысились значения достоверности (+0,54 %), полноты (+0,34 %) и AUC (+1,85 %).* ***Практическая значимость:*** *частный алгоритм, разработанный на основе метаалгоритма, помогает снизить вычислительную сложность процесса автоматического построения моделей машинного обучения, что облегчает применение машинного обучения в предметных областях, требующих оперативного построения и адаптации моделей машинного обучения к новым данным и новым задачам.*

Ключевые слова *— автоматизированное машинное обучение, синтез моделей машинного обучения, AutoML, метаобучение, онтологии для AutoML, управление синтезом моделей машинного обучения.*

Для цитирования: Жукова Н. А., Ковалевский В. Э. Метаалгоритм управления процессами синтеза моделей машинного обучения. *Информационно-управляющие системы*, 2025, № 6, с. 28–41. doi:10.31799/1684-8853-2025-6-28-41, EDN: KKSXFO
For citation: Zhukova N. A., Kovalevsky V. E. Meta-algorithm for the process control of complex machine learning model synthesis. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 28–41 (In Russian). doi:10.31799/1684-8853-2025-6-28-41, EDN: KKSXFO

Введение

Область машинного обучения (МО) включает в себя многочисленные алгоритмы, которые позволяют извлекать полезную информацию из большого количества необработанных данных. Эти алгоритмы стали особенно востребованными, когда прогресс в вычислительных технологиях привел к значительному увеличению объемов собираемых и хранимых данных. У алгоритмов МО есть два типа параметров. Параметры первого типа — это параметры, значения которых настраиваются в процессе обучения модели. Так, например, для нейронной сети такими параметрами являются веса нейронной сети, значения которых меняются по мере обучения сети на данных. Параметры второго типа — это параметры, значения которых определяют саму структуру модели. Значение данных параметров устанавливается заранее и не меняется в процессе обучения. Параметры такого типа также называются гиперпараметрами. Для нейронной сети к таким параметрам относятся количество слоев сети, связи между ними и тип активационной функции, а, например, для метода опорных векторов такими параметрами являются ядро и значение константы C, определяющей соотношение между размером разделяющей полосы и суммарной ошибкой. Наличие большого количества алгоритмов МО и отсутствие универсального алгоритма, позволяющего эффективно обрабатывать различные наборы данных, требует в каждом конкретном случае выбирать наиболее подходящий алгоритм или комбинацию алгоритмов и настраивать их гиперпараметры. Следует отметить, что настройка гиперпараметров оказывает значительное влияние на эффективность работы алгоритмов.

Построение модели МО может быть выполнено как в ручном режиме [1, 2], когда пользователь самостоятельно конструирует модель из различных алгоритмов, выбирая и настраивая их, основываясь на собственном опыте, так и в автоматизированном режиме [3, 4], когда синтез моде-

ли осуществляется программными средствами. При создании модели вручную пользователь может применять рекомендательные системы, содержащие экспертные знания, а также знания, полученные из описаний ранее решенных задач. Выполняя запрос к такой системе, пользователь указывает метасвойства текущей задачи и получает рекомендации по алгоритму и значениям его гиперпараметров. Также для выдачи рекомендаций по построению модели может быть использован нейросетевой подход, при котором конструируется нейронная сеть, которая затем обучается на различных задачах [5].

При автоматизированном построении моделей МО применяются средства автоматизированного машинного обучения (Automated Machine Learning – AutoML) [4, 6], которые предполагают применение машинного обучения к самому себе. Данные методы рассматривают множество алгоритмов и их гиперпараметров как параметры, которые также могут быть настроены на основе обучающих данных, а построение модели рассматривается как поиск наиболее подходящего варианта в пространстве гиперпараметров. Существует несколько различных подходов к поиску моделей МО, среди которых поиск по сетке (Grid Search) [7], случайный поиск (Random Search) [8], байесовская оптимизация (Bayesian Optimization) [9], генетические алгоритмы [10]. В случае использования AutoML системы она подменяет собой эксперта, и сама конструирует модель МО. Пользователь может либо применить получившуюся модель, либо использовать ее как исходную рекомендацию и дополнительно настроить. К настоящему времени разработано значительное число AutoML-систем, построенных на основе различных подходов, среди которых наиболее широкое применение получили AutoWEKA [11], Auto-sklearn [12], TPOT [13], H2O [14] и AutoKeras [15]. Каждая из рассмотренных систем при поиске моделей использует лишь ограниченное подмножество алгоритмов машинного обучения, при этом рассматриваемые различными системами подмножества могут пересекаться.

Реализуемое в AutoML построение моделей МО требует значительных временных и вычислительных ресурсов из-за большого пространства поиска, обусловленного большим количеством алгоритмов и сложностью настройки их гиперпараметров. Для ускорения процесса поиска моделей в AutoML-системах исследователями разработаны методы метаобучения, которые используют данные о ранее решенных задачах для установки начальной точки поиска или ограничения пространства поиска. Метаобучение основывается на опыте решения предшествующих задач и предполагает сбор метаданных, описывающих предыдущие задачи и использованные для их решения модели. Такие метаданные включают в себя точные конфигурации алгоритмов, использованных для построения моделей, в том числе настройки гиперпараметров, полученные оценки моделей, а также свойства решенных задач, называемые метасвойствами. При решении новой задачи выбор алгоритмов и их гиперпараметров основывается на оценке сходства решаемой и предыдущих задач, такое сходство задач может быть оценено, например, как евклидово расстояние в пространстве метасвойств или как расстояние Кульбака – Лейблера [16].

Классификация методов метаобучения [17] включает в себя: обучение на основе оценок моделей, что может использоваться для рекомендации общих конфигураций и пространств поиска конфигураций, а также для переноса знаний из эмпирически схожих задач [18]; обучение на основе метасвойств задач, предусматривающее определение метасвойств задач для нахождения схожих между собой задач и построение метамоделей, которые описывают взаимосвязи между метасвойствами данных, использованными алгоритмами и полученными оценками [19]; обучение на основе существующих моделей [20], которое предполагает перенос параметров обученной модели между схожими задачами, например с помощью трансферного обучения [21], или использование существующих моделей для обучения с малым числом запусков [22].

Существует ряд исследований, направленных на применение онтологий для оптимизации поиска моделей машинного обучения. Так, в [23] предлагается онтология для описания существующих AutoM-систем и их возможностей, которая позволяет выбирать наиболее подходящую AutoML-систему при построении моделей МО. Однако данная онтология не содержит дополнительных знаний, которые могут использоваться для выбора алгоритмов и настройки их гиперпараметров. В работе [24] предлагается онтология, содержащая экспертные знания, в частности знания об алгоритмах, применяемых на различных шагах обработки данных и ассоциированных с ними гиперпараметрах. Онтология используется для выбора алгоритмов МО и построения моделей МО вместо поиска по пространству гиперпараметров. Проведено сравнение предлагаемого решения с системой TPOT на одном наборе данных. Однако в работе не рассматриваются вопросы совместного использования различных существующих AutoML-систем при выборе алгоритмов и построении моделей МО. В [25] предлагается онтология, содержащая знания, позволяющие выбирать алгоритмы построения признакового пространства в зависимости от свойств обрабатываемых данных. Проведены экспериментальные

исследования на 10 наборах данных, в которых осуществляется выбор алгоритмов отбора признаков, но используемые алгоритмы МО фиксированы. Также отсутствует сравнение предлагаемого решения с другими системами. В [26] авторами представляется онтология для семантического описания моделей МО, включающая знания о наборах данных, использованных для обучения моделей, областях применения моделей и алгоритмах МО и позволяющая рекомендовать ранее использованные модели МО. В [27] модели МО представляются в виде графов знаний, при построении моделей МО используется информация о ранее построенных моделях для схожих наборов данных. В экспериментальных исследованиях построение графов выполняется на основе данных из открытого репозитория OpenML. Данное решение может рассматриваться как один из возможных подходов к построению моделей МО и использоваться наряду с другими существующими решениями при выполнении синтеза моделей. Кроме того, графовое представление моделей может применяться как эффективное средство визуализации моделей для работы с моделями конечных пользователей.

Проведенный анализ показал, что разработанные к настоящему времени методы AutoML имеют высокую вычислительную сложность и требуют значительных затрат ресурсов. Существующие AutoML-системы используют для построения моделей лишь ограниченные подмножества алгоритмов МО, которые могут пересекаться. Для оптимизации процесса поиска моделей разработаны различные методы, направленные на решение проблемы высокой вычислительной сложности поиска алгоритмов и настройки их гиперпараметров, однако они являются разрозненными и не предполагают их использования в комбинации с другими методами, что приводит к сложностям применения AutoML при решении практических задач. Для накопления знаний, позволяющих оптимизировать процессы поиска моделей МО, могут использоваться онтологии. Имеются подходы к использованию онтологий для оптимизации МО, в которых применяются онтологии, созданные экспертами вручную, а также подходы, рассматривающие их внедрение как замену другим методам оптимизации. В экспериментальных исследованиях решений, разработанных на основе онтологического подхода, часто отсутствует сравнение с существующими методами, что затрудняет оценку их эффективности. Перечисленные ограничения создают существенные сложности для широкого применения методов МО на практике.

В представленной работе предлагается общий метаалгоритм управления процессом синтеза моделей МО и реализующий его частный алгоритм, позволяющий обеспечить эффективный поиск моделей за счет ограничения пространства поиска с применением метаобучения и онтологий.

Постановка задачи оптимизации синтеза модели машинного обучения

Модель МО M: $X \to Y$ представляет собой алгоритм с настроенными гиперпараметрами, преобразующий вектор признаков $\vec{\mathbf{x}} \in X$ в целевое значение $y \in Y$, например, в метку класса в случае решения задачи классификации. Обозначим фиксированный набор базовых алгоритмов как $A = \{A^{(1)}, A^{(2)}, \ldots, A^{(n)}\}$. Для каждого алгоритма $A^{(i)}$ задается свой вектор гиперпараметров $\vec{\boldsymbol{\lambda}} \in \Lambda_{A^{(i)}}$. Пусть $D = \{(\vec{\mathbf{x}}_1, y_1), \ldots, (\vec{\mathbf{x}}_w, y_w)\}$ обозначает множество из w наблюдений, состоящих из векторов признаков $\vec{\mathbf{x}}_i$, для которых известны соответствующие им целевые значения y_i. Обозначим модель МО, состоящую из алгоритма A с гиперпараметрами $\vec{\boldsymbol{\lambda}}$, построенную с использованием знаний K, как $M_{A,\vec{\boldsymbol{\lambda}},K}$. Пусть $L(\cdot, \cdot)$ обозначает функцию потерь. В этом случае потери для модели $M_{A,\vec{\boldsymbol{\lambda}},K}$ при обработке данных D могут быть оценены следующим образом:

$$\hat{R}(M_{A,\vec{\boldsymbol{\lambda}},K}, D) = \frac{1}{w}\sum_{i=1}^{w} L(M(\vec{\mathbf{x}}_i), y_i). \quad (1)$$

Задача синтеза модели МО состоит в том, чтобы, используя управляющее воздействие I, построенное с помощью знаний K, найти такой алгоритм и значения его гиперпараметров, которые минимизируют потери при обработке новых данных D_{new}:

$$(A, \vec{\boldsymbol{\lambda}})^* \in \arg\min_{A \in A, \vec{\boldsymbol{\lambda}} \in \Lambda} \hat{R}(M_{A,\vec{\boldsymbol{\lambda}},K}, I(K), D_{new}). \quad (2)$$

Для решения данной задачи необходимо построить такую функцию f, которая на основе множества доступных алгоритмов A и знаний K синтезирует модель МО для обработки D_{new}:

$$f : A \times K \xrightarrow{D_{new}} M_{A,\vec{\boldsymbol{\lambda}},K}. \quad (3)$$

Функция f должна обеспечивать достижение минимума потерь $\hat{R}$ при ограничениях на время выполнения T:

$$f \in \arg\min \hat{R}(T, D_{new});\ T(\hat{R}, D_{new}) \le T_{\max}. \quad (4)$$

Управление синтезом моделей МО

Общая схема процесса управления синтезом модели МО включает три блока: 1) наблю-

даемый объект, данные от которого поступают к двум подсистемам — подсистеме управления синтезом и подсистеме синтеза; 2) подсистему управления синтезом, формирующую управляющие воздействия; 3) подсистему синтеза, формирующую на основе полученных данных и управляющих воздействий модель МО. В общем виде предлагаемая схема управления представлена на рис. 1.

Обобщенная схема управления синтезом модели МО предполагает выполнение следующих шагов.

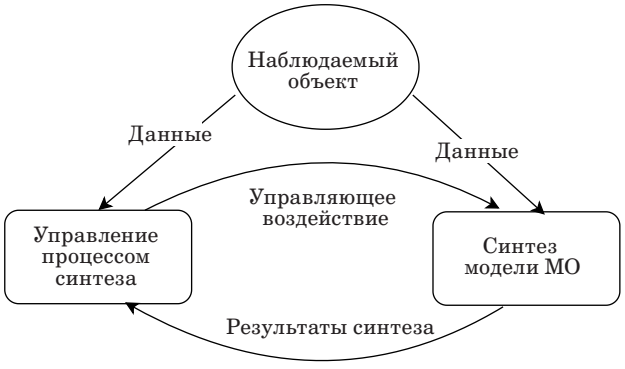
Шаг 1. При поступлении данных от наблюдаемого объекта подсистема управления процессом синтеза формирует управляющее воздействие на подсистему синтеза модели.

Шаг 2. Подсистема синтеза модели на основе управляющего воздействия и данных от наблюдаемого объекта синтезирует модель МО.

Шаг 3. Результаты применения модели передаются обратно в подсистему управления для оценки качества управляющего воздействия.

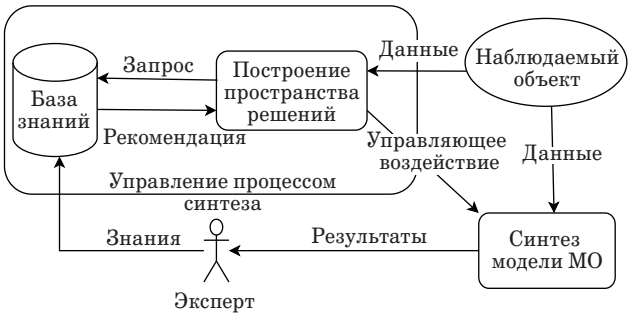
Подсистема управления процессом синтеза может быть реализована с помощью различных средств, в частности основана на использовании экспертных знаний, или, при наличии достаточно большой обучающей выборки ранее решенных задач, могут использоваться методы метаобучения.

При управлении процессом синтеза модели МО с помощью экспертных знаний реализуется схема управления, представленная на рис. 2. Подсистема управления процессом синтеза модели МО состоит из двух блоков: 1) базы знаний и 2) блока построения пространства решений. Эксперт на основе имеющихся у него знаний формирует базу знаний. Блок построения пространства решений отправляет запрос в базу знаний и на основе полученных от нее рекомендаций и данных, поступающих от наблюдаемого



■ ***Рис. 1.*** Обобщенная схема управления синтезом модели МО

■ ***Fig. 1.*** A generalized scheme of ML model synthesis control



■ ***Рис. 2.*** Управление процессом синтеза с использованием экспертных знаний

■ ***Fig. 2.*** A control of the synthesis process with the use of an expert knowledge

объекта, формирует управляющие воздействия. Подсистема синтеза строит модель МО на основе полученных данных и с учетом управляющих воздействий. Сформированная результирующая модель может быть проанализирована экспертом, который в случае необходимости вносит изменения в базу знаний.

Функция f для случая использования экспертных знаний при управлении процессом синтеза модели МО имеет вид

$$f_E : A \times I(K_E) \xrightarrow{D_{new}} M_{A,\bar{\lambda},K_E};$$

$$T(M_{A,\bar{\lambda},K_E}, \hat{R}) \le T_{\max}, \qquad (5)$$

где K_E — знания, основанные на экспертных знаниях E, используемые при формировании управляющего воздействия; $M_{A,\bar{\lambda},K_E}$ — модель, построенная с использованием знаний K_E.

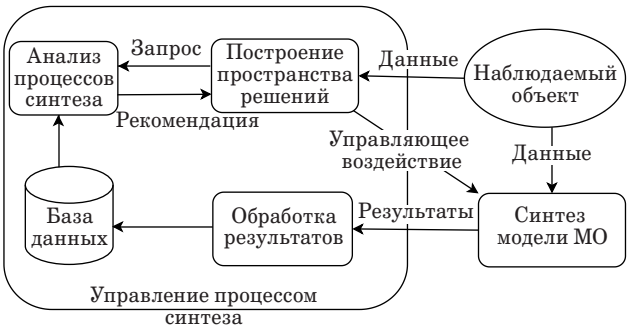
Процесс управления, основанный на экспертных знаниях, состоит из следующих шагов.

Шаг 1. Эксперт заполняет базу знаний на основе имеющегося у него опыта.

Шаг 2. Сформированная база знаний используется подсистемой управления процессом синтеза для выработки управляющих воздействий.

Шаг 3. Полученные в результате синтеза модели оцениваются экспертом. При необходимости эксперт вносит изменения в базу знаний.

При наличии накопленных данных о ранее решенных задачах управление синтезом может осуществляться на основе знаний, извлекаемых из накопленных данных, при этом для извлечения знаний могут использоваться различные методы, в частности методы метаобучения. Схема управления синтезом на основе знаний, извлекаемых из накопленных данных, представлена на рис. 3. Подсистема управления процессом синтеза модели МО в этом случае состоит из четырех блоков: 1) базы данных, в которой накапливаются данные о ранее решенных задачах; 2) блока анализа процессов синтеза, обрабатывающего



■ ***Рис. 3.*** Управление процессом синтеза на основе знаний, извлекаемых из данных о ранее решенных задачах

■ ***Fig. 3.*** A control of the synthesis process using the knowledge obtained from the data about previously solved tasks

данные о ранее построенных моделях МО для их последующего использования при выдаче рекомендаций; 3) блока построения пространства решений, отправляющего запрос блоку анализа и получающего от него рекомендации; 4) блока обработки результатов, преобразующего результаты решения новой задачи в форму, позволяющую их размещать в базе данных.

При такой схеме работы происходит постоянное автоматическое уточнение знаний, используемых при выдаче рекомендаций с каждым циклом обработки новых данных.

Функция f для случая использования знаний, автоматически извлекаемых из данных о ранее решенных задачах, имеет вид

$$f_{D_{prev}} : A \times I(K_{D_{prev}}) \xrightarrow{D_{new}} M_{A,\bar{\lambda},K_{D_{prev}}};$$

$$T(M_{A,\bar{\lambda},K_{D_{prev}}}, \hat{R}) \le T_{\max}, \qquad (6)$$

где D_{prev} – данные о ранее решенных задачах, используемые при формировании управляющего воздействия; $M_{A,\bar{\lambda},K_{D_{prev}}}$ – модель, построенная с использованием знаний, полученных при обработке D_{prev}.

Процесс управления, основанный на использовании автоматически извлекаемых знаний, состоит из следующих шагов.

Шаг 1. Блок анализа процессов синтеза получает информацию из базы данных о ранее решенных задачах, примененных алгоритмах и полученных результатах и использует эти данные для обучения.

Шаг 2. Блок построения пространства решений отправляет запрос к блоку анализа процессов и получает рекомендации по алгоритмам, которые могут обеспечить эффективное решение текущей задачи.

Шаг 3. На основе полученных рекомендаций формируется управляющее воздействие, которое передается в подсистему синтеза моделей.

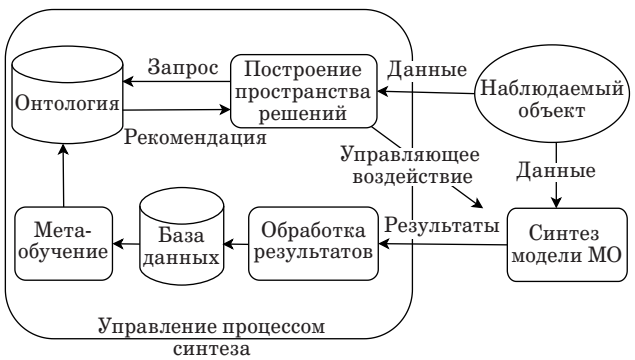
Шаг 4. Результаты синтеза передаются обратно в подсистему управления, где они обрабатываются и пополняют базу данных, которая далее используется в блоке анализа.

Частный алгоритм управления синтезом с применением метаобучения

В качестве частного алгоритма управления процессом синтеза, реализующего предложенный общий метаалгоритм, предлагается алгоритм, основанный на ограничении пространства поиска модели МО за счет метаобучения по накопленным данным и наполнения полученными знаниями базы знаний, представленной в форме онтологии. В этом случае подсистема управления процессом синтеза модели МО состоит из пяти блоков: 1) базы данных, содержащей информацию о ранее решенных задачах; 2) блока метаобучения, обрабатывающего обучающий набор из базы данных и формирующего на его основе базу знаний; 3) онтологии, хранящей выявленные взаимосвязи между метасвойствами обрабатываемых данных и примененными алгоритмами; 4) блока построения пространства решений, получающего данные от наблюдаемого объекта, отправляющего запрос к онтологии и формирующего на основе данных и полученных рекомендаций управляющее воздействие; 5) блока обработки результатов, преобразующего полученные результаты в форму, позволяющую их записывать в базу данных.

Схема управления синтезом с применением метаобучения представлена на рис. 4.

При применении метаобучения выполняется сравнение метасвойств данных, требующих обработки, и метасвойств ранее обработанных дан-



■ ***Рис. 4.*** Управление процессом синтеза на основе метаобучения и онтологии

■ ***Fig. 4.*** A control of the synthesis process using meta-learning and ontology

ных. В этом случае работа системы разбивается на два этапа: 1) этап предварительной подготовки и 2) этап работы с новыми данными.

На этапе предварительной подготовки выполняются следующие шаги.

Шаг 1. Задаются используемые метасвойства данных.

Шаг 2. Формируется база данных с обучающей выборкой, сформированной с учетом выбранных метасвойств.

Шаг 3. Обучающая выборка используется для метаобучения и формирования онтологии.

Этап работы с новыми данными состоит из следующих шагов.

Шаг 1. Блок построения пространства решений получает новые данные и выполняет запрос к онтологии, построенной на этапе предварительной подготовки.

Шаг 2. На основе полученных рекомендаций формируется управляющее воздействие на подсистему синтеза моделей.

Шаг 3. Подсистема синтеза моделей получает новые данные и управляющее воздействие и строит модель МО.

Шаг 4. Полученные результаты обрабатываются и пополняют базу данных новыми данными о решенной задаче.

При управлении синтезом модели МО с применением метаобучения и онтологии управление состоит в ограничении пространства поиска алгоритмов МО с применением правил, выявленных на основе обработки данных о ранее решенных задачах. Правила отображают взаимосвязи между метасвойствами данных и алгоритмами МО, подходящими для обработки данных с такими метасвойствами. Для построения правил возможно использование различных наборов метасвойств, ограничения на состав используемых метасвойств не накладываются. Определить метасвойства, формируемые для конкретного набора данных, можно с использованием метода выбора признаков [25] и др. Для ограничения пространства поиска определенными алгоритмами необходимо, чтобы применяемая AutoML система поддерживала использование подобного рода ограничений. К системам, позволяющим задавать ограничения на пространство поиска, относятся, в частности, Auto-sklearn и H2O.

Метаобучение для ограничения пространства поиска

Блоку метаобучения необходима обучающая выборка, на основе которой выявляются взаимосвязи между метасвойствами обрабатываемых данных и рекомендованными алгоритмами. При составлении обучающей выборки для метаобучения необходимо определить набор задач, которые будут решаться средствами AutoML. Для выбранных задач определяются значения метасвойств обрабатываемых данных и выполняется построение моделей МО. Полученные значения метасвойств и рекомендованные алгоритмы пополняют обучающую выборку. Процесс построения обучающей выборки приведен в алгоритме 1, где функция dataLoad используется для загрузки набора данных ds из репозитория, metaFeaturesExtract — для вычисления метасвойств данных data, а modelSearch производит поиск с помощью библиотеки AutoML модели МО, подходящей для обработки данных data за время, не превышающее T_{max}.

Алгоритм 1. createMetaDS(dsets, AutoML, T_{max}) — формирование обучающей выборки.

Входные данные: dsets — множество наборов данных; AutoML — используемая библиотека автоматизированного МО, T_{max} — максимальное время на поиск модели

Выходные данные: mfeat — набор метасвойств и рекомендуемых алгоритмов

```
1: mfeat = ∅
{Цикл для всех наборов данных ds из dsets}
2: repeat
3:    data = dataLoad(ds)
4:    meta = metaFeaturesExtract(data)
5:    algo = modelSearch (data, AutoML(T_max))
6:    mfeat = mfeat ∪ (algo ∪ meta)
7: until (ds ∈ dsets)
8: return mfeat
```

После получения обучающей выборки ее можно использовать для выявления связей между метасвойствами данных и рекомендованными для их обработки алгоритмами. Для этого строится дерево решений, в котором листьями являются выбранные алгоритмы, а узлами — используемые метасвойства. Переход от узла к левой ветке осуществляется в случае отсутствия данного метасвойства, а к правой — при его наличии. Полученный таким способом классификатор прост в интерпретации и может быть легко применен для добавления новых связей между элементами онтологии. Построение дерева решений и его использование для наполнения онтологии приведено в алгоритме 2, где функция treeCreation строит дерево решений на основе обучающей выборки mfeat, travelTree — рекурсивная функция, которая проходит по созданному дереву и формирует на его основе правила, размещаемые в онтологии.

Алгоритм 2. Построение дерева решений и наполнение онтологии правилами.

Входные данные: fileOnto — файл с онтологией, dsets — множество наборов данных; AutoML — используемая библиотека автомати-

зированного МО, T_{max} — максимальное время на поиск модели

Выходные данные: fileUpdated — файл, содержащий обновленную онтологию

1: mfeat = createMetaDS(dsets, AutoML, T_{max})
2: tree = treeCreation (mfeat)
3: fileUpdated = travelTree(tree, ∅, fileOnto)
4: **return** fileUpdated

Рекурсивный проход по дереву решений travelTree для записи всех путей и формирования на их основе правил для размещения в онтологии приведен в алгоритме 3, где функция getLeftNode возвращает левый подузел, getRightNode возвращает правый подузел, getFeature возвращает метасвойство, getValue возвращает значение, а extendOntology обновляет онтологию на основе файла с онтологией и сформированного пути.

Алгоритм 3. travelTree(node, list_of_nodes, file) — проход по дереву решений.

Входные данные: node — текущий узел, list_of_nodes — формируемый путь до листа, file — файл с онтологией

Выходные данные: file — файл с обновленной онтологией

1: left_node = getLeftNode(node)
2: right_node = getRightNode(node)
3: feature = getFeature(node)
4: **if** left_node == ∅ and right_node == ∅ **then**
5: list_of_nodes = list_of_nodes ∪ getValue(node)
6: file = extendOntology(file, list_of_nodes)
7: **end if**
8: **if** left_node != ∅ **then**
9: list_of_nodes = list_of_nodes ∪ (“No”+feature)
10: travelTree(left_node, list_of_nodes, file)
11: **end if**
12: **if** right_node != ∅ **then**
13: list_of_nodes = list_of_nodes ∪ feature
14: travelTree(right_node, list_of_nodes, file)
15: **end if**
16: **return** file

В алгоритме 3 осуществляется проход по дереву принятия решений, при этом происходит запись пути от корня до листа-алгоритма. При достижении листа весь сохраненный путь записывается в онтологию, содержащуюся в файле file, с помощью функции extendOntology, описанной в алгоритме 4, где функция getOntology читает онтологию из файла, pop извлекает из списка последний элемент (алгоритм), getAutoML читает из онтологии данные о AutoML системе, информация о которой будет дополнена, updateOntology обновляет онтологию для конкретного алгоритма и AutoML системы, а saveOntology сохраняет полученную онтологию в файл.

Алгоритм 4. extendOntology(file, list_of_nodes) — расширение онтологии.

Входные данные: file — файл с онтологией, list_of_nodes — путь до листа-алгоритма

Выходные данные: fileUpdated — файл с обновленной онтологией

1: onto = getOntology(file)
2: algo =pop(list_of_nodes)
3: info_to_add = getAutoML(onto)
{Цикл по всем узлам node в списке list_of_nodes}
4: **repeat**
5: info_to_add = info_to_add ∪ suitableFor(node)
6: **until** node ∈ list_of_nodes
7: onto = updateOntology(onto, algo, info_to_add)
8: fileUpdated = saveOntology(onto)
9: **return** fileUpdated

Вначале из сохраненного пути извлекается рекомендованный алгоритм, а затем все остальные части пути сохраняются как атрибуты данного алгоритма.

Обработка новых данных

При поступлении новых данных, для которых необходимо построить модель МО, вначале вычисляются значения их метасвойств, после этого на основе вычисленных метасвойств выполняется запрос к онтологии, возвращающей набор алгоритмов, которыми следует ограничить пространство поиска. Данные разбиваются на обучающую и тестовую выборки, и на основе обучающей выборки выполняется поиск модели МО, подходящей для обработки поступивших данных, с учетом ограниченного пространства поиска. После того как подходящая модель найдена, с помощью тестовой выборки вычисляются метрики качества полученной модели. Вычисленные метасвойства, информация о модели и значения ее метрик пополняют базу данных о решенных задачах, которая затем используется для обновления онтологии с помощью алгоритмов 3, 4. Процесс обработки новых данных приведен в алгоритме 5, где функция ontologyQuery с помощью метасвойств meta выполняет запрос к онтологии, записанной в файле fileOnto, функция splitData разбивает данные data на обучающую и тестовую выборки в соответствии с соотношением ratio, функция modelSearch производит поиск с помощью библиотеки AutoML модели МО, подходящей для обработки данных train_data за время, не превышающее T_{max} и с ограничением пространства поиска алгоритмами из множества algos_list, функция calcMetrics вычисляет значения метрик модели model с помощью тестовых данных test_data, функ-

ция updateDatabase обновляет базу данных database метасвойствами meta, информацией о модели model и ее метриках metrics, функция getDataFromDB извлекает из базы данных database данные mfeat, необходимые для обновления онтологии с использованием функции treeCreation.

Алгоритм 5. Обработка новых данных.

Входные данные: data — обрабатываемые данные; fileOnto — файл с онтологией, AutoML — используемая библиотека автоматизированного МО, ratio — соотношение разделения данных на обучающие и тестовые, T_{max} — максимальное время на поиск модели

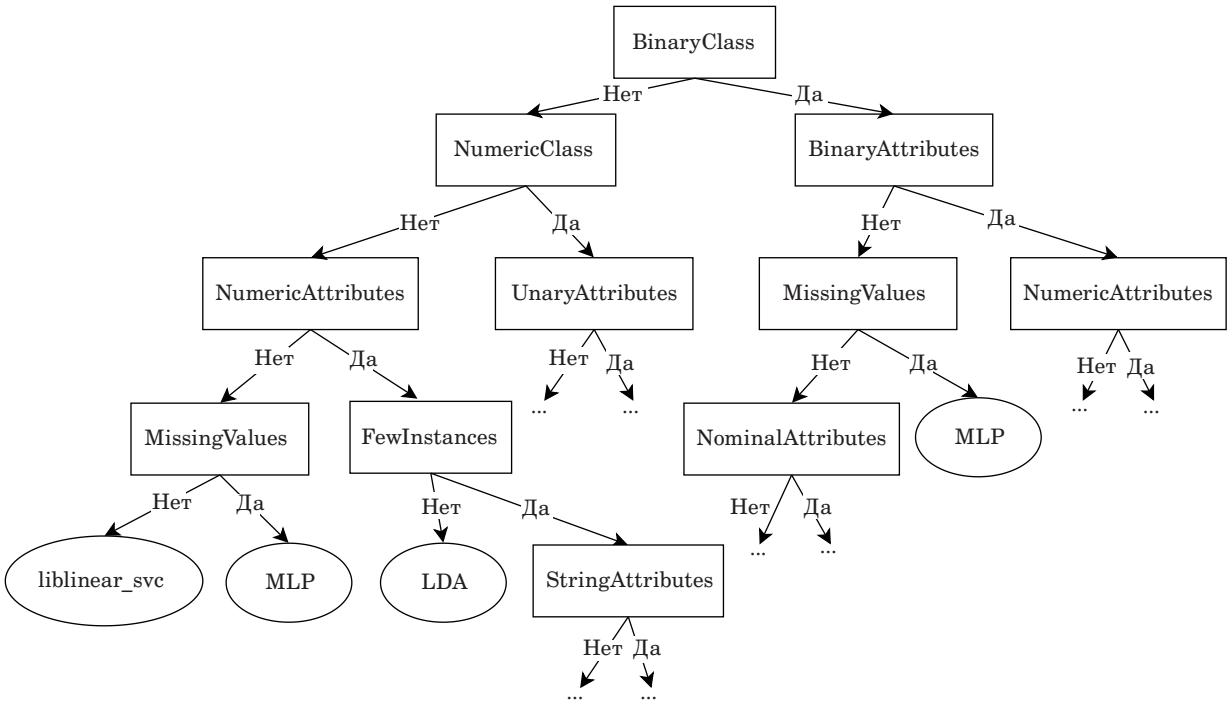
Выходные данные: fileUpdated — файл, содержащий обновленную онтологию

1: meta = metaFeaturesExtract(data)
2: algos_list = ontologyQuery(fileOnto, meta)
3: train_data = splitData(data, ratio)
4: test_data = data \ train_data
5: model = modelSearch (train_data, AutoML(algos_list, T_{max}))
6: metrics = calcMetrics(model, test_data)
7: database = updateDatabase(database, model, metrics, meta)
8: mfeat = getDataFromDB(database)
9: tree = treeCreation (mfeat)
10: fileUpdated = travelTree(tree, ∅, fileOnto)
11: **return** fileUpdated

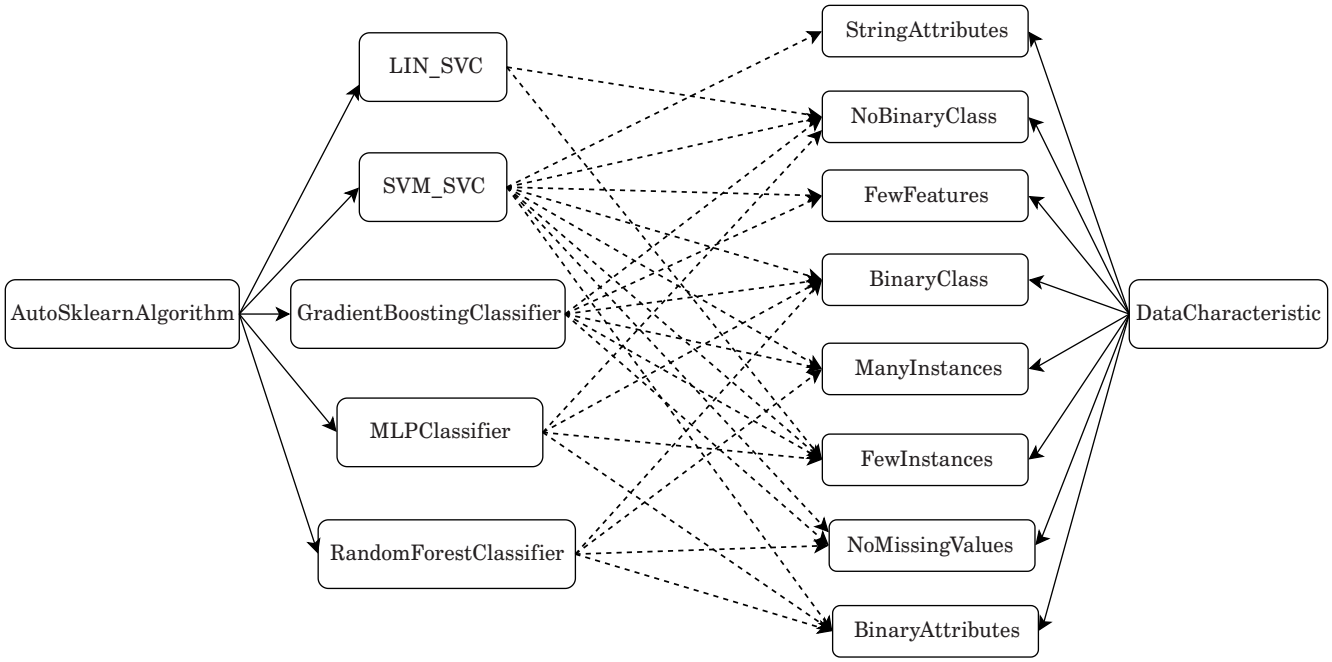
Экспериментальные исследования

Построение онтологии

При проведении экспериментальных исследований использовалась AutoML-система Auto-sklearn — решение, основанное на байесовской оптимизации и выполняющее поиск по 16 алгоритмам библиотеки алгоритмов МО Scikit-learn [28]. При формировании обучающей выборки для метаобучения и наполнения онтологии были использованы 35 наборов данных из открытого репозитория OpenML: letter, balance-scale, mfeat-fourier, mfeat-karhunen, mfeat-pixel, mfeat-zernike, breast-w, mushroom, optdigits, credit-approval, pendigits, sick, soybean, spambase, splice, tic-tac-toe, vehicle, waveform-5000, electricity, satimage, isolet, vowel, scene, monks-problems-1, JapaneseVowels, synthetic_control, irish, analcatdata_authorship, collins, ada_agnostic, gina_agnostic, KDDCup09_appetency, gas-drift, har, hill-valley. Загруженные наборы данных и библиотека Auto-sklearn были использованы для формирования обучающей выборки с помощью алгоритма 1 с вычислением следующих метасвойств: ManyInstances, FewInstances, ManyFeatures, FewFeatures, BinaryClass, NumericClass, NominalClass, StringClass, BinaryAttributes, MissingValues, NominalAttributes, NumericAttributes, UnaryAttributes, StringAttributes. Сформированная выборка была передана в алгоритм 2 для построения дерева решений. Часть сформированного дерева решений показана на рис. 5.



■ ***Рис. 5.*** Часть дерева решений, сформированного для выбора алгоритмов МО по метасвойствам данных
■ ***Fig. 5.*** A part of the decision tree created to determine a suitable ML algorithms based on meta-features of data



■ ***Рис. 6.*** Фрагмент онтологии, описывающей связи между алгоритмами МО и метасвойствами данных
■ ***Fig. 6.*** Part of the ontology describing the connection between algorithms and meta-features

С помощью редактора онтологий Protégé версии 5.6.4 построена онтология, включающая две независимые составляющие: 1) описания доступных в используемой AutoML-системе алгоритмов; 2) описания используемых метасвойств данных. Затем разработанная онтология и сформированное дерево решений использовались алгоритмами 3 и 4 для расширения онтологии [29] путем добавления связей между метасвойствами и алгоритмами МО. Фрагмент построенной таким образом онтологии приведен на рис. 6.

Синтез моделей МО
с использованием онтологии

Экспериментальные исследования были направлены на сравнительную оценку времени, требуемого для синтеза моделей МО с использованием онтологии и без, при этом также оценивались следующие показатели моделей: достоверность (accuracy), точность (precision), полнота (recall), F-мера (F-score) и площадь под ROC-кривой (AUC). Для проведения экспериментальных исследований использована онлайн-платформа Google Colab, предоставляющая виртуальный процессор Intel Xeon, RAM 13 ГБ, графический процессор (GPU) NVIDIA Tesla K80 с видеопамятью 16 ГБ и программное обеспечение Python v.3.10.12, Owlready2 v.2-0.47 и Auto-sklearn v.0.15. Для целей тестирования использовано 15 наборов данных из репозитория OpenML. Каждый тестовый набор данных был разбит на обучающую и тестовую выборки в соотношении 4:1. Над обучающими выборками выполнены следующие операции с использованием библиотеки Auto-sklearn:

1) поиск модели с ограничением времени поиска 15 мин;

2) поиск модели с ограничением времени поиска и ограничением точности;

3) поиск модели с ограничениями времени поиска, точности и пространства поиска. Для ограничения пространства поиска использовалась построенная онтология.

Для найденной в каждом случае модели были вычислены ее показатели, а также записаны количество времени, которое занял поиск, и рекомендованный алгоритм. Результаты проведенных экспериментов представлены в таблице.

Проведенные эксперименты показали, что среднее время поиска модели, ограниченного только временем работы 900 с, составило 896,44 с, среднее время поиска модели, ограниченного временем работы и точностью, составило 433,59 с, среднее время поиска модели, ограниченного временем работы, точностью и подмножеством алгоритмов, полученных путем запроса к онтологии по метасвойствам набора данных, составило 255,31 с. Таким образом, за счет сокращения пространства поиска с помощью запросов к онтологии время поиска было сокращено на 71,52 % по сравнению с поиском, ограниченным только временем работы, и на 41,12 % по сравнению с поиском, ограниченным временем и точностью. В восьми случаях из 15 алгоритм, предложен-

■ Метрики моделей, найденных в условиях ограничений на время поиска и (или) точности с использованием / без использования онтологии

■ Metrics of models found under time and/or accuracy constraints while limiting the search space with ontology and without

Набор данных	Ограничение по точности, %	Использование онтологии	Алгоритм	Достоверность	Точность	Полнота	F-мера	AUC	Время, с
Adult	Нет	Нет	gradient boosting	0,870	0,770	0,665	0,714	0,801	895,50
	0,14	Нет	gradient boosting	0,872	0,788	0,653	0,714	0,798	371,60
	0,14	Few Features	gradientboosting	0,872	0,788	0,653	0,714	0,798	286,34
Banking	Нет	Нет	gradientboosting	0,907	0,624	0,503	0,557	0,732	894,26
	0,09	Нет	gradient boosting	0,907	0,624	0,503	0,557	0,732	628,15
	0,09	Few Features	gradientboosting	0,907	0,624	0,503	0,557	0,732	227,43
Cars	Нет	Нет	gradientboosting	0,997	0,996	0,982	0,989	1	896,38
	0,01	Нет	mlp	0,991	0,990	0,960	0,974	0,999	115,97
	0,01	String Attributes	libsvm svc	0,986	0,965	0,974	0,969	0,999	18,273
Amazon	Нет	Нет	extra trees	0,950	0,953	0,996	0,974	0,566	904,52
	0,06	Нет	randomforest	0,947	0,949	0,997	0,972	0,527	30,168
	0,06	String Attributes	libsvm svc	0,946	0,946	1	0,972	0,5	27,777
Australian	Нет	Нет	ada boost	0,884	0,944	0,797	0,864	0,878	899,91
	0,15	Нет	ada boost	0,884	0,944	0,797	0,864	0,878	320,21
	0,15	Binary Class	random forest	0,877	0,912	0,813	0,860	0,872	103,82
blood	Нет	Нет	extra trees	0,787	0,68	0,415	0,515	0,671	896,01
	0,2	Нет	mlp	0,8	0,739	0,415	0,531	0,680	118,56
	0,2	Few Instances	mlp	0,8	0,739	0,415	0,531	0,680	112,55
kc1	Нет	Нет	mlp	0,879	0,613	0,328	0,427	0,647	895,46
	0,14	Нет	mlp	0,879	0,613	0,328	0,427	0,647	895,27
	0,14	Binary Class	mlp	0,870	0,543	0,328	0,409	0,642	726,62
christine	Нет	Нет	random forest	0,743	0,733	0,731	0,732	0,742	894,98
	0,26	Нет	random forest	0,743	0,733	0,731	0,732	0,742	284,20
	0,26	Many Instances	random forest	0,743	0,733	0,731	0,732	0,742	279,32
cnae9	Нет	Нет	sgd	0,926	0,929	0,922	0,923	0,996	894,19
	0,05	Нет	passive aggress	0,940	0,942	0,937	0,938	0,998	894,25
	0,05	Few Instances	liblinear svc	0,926	0,926	0,923	0,923	0,996	540,43
fabert	Нет	Нет	random forest	0,691	0,700	0,655	0,662	0,919	899,49
	0,32	Нет	random forest	0,691	0,700	0,655	0,662	0,919	41,934
	0,32	NoMissingValues	random forest	0,691	0,700	0,655	0,662	0,919	40,611
helena	Нет	Нет	k_nearestneighbors	0,179	0,089	0,086	0,086	0,539	895,39
	0,7	Нет	k_nearest_neighbors	0,179	0,089	0,086	0,086	0,539	895,49
	0,7	NoBinaryClass	liblinear svc	0,280	0,096	0,103	0,079	0,807	893,85
jannis	Нет	Нет	gradient boosting	0,712	0,638	0,543	0,559	0,872	893,60
	0,3	Нет	gradient boosting	0,712	0,638	0,543	0,559	0,872	223,33
	0,3	NoBinaryClass	gradient boosting	0,712	0,638	0,543	0,559	0,872	137,20
jasmine	Нет	Нет	random forest	0,812	0,751	0,940	0,835	0,811	895,76
	0,18	Нет	random forest	0,812	0,751	0,940	0,835	0,811	895,84
	0,18	Binary Attributes	random forest	0,814	0,753	0,940	0,837	0,813	97,186

■ *Окончание таблицы*
■ *The end of the Table*

Набор данных	Ограничение по точности, %	Использование онтологии	Алгоритм	Достоверность	Точность	Полнота	F-мера	AUC	Время, с
kr-vs-kp	Нет	Нет	gradient boosting	0,992	0,997	0,988	0,993	0,992	895,36
	0,01	Нет	gradient boosting	0,992	0,994	0,991	0,993	0,992	231,18
	0,01	String Attributes	libsvm svc	0,992	0,994	0,991	0,993	0,992	9,271
mfeat-factors	Нет	Нет	liblinear svc	0,965	0,968	0,965	0,966	0,999	895,85
	0,02	Нет	liblinear svc	0,965	0,968	0,965	0,966	0,999	557,75
	0,02	Few Instances	libsvm svc	0,965	0,967	0,965	0,966	0,993	329,04

ный поиском, ограниченным с использованием онтологии, был идентичен алгоритму, предложенному при поиске, ограниченном временем и точностью. Из этих восьми случаев в шести случаях значения метрик не изменились, в одном случае значения метрик, полученных с помощью поиска, ограниченного с использованием онтологии, улучшились, а в одном случае ухудшились. Изменение значений метрик при сохранении используемого алгоритма объясняется тем, что при ограничении пространства поиска меньшим количеством алгоритмов при поиске моделей выделяется больше времени на подбор гиперпараметров. Из семи случаев, когда поиском, ограниченным онтологией, был выбран другой алгоритм для построения модели, в одном случае улучшились значения достоверности (+56,79 %), точности (+7,11 %), полноты (+20,5 %), AUC (+49,91 %) и ухудшилось значение F-меры (–7,98 %), в двух случаях значения метрик не изменились, и в четырех случаях ухудшились значения достоверности (–0,73 %), точности (–1,99 %), F-меры (–0,65 %), AUC (–1,01 %) и улучшились значения полноты (+0,48 %). В среднем при поиске, ограниченном онтологией, улучшились значения достоверности (+0,54 %), полноты (+0,34 %) и AUC (+1,85%) и ухудшились средние значения точности (–1,21%) и F-меры (–0,45%). Ухудшение точности и F-меры может быть объяснено недостаточным объемом знаний в онтологии, используемой подсистемой управления синтезом моделей. При увеличении объемов обработанных данных и увеличении объема знаний можно ожидать улучшение значений метрик получаемых моделей МО.

Заключение

В рамках проведенного исследования был разработан общий метаалгоритм управления процессами синтеза моделей МО. Метаалгоритм обеспечил возможность использования базы знаний, наполняемой из различных источников, включая экспертные знания и знания, получаемые с применением существующих систем автоматизированного МО, для сокращения пространства поиска алгоритмов при синтезе моделей МО. Предложен частный алгоритм управления процессами синтеза на основе метаобучения и построения онтологии, реализующий общий метаалгоритм, который, как показали экспериментальные исследования, позволил ускорить поиск моделей в среднем на 41,12 % и улучшить средние значения трех из пяти метрик результирующих моделей.

Дальнейшая работа будет направлена на разработку других частных алгоритмов управления процессами синтеза моделей МО, определяемых в рамках общего метаалгоритма, на повышение эффективности предложенного частного алгоритма за счет использования дополнительных метасвойств и на реализацию предложенного частного алгоритма с использованием других систем автоматизированного МО.

Финансовая поддержка

Исследование выполнено при поддержке государственного бюджета, номер проекта FFZF-2025-0019.

Литература

1. **Горюнов М. Н., Мацкевич А. Г., Рыболовлев Д. А.** Синтез модели машинного обучения для обнаружения компьютерных атак на основе набора данных CICIDS2017. *Труды Института системного программирования РАН*, 2020, № 32, с. 81–94. doi:10.15514/ISPRAS–2020–32(5)–6
2. **Pei Y.** A comparative study of machine learning and automatic machine learning models for facial mask

recognition. *8th International Conference on Computer and Communication Systems (ICCCS)*, 2023, pp. 1047–1051. doi:10.1109/ICCCS57501.2023.10151333

3. **Kovalevsky V., Stankova E., Zhukova N., Ogiy O., Tristanov A.** *AutoML Framework for Labor Potential Modeling*. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Science and Business Media Deutschland GmbH, 2023, pp. 87–98. doi:10.1007/978-3-031-36808-0_6
4. **Baratchi M., Wang C., Limmer S., van Rijn J. N., Hoos H., Bäck T., Olhofer M.** Automated machine learning: Past, present and future. *Artificial Intelligence Review*, 2024, no. 57. doi:10.1007/s10462-024-10726-1
5. **Salehin I., Islam Md. S., Saha P., Noman S. M., Tuni A., Hasan Md. M., Baten Md. A.** AutoML: A systematic review on automated machine learning with neural architecture search. *Journal of Information and Intelligence*, 2024, no. 2, pp. 52–81. doi:10.1016/j.jiixd.2023.10.002
6. **Попова И. А., Ревунков Г. И., Гапанюк Ю. Е.** AutoML: исследование существующих программных реализаций и определение общей внутренней структуры решений. *Труды Института системного анализа Российской академии наук*, 2023, № 73, с. 43–54. doi:10.14357/20790279230106
7. **Radzi S. F. M., Karim M. K. A., Saripan M. I., Rahman M. A. A., Isa I. N. C., Ibahim M. J.** Hyperparameter tuning and pipeline optimization via Grid Search Method and Tree-Based AutoML in breast cancer prediction. *Journal of Personalized Medicine*, 2021, no. 11. doi:10.3390/jpm11100978
8. **Pokhrel P., Lazar A.** A comparison of AutoML hyperparameter optimization tools for tabular data. *The International FLAIRS Conference Proceedings*, 2023, no. 36. doi:10.32473/flairs.36.133357
9. **Karras A., Karras C., Schizas N., Avlonitis M., Sioutas S.** AutoML with bayesian optimizations for big data management. *Information*, 2023, no. 14. doi:10.3390/info14040223
10. **Pomsuwan T., Freitas A. A.** Genetic algorithm-based Auto-ML system for survival analysis. *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, ACM, New York, NY, USA, 2024, pp. 370–377. doi:10.1145/3605098.3635954
11. **Thornton C., Hutter F., Hoos H. H., Leyton-Brown K.** Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms. *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 2013, pp. 847–855. doi:10.1145/2487575.2487629
12. **Feurer M., Eggensperger K., Falkner S., Lindauer M., Hutter F.** Auto-Sklearn 2.0: Hands-free AutoML via meta-learning. *Journal of Machine Learning Research*, 2022, no. 23, pp. 1–61. doi:10.5555/3586589.3586850
13. **Olson R. S., Moore J. H.** *TPOT: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning*. Automated Machine Learning: Methods, Systems, Challenges/ F. Hutter, L. Kotthoff and J. Vanschoren (Eds.). Springer International Publishing, Cham, 2019, pp. 151–160. doi:10.1007/978-3-030-05318-5_8
14. **LeDell E., Poirier S.** H2O AutoML: Scalable automatic machine learning. *7th ICML Workshop on Automated Machine Learning (AutoML)*, 2020, pp. 1–16.
15. **Jin H., Chollet F., Song Q., Hu X.** AutoKeras: An AutoML library for deep learning. *Journal of Machine Learning Research*, 2023, no. 24, pp. 1–6. doi:10.5555/3648699.3648705
16. **Jose S. T., Simeone O., Durisi G.** Transfer meta-learning: Information-theoretic bounds and information meta-risk minimization. *IEEE Transactions on Information Theory*, 2022, no. 68, pp. 474–501. doi:10.1109/TIT.2021.3119605
17. **Khan I., Zhang X., Rehman M., Ali R.** A literature survey and empirical study of meta-learning for classifier selection. *IEEE Access*, 2020, no. 8, pp. 10262–10281. doi:10.1109/ACCESS.2020.2964726
18. **Gomaa I., Mokhtar M. O. H., El-Tazi N., Zidane A.** SML-AutoML: A smart meta-learning automated machine learning framework. *Advances in Artificial Intelligence and Machine Learning*, 2024, no. 04, pp. 3071–3096. doi:10.54364/AAIML.2024.44176
19. **Kotlar M., Punt M., Radivojevic Z., Cvetanovic M., Milutinovic V.** Novel meta-features for automated machine learning model selection in anomaly detection. *IEEE Access*, 2021, no. 9, pp. 89675–89687. doi:10.1109/ACCESS.2021.3090936
20. **Dyrmishi S., Elshawi R., Sakr S.** A decision support framework for AutoML systems: A meta-learning approach. *International Conference on Data Mining Workshops (ICDMW)*, 2019, pp. 97–106. doi:10.1109/ICDMW.2019.00025
21. **Li Y., Shen Y., Jiang H., Bai T., Zhang W., Zhang C., Cui B.** Transfer learning based search space design for hyperparameter tuning. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, 2022, pp. 967–977. doi:10.1145/3534678.3539369
22. **Wang A., Zhang K., Wu H., Chen H., Wang M.** Meta-learning-integrated neural architecture search for few-shot hyperspectral image classification. *Electronics (Basel)*, 2025, no. 14. doi:10.3390/electronics14152952
23. **Humm B. G., Zender A.** An ontology-based concept for meta AutoML. *17th IFIP International Conference on Artificial Intelligence Applications and Innovations*, 2021, pp. 117–128. doi:10.1007/978-3-030-79150-6_10
24. **Aliev M. R., Baimuratov I. R.** Automation of machine learning pipeline design by an ontology as an integrative meta-learning model. *The XIII Majorov International Conference on Software Engineering and Computer Systems (MICSECS)*, 2021.
25. **Nayak A., Božić B., Longo L.** *An Ontological Approach for Recommending a Feature Selection Algorithm*. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and

Lecture Notes in Bioinformatics). Springer Science and Business Media Deutschland GmbH, 2022, pp. 300–314. doi:10.1007/978-3-031-09917-5_20
26. **Kallab L., Mansour E., Chbeir R.** SML: Semantic machine learning model ontology. *24th International Conference Web Information Systems Engineering (WISE)*, 2023, pp. 896–911. doi:10.1007/978-981-99-7254-8_70
27. **Klironomos A.** Towards democratized machine learning: A semantic web approach. *Companion Proceedings of the ACM on Web Conference*, ACM, New York, NY, USA, 2025, pp. 697–700. doi:10.1145/3701716.3715280
28. **Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay É.** Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011, no. 12, pp. 2825–2830. doi:10.5555/1953048.2078195
29. **Ковалевский В. Э., Ман Т., Игнатов Д. И., Жукова Н. А., Куликов И. А.** *Auto-Onto — платформа автоматизированного машинного обучения с использованием онтологий*. https://github.com/DarkEol/AutoML/tree/main/AutoML-Ontology (дата обращения: 08.05.2025).

UDC 004.852
doi:10.31799/1684-8853-2025-6-28-41
EDN: KKSXFO

Meta-algorithm for the process control of complex machine learning model synthesis

N. A. Zhukova[a], Dr. Sc., Tech., Associate Professor, ORCID.ORG/0000-0001-5877-4461, nazhukova@mail.ru
V. E. Kovalevsky[a], Junior Researcher, orcid.org/0000-0002-0414-906X
[a]St. Petersburg Federal Research Center of the Russian Academy of Sciences, 39, 14th Line V.O., 199178, Saint-Petersburg, Russian Federation

Introduction: Automated machine learning methods allow automating synthesis of machine learning models adapted to specific data processing. However, these methods require significant time and computational costs. **Purpose:** To develop a meta-algorithm for the process control of synthesis of machine learning models, that would reduce the computational complexity of automated synthesis of machine learning models. **Results:** We propose a general meta-algorithm for the process control of synthesis of complex machine learning models and a specific algorithm that allows limiting the search space through meta-learning. The proposed specific algorithm is based on using meta-features of data and an ontology that contains rules for selecting machine learning algorithms depending on the meta-features of the processed data. The ontology is constructed by pre-processing the results of previously synthesized machine learning models. In addition, for the specific algorithm, we develop an algorithm for building a training set and an algorithm for constructing an ontology to reduce the search space. The experiments have shown that the use of the proposed specific algorithm reduces the time of the synthesis of machine learning models by 41.12%. Moreover, the obtained models have increased accuracy (+0.54%), recall (+0.34%) and AUC (+1.85%). **Practical relevance:** The specific algorithm developed on the base of the meta-algorithm allows reducing the computational complexity of the process of automated machine learning model synthesis and enables the application of machine learning in subject areas that require prompt construction and adaptation of machine learning models to new data and tasks.

Keywords — automated machine learning, synthesis of machine learning models, AutoML, meta-learning, ontologies for AutoML, control of machine learning model synthesis.

For citation: Zhukova N. A., Kovalevsky V. E. Meta-algorithm for the process control of complex machine learning model synthesis. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 28–41 (In Russian). doi:10.31799/1684-8853-2025-6-28-41, EDN: KKSXFO

Financial support
This work was supported by the state budget, project No. FFZF-2025-0019.

References

1. Goryunov M. N., Matskevich A. G., Rybolovlev D. A. Synthesis of a machine learning model for detecting computer attacks based on the CICIDS2017 dataset. *Proceedings of the Institute for System Programming of the RAS*, 2020, no. 32, pp. 81–94 (In Russian). doi:10.15514/ISPRAS–2020–32(5)–6
2. Pei Y. A comparative study of machine learning and automatic machine learning models for facial mask recognition. *8th International Conference on Computer and Communication Systems (ICCCS)*, 2023, pp. 1047–1051. doi:10.1109/ICCCS57501.2023.10151333
3. Kovalevsky V., Stankova E., Zhukova N., Ogiy O., Tristanov A. *AutoML framework for labor potential modeling*. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Science and Business Media Deutschland GmbH, 2023, pp. 87–98. doi:10.1007/978-3-031-36808-0_6
4. Baratchi M., Wang C., Limmer S., van Rijn J. N., Hoos H., Bäck T., Olhofer M. Automated machine learning: Past, present and future. *Artificial Intelligence Review*, 2024, no. 57. doi:10.1007/s10462-024-10726-1
5. Salehin I., Islam Md. S., Saha P., Noman S. M., Tuni A., Hasan Md. M., Baten Md. A. AutoML: A systematic review on automated machine learning with neural architecture search. *Journal of Information and Intelligence*, 2024, no. 2, pp. 52–81. doi:10.1016/j.jiixd.2023.10.002
6. Popova I. A., Revunkov G. I., Gapanyuk Y. E. AutoML: Examining existing software implementations and determining the overall internal structure of solutions. *Proceeding of the Institute for Systems Analysis of the Russian Academy of Science*, 2023, no. 73, pp. 43–54 (In Russian). doi:10.14357/20790279230106
7. Radzi S. F. M., Karim M. K. A., Saripan M. I., Rahman M. A. A., Isa I. N. C., Ibahim M. J. Hyperparameter tuning and pipeline optimization via Grid Search Method and Tree-Based AutoML in breast cancer prediction. *Journal of Personalized Medicine*, 2021, no. 11. doi:10.3390/jpm11100978

8. Pokhrel P., Lazar A. A comparison of AutoML hyperparameter optimization tools for tabular data. *The International FLAIRS Conference Proceedings*, 2023, no. 36. doi:10.32473/flairs.36.133357
9. Karras A., Karras C., Schizas N., Avlonitis M., Sioutas S. AutoML with bayesian optimizations for big data management. *Information*, 2023, no. 14. doi:10.3390/info14040223
10. Pomsuwan T., Freitas A. A. Genetic algorithm-based Auto-ML system for survival analysis. *Proceedings of the 39th ACM/SIGAPP Symposium on Applied Computing*, ACM, New York, NY, USA, 2024, pp. 370–377. doi:10.1145/3605098.3635954
11. Thornton C., Hutter F., Hoos H. H., Leyton-Brown K. Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms. *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, 2013, pp. 847–855. doi:10.1145/2487575.2487629
12. Feurer M., Eggensperger K., Falkner S., Lindauer M., Hutter F. Auto-sklearn 2.0: Hands-free AutoML via meta-learning. *Journal of Machine Learning Research*, 2022, no. 23, pp. 1–61. doi:10.5555/3586589.3586850
13. Olson R. S., Moore J. H. *TPOT: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning*. In: *Automated Machine Learning: Methods, Systems, Challenges*. F. Hutter, L. Kotthoff and J. Vanschoren (Eds.). Springer International Publishing, Cham, 2019, pp. 151–160. doi:10.1007/978-3-030-05318-5_8
14. LeDell E., Poirier S. H2O AutoML: Scalable automatic machine learning. *7th ICML Workshop on Automated Machine Learning (AutoML)*, 2020, pp. 1–16.
15. Jin H., Chollet F., Song Q., Hu X. AutoKeras: An AutoML library for deep learning. *Journal of Machine Learning Research*, 2023, no. 24, pp. 1–6. doi:10.5555/3648699.3648705
16. Jose S. T., Simeone O., Durisi G. Transfer meta-learning: Information-theoretic bounds and information meta-risk minimization. *IEEE Transactions on Information Theory*, 2022, no. 68, pp. 474–501. doi:10.1109/TIT.2021.3119605
17. Khan I., Zhang X., Rehman M., Ali R. A literature survey and empirical study of meta-learning for classifier selection. *IEEE Access*, 2020, no. 8, pp. 10262–10281. doi:10.1109/ACCESS.2020.2964726
18. Gomaa I., Mokhtar M. O. H., El-Tazi N., Zidane A. SML-AutoML: A smart meta-learning automated machine learning framework. *Advances in Artificial Intelligence and Machine Learning*, 2024, no. 04, pp. 3071–3096. doi:10.54364/AAIML.2024.44176
19. Kotlar M., Punt M., Radivojevic Z., Cvetanovic M., Milutinovic V. Novel meta-features for automated machine learning model selection in anomaly detection. *IEEE Access*, 2021, no. 9, pp. 89675–89687. doi:10.1109/ACCESS.2021.3090936
20. Dyrmishi S., Elshawi R., Sakr S. A decision support framework for AutoML systems: A meta-learning approach. *International Conference on Data Mining Workshops (ICDMW)*, 2019, pp. 97–106. doi:10.1109/ICDMW.2019.00025
21. Li Y., Shen Y., Jiang H., Bai T., Zhang W., Zhang C., Cui B. Transfer learning based search space design for hyperparameter tuning. *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ACM, New York, NY, USA, 2022, pp. 967–977. doi:10.1145/3534678.3539369
22. Wang A., Zhang K., Wu H., Chen H., Wang M. Meta-learning-integrated neural architecture search for few-shot hyperspectral image classification. *Electronics (Basel)*, 2025, no. 14. doi:10.3390/electronics14152952
23. Humm B. G., Zender A. An ontology-based concept for meta AutoML. *17th IFIP International Conference on Artificial Intelligence Applications and Innovations*, 2021, pp. 117–128. doi:10.1007/978-3-030-79150-6_10
24. Aliev M. R., Baimuratov I. R. Automation of machine learning pipeline design by an ontology as an integrative meta-learning model. *The XIII Majorov International Conference on Software Engineering and Computer Systems (MICSECS)*, 2021.
25. Nayak A., Božić B., Longo L. *An Ontological Approach for Recommending a Feature Selection Algorithm*. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Science and Business Media Deutschland GmbH, 2022, pp. 300–314. doi:10.1007/978-3-031-09917-5_20
26. Kallab L., Mansour E., Chbeir R. SML: Semantic machine learning model ontology. *24th International Conference Web Information Systems Engineering (WISE)*, 2023, pp. 896–911. doi:10.1007/978-981-99-7254-8_70
27. Klironomos A. Towards democratized machine learning: A semantic web approach. *Companion Proceedings of the ACM on Web Conference 2025*, ACM, New York, NY, USA, 2025, pp. 697–700. doi:10.1145/3701716.3715280
28. Pedregosa F., Varoquaux G., Gramfort A., Michel V., Thirion B., Grisel O., Blondel M., Prettenhofer P., Weiss R., Dubourg V., Vanderplas J., Passos A., Cournapeau D., Brucher M., Perrot M., Duchesnay É. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 2011, no. 12, pp. 2825–2830. doi:10.5555/1953048.2078195
29. Kovalevsky V., Man T., Ignatov D., Zhukova N., Kulikov I. *Auto-Onto — platforma avtomatizirovannogo mashinnogo obuchenie s ispol'zovaniem ontologij* [Auto-Onto — A Framework for AutoML extended by use of ontologies]. 2024. Available at: https://github.com/DarkEol/AutoML/tree/main/AutoML-Ontology (accessed 8 May 2025).

УДК 004.032
doi:10.31799/1684-8853-2025-6-42-50
EDN: EGLAUQ

Научные статьи
Articles

Оценка характеристик модели распределенных транзакционных приложений с микросервисной архитектурой и параллельными узлами

А. В. Горбунова[a], *канд. физ.-мат. наук, старший научный сотрудник, orcid.org/0000-0002-9183-0426, avgorbunova@list.ru*
[a]*Институт проблем управления им. В. А. Трапезникова РАН, Профсоюзная ул., 65, Москва, 117997, РФ*

Введение: *микросервисная архитектура, позволяющая создавать приложения как набор независимых микросервисов для совместной работы над выполнением некоторого общего клиентского запроса, стала в последнее время основой, или даже стандартом, для развертывания сложных систем, затрагивающих множество физических структур и устройств. Кроме того, внедрение в подобные системы, особенно системы с высокой загрузкой, параллельные сценарии обслуживания, позволяет повысить их эффективность и производительность.* ***Цель:*** *разработать математическую модель распределенных транзакционных приложений с микросервисной архитектурой и параллельными узлами и оценить такой показатель ее функционирования, как среднее время отклика.* ***Результаты:*** *представлена математическая модель распределенных транзакционных приложений с микросервисной архитектурой в виде сети массового обслуживания с последовательными узлами, один из которых имеет параллельную структуру с несколькими подузлами, число которых больше двух. На основе метода декомпозиции для анализа сетей массового обслуживания предлагается подход к оценке среднего времени отклика рассматриваемой системы с использованием известных результатов для оценки отдельных узлов сети типа G/G/1, а также узлов с разделением и параллельным обслуживанием. Результаты вычислительных экспериментов позволяют сделать выводы о допустимости использования предложенного подхода, а также получить рекомендации относительно применимости формул для различных уровней загруженности системы, в частности тех, для которых средняя погрешность аппроксимации не превышает 10 %.* ***Практическая значимость:*** *предложенная в работе модель и метод ее исследования могут быть использованы для первичной оценки и прогнозирования среднего времени отклика транзакционных приложений с параллельными узлами при разных уровнях загруженности системы и, как следствие, способствовать поддержанию необходимого качества обслуживания пользователей транзакционных приложений.*

Ключевые слова *— транзакционные приложения, микросервисная архитектура, распределенные системы, параллельные операции, сеть массового обслуживания, среднее время отклика, fork-join, G/G/1.*

Для цитирования: Горбунова А. В. Оценка характеристик модели распределенных транзакционных приложений с микросервисной архитектурой и параллельными узлами. *Информационно-управляющие системы*, 2025, № 6, с. 42–50. doi:10.31799/1684-8853-2025-6-42-50, EDN: EGLAUQ
For citation: Gorbunova A. V. Evaluation of the characteristics of a distributed transactional application model with microservice architecture and fork-join structures. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 42–50 (In Russian). doi:10.31799/1684-8853-2025-6-42-50, EDN: EGLAUQ

Введение

Транзакционные приложения представляют собой системы, обслуживающие (оказывающие) транзакционные услуги, под которыми в большинстве случаев подразумевается обработка финансовых или коммерческих операций. Примерами транзакционных приложений могут являться системы 1) онлайн-банкинга, 2) электронной коммерции, 3) бронирования билетов и другие сервисы, связанные с транзакциями. В целом, речь идет о системах, управляющих большим потоком транзакций и имеющих, как правило, распределенную базу данных, поскольку традиционные базы данных с реляционной архитектурой в такой ситуации оказываются недостаточно продуктивными [1–3]. При этом под транзакционными услугами, соответственно, понимаются в первом случае операции по управлению пользователем своими счетами, а именно перевод денежных средств, оплата различного рода счетов, обмен валюты и т. п., во втором случае — операции, связанные с онлайн-торговлей товарами или услугами, а именно прием и обработка заказов, проверка статуса заказа и т. п., в третьем случае — это операции, связанные с продажей билетов, оплатой их стоимости и т. д. То есть это операции с базами данных, которые обслуживают рабочий процесс и могут включать в себя создание, удаление или изменение данных.

Естественно, что для систем транзакционных услуг важную роль играет именно производительность и, возможно, даже большую по сравнению, например, с интерактивностью (активным взаимодействием пользователя с системой), как

в случае с платформами, организующими видеоконференции, онлайн-игры или онлайн-чаты, позволяющими пользователям общаться в реальном времени через интернет. Особенно это касается высоконагруженных систем.

В таких условиях предпочтительным выбором для большинства поставщиков услуг становится микросервисная архитектура [1–5]. В отличие от монолитной архитектуры, являющейся структурой со связанными в единое целое компонентами, микросервисная архитектура представляет собой систему, состоящую из отдельных структурных элементов — микросервисов, которые могут иметь свои собственные базы данных. При этом предполагается, что клиент, направляя свой запрос в систему, может инициировать процедуру одновременного обращения к нескольким микросервисам с собственной базой данных, в каждую из которых требуется внести необходимые изменения. В этом случае говорят о распределенных транзакциях [6]. Например, процесс перевода денежных средств может задействовать два микросервиса: один списывает средства со счета отправителя, а второй зачисляет их на нужный расчетный счет получателя [1, 7, 8].

Использование микросервисной архитектуры имеет свои преимущества и недостатки [4, 5]. В частности, разбиение сложной архитектуры на более простые и независимые элементы облегчает добавление новых микросервисов и обеспечивает масштабируемость, однако усложняет координацию транзакций; распределение по различным узлам, в том числе и параллельным, снижает общую нагрузку на систему и повышает производительность, уменьшая время отклика системы, но при этом порождает сложности с синхронизацией и согласованностью данных [4, 5, 9].

Несмотря на неизбежно сопутствующие трудности, характерные для распределенных систем, опыт внедрения описанных технологий для организации рабочих процессов управления транзакциями (финансовыми транзакциями) на примере платформы PayPal является довольно успешным [10, 11].

Таким образом, на первый план выходит необходимость адекватно прогнозировать показатели производительности систем транзакционных услуг при меняющейся рабочей нагрузке, что в свою очередь позволит оценить ее надежность, а также повысить запас прочности и заложить потенциал адаптации к меняющимся условиям внешней среды и требованиям пользователей.

Традиционно для оценки характеристик качества функционирования различных телекоммуникационных систем используются инструменты теории массового обслуживания. Так, в работе [12] предлагается использовать сети Джексона для математического моделирования и первичного анализа систем транзакционных услуг, содержащих параллельные узлы, а для более сложных вариантов распределений (неэкспоненциальных) предлагается использовать имитационное моделирование. В статье [13] для исследования характеристик рабочих процессов транзакционных услуг рассматриваются математические модели с узлами более сложной архитектуры (G/G/1), однако без учета параллелизма.

В данной статье предлагается математическая модель для оценки среднего времени отклика систем последовательных транзакционных услуг с микросервисной архитектурой, содержащей параллельные узлы в виде сети массового обслуживания с линейной топологией, некоторые узлы которой представляют собой так называемые fork-join-структуры. При этом рассматривается случай, когда время обслуживания характеризуется распределением Парето, а входящий в систему поток является пуассоновским. Несмотря на то, что в работе исследуется частный случай подобной системы, предложенный метод для оценки характеристик модели позволяет распространить его и на другие варианты вероятностных распределений для интервалов времени между поступлениями очередных запросов и длительностей интервалов времени их обслуживания.

Для определения характеристик узлов непараллельной структуры сетей линейной топологии используется несколько вариантов аппроксимаций, представляющих собой известные классические результаты. В целом же предполагается обращение к методу декомпозиции — оценке показателей производительности каждого узла системы по отдельности с дальнейшим использованием полученных результатов для оценки характеристик сети в итоге.

Для оценки же характеристик параллельных узлов (fork-join) применяется комплексный подход, включающий методы интеллектуального анализа данных, который позволяет получить хорошее качество приближения для аналитического выражения.

Fork-join-структура представляет собой узел, при поступлении на который запрос разделяется на подзапросы, каждый из которых направляется на обслуживание в отдельный подузел, причем время обслуживания всего запроса является максимумом из всех времен пребывания подзапросов в своих подузлах. За счет такой организации обслуживания запроса повышается производительность по сравнению с последовательным выполнением операций, при этом выигрыш во времени получается значительным. Поэтому подобные структуры довольно популярны, даже несмотря на сложности, связанные с техниче-

ской реализацией корректного процесса разбиения запроса на более мелкие задачи.

Математическая модель системы транзакционных услуг с параллельными узлами

Итак, рассмотрим распределенное транзакционное приложение. Поскольку предполагается, что возможно одновременное обращение к нескольким микросервисам, то оно будет содержать параллельный узел, который будет моделироваться с помощью fork-join-системы массового обслуживания, содержащей K подузлов. Переход рабочего процесса к следующему узлу будет означать завершение всех необходимых операций на каждом из микросервисов системы данного узла.

Допустим, что входящий в систему поток является пуассоновским, причем средняя длительность интервала между соседними поступлениями требований равна $1/\lambda$, а длительность интервала обслуживания на приборе как каждого параллельного подузла, так и каждого следующего узла системы имеет распределение Парето со следующей функцией распределения:

$$B_{Pa}(t) = 1 - \left(\frac{\alpha - 1}{\alpha}\frac{1}{t}\right)^{\alpha}, \; t \geq \frac{\alpha - 1}{\alpha} \qquad (1)$$

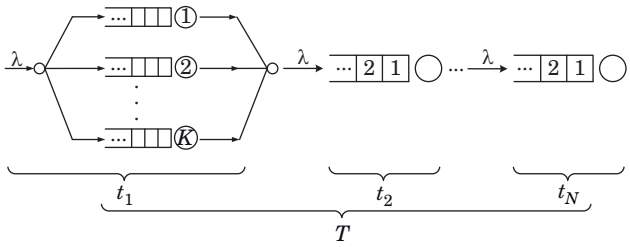
со средним значением $b_{Pa} = 1$, вторым моментом $b_{Pa}^{(2)} = \dfrac{(\alpha - 1)^2}{\alpha(\alpha - 2)}$ и параметром $\alpha > 3$.

Выбор распределения Парето, т. е. степенного распределения, обуславливается распространенностью его применения для моделирования процессов, протекающих в различного рода сетях передачи данных. Что касается пуассоновского входящего потока, то, несмотря на некоторые его ограничения, он все еще используется для построения моделей и первичной оценки параметров инфокоммуникационных систем [14–18].

После прохождения параллельного узла рабочий процесс последовательно переходит от одного узла системы к другому до N-го узла включительно, пока не завершит все необходимые операции по обслуживанию запроса пользователя системы.

Схема описанной модели функционирования приложения (рис. 1) представляет собой сеть массового обслуживания линейной архитектуры, но с параллельным узлом.

Одним из основных методов исследования сетей массового обслуживания является метод декомпозиции, который предполагает анализ отдельных фрагментов сети изолированно.



■ ***Рис. 1.*** Схема модели транзакционного приложения с микросервисной архитектурой и параллельным узлом

■ ***Fig. 1.*** Transactional application model diagram with microservice architecture and parallel node

Причем под фрагментом иногда может подразумеваться не только один узел, но и некоторая их совокупность.

Такой подход позволяет получить точные аналитические решения лишь для ограниченного класса сетей, к которым относятся открытые экспоненциальные сети Джексона и некоторые их расширения. В остальных же случаях, которых существенное большинство, данный способ предполагает приближенный анализ. При этом, разумеется, нельзя не отметить, что в отдельных ситуациях точный аналитический подход все-таки возможен, но из-за высокой размерности пространства состояний исследуемых систем, как правило, является нерациональным.

Оценка среднего времени отклика системы

Одним из наиболее важных показателей производительности системы является ее среднее время отклика. Корректная оценка этой характеристики важна для провайдеров (поставщиков услуг) в связи с необходимостью соблюдения соглашения о качестве оказываемых ими услуг (Quality of Service, QoS). Кроме того, на основе полученных оценок выстраивается стратегия выделения необходимого количества ресурсов под выполнение соответствующих задач, поскольку поддержание работоспособности системы является затратной статьей, что сказывается на общей стоимости предоставляемых провайдером услуг и его конкурентоспособности [19].

Среднее время отклика всей сети определяется суммой средних времен прохождения запроса через каждый отдельный узел

$$T = t_1 + t_2 + \ldots + t_N. \qquad (2)$$

Согласно методу декомпозиции остается определить величину t_i для каждого имеющегося в системе узла, $i = 1, \ldots, N$. Поэтому на первый

план выходят методы, предполагающие одномерную диффузионную аппроксимацию узлов типа G/G/1. При этом стоит отметить, что иногда они допускают довольно серьезные относительные погрешности приближения в зависимости от величины загрузки узлов и выбранных конкретных типов распределений для входящего потока и времен обслуживания.

Кроме того, информация о распределении входящего потока и, соответственно, его первых и вторых моментах доступна только для самого первого узла, который в данном случае представляет собой систему типа fork-join, т. е. параллельный узел.

Для остальных же узлов, учитывая неограниченные емкости накопителей, а также линейную архитектуру сети, можем допустить, что среднее время между соседними поступлениями требований будет таким же, как и на первом узле, а именно $1/\lambda$.

Изучение характеристик выходных потоков узлов рассматриваемой сети и, в частности, самого первого узла, имеющего параллельную структуру, представляет собой отдельную непростую задачу, поэтому допущения, касающиеся оценки моментов выходящих потоков, будут накладывать определенные ограничения, влияющие на точность получаемого решения.

Что касается вторых моментов, а точнее, коэффициентов вариации CV_i для входящих в i-й узел потоков ($i = 2, ..., N$), необходимых для оценки среднего времени пребывания в каждом из узлов, то здесь можно воспользоваться некоторыми известными приближениями. Согласно [20–22]:

$$CV_i = CV_{Pa_{i-1}}; \tag{3}$$

$$CV_i = \rho_{i-1}\left(1-\rho_{i-1}\right) + \rho_{i-1}^2 CV_{Pa_{i-1}}^2 + \left(1-\rho_{i-1}\right)CV_{i-1}^2; \tag{4}$$

$$CV_i = CV_{i-1}^2 + 2\rho_{i-1}CV_{Pa_{i-1}}^2 - \rho_{i-1}\left(CV_{i-1}^2 + CV_{Pa_{i-1}}^2\right)\times \times f\left(\rho_{i-1},\, CV_{i-1},\, CV_{Pa_{i-1}}\right); \tag{5}$$

$$CV_i = \rho_{i-1}^2 CV_{Pa_{i-1}}^2 + \left(1-\rho_{i-1}^2\right)CV_{i-1}^2, \tag{6}$$

где ρ_{i-1} — загрузка $(i-1)$-го узла, которая в рамках рассматриваемой модели идентична для всех узлов: $\rho_i = \rho = \lambda$, $i = 1, ..., N$; $CV_{Pa_{i-1}}$ — коэффициент вариации времени обслуживания:

$$CV_{Pa_i} = CV_{Pa} = \frac{1}{\sqrt{\alpha(\alpha-2)}},\ i=1,\ ...,\ N,$$

причем допустим, что это справедливо и для первого узла; функцию $f(\rho, CV_i, CV_{Pa})$ определим далее по тексту.

Выражения (3)–(6) используются для оценки среднего времени пребывания в i-м узле [23]

$$t_i \approx \frac{\rho}{2(1-\rho)}\left(CV_i^2 + CV_{Pa}^2\right)f\left(\rho,\, CV_i,\, CV_{Pa}\right) + 1,\quad i = 2,\ ...,\ N, \tag{7}$$

где

$$f(\rho,\, CV_i,\, CV_{Pa}) = \begin{cases} \exp\left\{-\dfrac{2(1-\rho)}{3\rho}\dfrac{\left(1-CV_i^2\right)^2}{CV_i^2 + CV_{Pa}^2}\right\}, & \text{если } CV_i \le 1; \\ \exp\left\{-(1-\rho)\dfrac{CV_i^2 - 1}{CV_i^2 + 4CV_{Pa}^2}\right\}, & \text{если } CV_i > 1. \end{cases} \tag{8}$$

Для времени пребывания в самом первом узле, который представляет собой параллельную структуру, состоящую из K подузлов, воспользуемся приближением, представленным в работах [24, 25]:

$$t_1 \approx 1 + \frac{(\alpha-1)^2}{2\alpha(\alpha-2)}\frac{\rho}{1-\rho} + \left(K^{\frac{1}{\alpha}} - 1\right)\times \times \left(1{,}25918 + 0{,}36996\alpha - 1{,}97400\rho - 0{,}28495\alpha\rho + 1{,}40841\rho^2 - 0{,}01122\alpha^2\right)\times \times \sqrt{\frac{1}{\alpha(\alpha-2)} + \frac{(\alpha-1)^3}{3\alpha^2(\alpha-3)}\frac{\rho}{1-\rho} + \frac{(\alpha-1)^4}{4\alpha^2(\alpha-2)^2}\frac{\rho^2}{(1-\rho)^2}}. \tag{9}$$

Данное выражение показывает хорошее качество приближения для значений параметров модели $\alpha \in [4; 10]$, $\rho \in [0{,}1; 0{,}9]$ и числа подузлов параллельной структуры $K = 2, ..., 20$, при этом средняя относительная погрешность приближения составляет около 1,6 %, а максимальная не превышает 4 %.

Стоит отметить, что анализ представленной сети и оценка ее характеристик в случае показательного распределения времени обслуживания со средним значением $b = 1$ и, соответственно, загрузкой $\rho = \lambda < 1$ на каждом из приборов не будет представлять серьезной сложности, если допустить, что поток запросов, поступающий на второй узел, будет также пуассоновским. Тогда времена пребывания в каждом i-м узле, начиная со второго и заканчивая последним, будут иметь показательное распределение с параметром $(1 - \lambda)$, а общее суммарное время прохождения запроса через $(N - 1)$ узел сети составит $\dfrac{N-1}{1-\lambda}$.

Единственная трудность — в определении времени пребывания на первом параллельном узле сети, для которого в случае $K > 2$ нет точных решений [14, 26–31]. Однако в этой ситуации предлагается воспользоваться приближением, полученным в работе [27], которое уточняет наиболее известную аппроксимацию для fork-join-системы с пуассоновским входящим потоком и экспоненциальным распределением времени обслуживания из [26]. Таким образом, выражение для оценки среднего времени пребывания в сети T, учитывая, что $\rho = \lambda$, будет иметь вид

$$T \approx \frac{N-1}{1-\lambda} + \frac{\lambda}{1-\lambda}\left(\frac{H_K}{H_2} - 1\right) \times$$
$$\times \left(0{,}08720 - 0{,}07024\left(\frac{H_K}{H_2} - 1\right) + 0{,}0964\lambda\right) +$$
$$+ \left[\frac{H_K}{H_2} + \frac{4}{11}\left(1 - \frac{H_K}{H_2}\right)\lambda\right]\frac{12-\lambda}{8}\frac{1}{1-\lambda}, \qquad (10)$$

где $H_K = \sum_{i=1}^{K} 1/i$ — частичная сумма гармонического ряда; K — число подузлов в первом параллельном узле.

Численный эксперимент

В данном разделе проверим работоспособность предложенной аппроксимации среднего времени отклика для модели распределенных транзакционных приложений и ее качество, учитывая сделанные предположения о параметрах первого параллельного узла, используемых при аппроксимации коэффициентов вариации входящего потока для последующих узлов сети.

Для этого рассмотрим модель сети транзакционного приложения, число последовательных узлов N в которой меняется от трех до 10, а количество параллельных подузлов K первого узла равно пяти. Уровень загрузки каждого узла $\rho = \lambda \in [0{,}05; 0{,}90]$ с шагом 0,05. Для оценки величины t_i, $i = 1, \ldots, N$, будем использовать выражение (7), где величина коэффициента вариации для входящего в узел потока будет вычисляться по одной из формул (3)–(6), а для оценки среднего времени пребывания в первом узле, соответственно, формулу (9).

Будем сравнивать значения, рассчитанные по аналитическим формулам, и результаты имитационного моделирования математической модели транзакционного приложения, описанной выше. Для имитационного моделирования используется программная среда Python. Для повышения качества симуляции число запросов, пропускаемых через систему, в рамках одного запуска модели для определения одного значения математического ожидания случайной величины времени отклика будет составлять 5 млн для низких значений уровня загруженности сети, т. е. меньших 0,50, и 10 млн для значений загрузки от 0,50 и выше.

Для оценки качества аппроксимации рассмотрим среднюю относительную погрешность приближения, а также ее максимальное значение:

$$\text{MAPE} = \frac{1}{L}\sum_{j=1}^{L}\left|\frac{T_j^* - T_j}{T_j}\right| \cdot 100\%;$$

$$\text{MaxAPE} = \max_{1 \le j \le L}\left|\frac{T_j^* - T_j}{T_j}\right| \cdot 100\%,$$

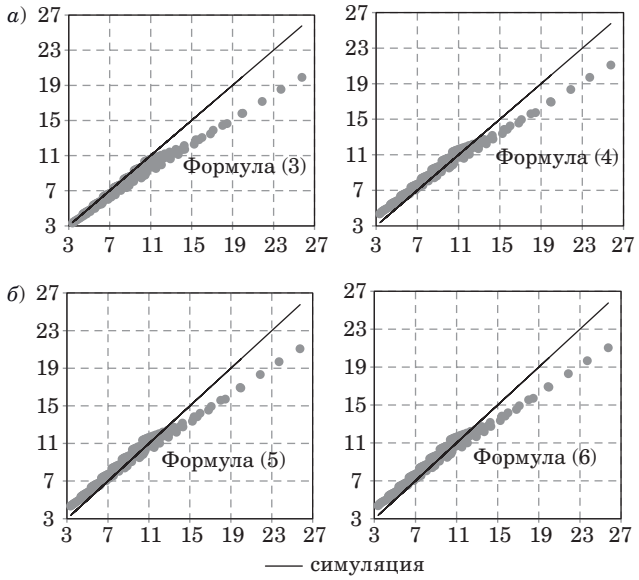
где L — общее количество наборов входных параметров, для которых делаются расчеты, в данном случае $L = 144$; T_j^* — оценка среднего значения времени отклика, рассчитанная по аналитическим формулам (2)–(9); T_j — значение среднего времени отклика модели системы, рассчитанное на статистических данных имитационного моделирования для j-го набора значений входных параметров (значения уровня загрузки системы ρ и общего числа узлов системы).

Значения заявленных типов погрешностей приближения для всех уровней загрузки системы $\rho \in [0{,}05; 0{,}90]$ представлены в табл. 1. Несмотря на то, что средняя погрешность аппроксимации остается в рамках инженерной, т. е. не превышает или совсем незначительно превышает 10 %, ее максимальное значение слишком велико. Поэтому проведем более детальный анализ использования выражений (3)–(6) для расчетов при оценке показателей узлов со второго по 10-й, так как средняя погрешность приближения для расчетов времени пребывания запроса в первом узле не превышает 2 % [23]. На рис. 2, *а* и *б* наглядно сравниваются результаты имитационно-

■ ***Таблица 1.*** Погрешности аппроксимации для среднего времени отклика системы, рассчитанные с помощью формул (3)–(6) для коэффициентов вариации и для значений загрузки системы $\rho \in [0{,}05; 0{,}90]$

■ ***Table 1.*** Approximation errors for the average system response time, calculated using formulas (3)–(6) for the variation coefficients and for the system load values $\rho \in [0.05; 0.90]$

Расчетная формула	MAPE, %	MaxAPE, %
(3)	8,068	22,674
(4)	10,094	30,173
(5)	9,859	29,341
(6)	10,047	30,165



■ ***Рис. 2.*** Сравнение результатов имитационного моделирования величины среднего времени отклика с результатами расчетов по аналитическим формулам (3), (4) (*а*) и (5), (6) (*б*)

■ ***Fig. 2.*** Comparison of the results of simulation modeling of the average response time with the results of calculations using analytical formulas using formulas (3), (4) (*а*) and (5), (6) (*б*)

го моделирования и расчетов по аналитическим формулам.

Формула (3) показывает наилучшее приближение для низких уровней загрузки системы до 0,50 включительно, что отражено в табл. 2 и на соответствующем графике. В отношении формул (4)–(6) ситуация ожидаемо противоположная и свойственная в целом для оценки характеристик подобных узлов (типа G/G/1), т. е. показывает лучшее качество приближения для более высоких уровней нагрузки на узлы, причем на более низком уровне она завышается, а на более высоком уровне занижается. Результаты вычислений схожи, хотя лучший из трех демонстрирует формула (4).

■ ***Таблица 2.*** Погрешности аппроксимации для среднего времени отклика системы, рассчитанные по формулам (3)–(6) для коэффициентов вариации и различных значений загрузки системы

■ ***Table 2.*** Approximation errors for the average system response time calculated using formulas (3)–(6) for the variation coefficients and different system load values

Расчетная формула	MAPE, %	MaxAPE, %
(3), $\rho \in [0{,}05; 0{,}50]$	3,242	7,945
(4), $\rho \in [0{,}55; 0{,}90]$	7,005	18,106
(5), $\rho \in [0{,}55; 0{,}90]$	7,025	18,263
(6), $\rho \in [0{,}55; 0{,}90]$	7,077	18,370

Тем не менее, резюмируя, можно отметить пригодность представленных в работе аналитических формул для первичной оценки среднего времени отклика модели распределенных транзакционных приложений, особенно учитывая, что они не требуют больших вычислительных затрат и дают быстрый результат.

Заключение

Рассмотрена математическая модель распределенных транзакционных приложений с микросервисной архитектурой и параллельными узлами в виде сети массового обслуживания с линейной архитектурой и параллельным узлом. Предполагается, что время обслуживания на узлах имеет распределение Парето, а входящий поток является пуассоновским.

На основе метода декомпозиции проведена оценка среднего времени отклика каждого узла в отдельности, что позволяет получить приближение для среднего времени отклика всей сети. Оценка для непараллельных узлов выполнена с помощью формулы для оценки среднего времени отклика для систем типа G/G/1, в которой фигурируют коэффициенты вариации для входящего потока и времен обслуживания. В отличие от коэффициентов вариации для времени обслуживания в узлах сети, которые известны в силу постановки задачи, сложность представляет оценка коэффициентов вариации для времен между соседними поступлениями запросов между узлами сети, поэтому для их оценки использовано несколько типов приближений. Несмотря на то, что формула для оценки среднего времени отклика внутренних узлов сети может иногда давать не вполне удовлетворительные результаты в условиях слабой загрузки системы и зависит от типа распределения, проведенные вычислительные эксперименты позволяют говорить о приемлемом качестве приближения для случая распределения Парето.

Для оценки среднего времени отклика fork-join-узла применен комплексный подход, включающий имитационное моделирование, визуальный анализ данных и оптимизацию (метод подробно описан в [23]), а средняя погрешность приближения не превышает 2 %. Так, средняя погрешность аппроксимации для среднего времени отклика модели системы транзакционных микросервисных приложений со смешанной последовательной/параллельной структурой в случае более высоких уровней загрузки сети (выше 55 %) не превышает 10 %, а в случае более низкой загрузки (ниже 50 %) — соответственно 5 %.

Таким образом, предложенный подход позволяет оценить среднее время отклика для модели системы транзакционных микросервисных при-

ложений, точность его будет выше за счет качественной оценки времени отклика fork-join-узла.

Кроме того, при таком подходе возможно провести оценку показателей системы и с другими типами распределений, по крайней мере первичную, что позволит поставщикам услуг получить необходимые прогнозы и использовать их при проектировании подобных систем.

Рассмотренная в статье модель транзакционного приложения имеет линейную топологию после узла типа fork-join, при этом реальные взаимодействия микросервисов могут представлять собой сложные графы. Соответственно, изучение сетей более сложной архитектуры, т. е. отличной от линейной, может являться одним из возможных направлений для будущих исследований, так же как и оценка моментов изучаемых случайных величин более высокого порядка, например дисперсии.

Литература

1. **Бондаренко А. С., Зайцев К. С.** Использование систем управления контейнерами для построения распределенных облачных информационных систем с микросервисной архитектурой. *Международный журнал гуманитарных и естественных наук*, 2022, № 64, с. 62–65. doi:10.24412/2500-1000-2022-1-1-62-65
2. **Miao K., Li J., Hong W., Chen M.** A microservice-based big data analysis platform for online educational applications. *Scientific Programming*, 2020, vol. 2020, pp. 1–13. doi:10.1155/2020/6929750
3. **Hao J., Zhao J., Li Y.** Research on decompostion method of relational database oriented to microservice refactoring. *2023 24st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2023, pp. 282–285.
4. **Berardi D., Giallorenzo S., Mauro J., Melis A., Montesi F., Prandini M.** Microservice security: A systematic literature review. *PeerJ Computer Science*, 2022, vol. 8, p. :e779. doi:10.7717/peerj-cs.779
5. **Velepucha V., Flores P.** A survey on microservices architecture: Principles, patterns and migration challenges. *IEEE Access*, 2023, vol. 11, pp. 88339–88358. doi:10.1109/ACCESS.2023.3305687
6. **Harrison G., Marshall A., Custer C.** *Architecting Distributed Transactional Applications*. O'Reilly Media, Incorporated, 2023. 41 p.
7. **Гольчевский Ю. В., Ермоленко А. В.** Актуальность использования микросервисов при разработке информационных систем. *Вестник Сыктывкарского университета. Серия 1. Математика. Механика. Информатика*, 2020, № 2, с. 25–36. EDN: MYITJK
8. **Артамонов Ю. С., Востокин С. В.** Разработка распределенных приложений сбора и анализа данных на базе микросервисной архитектуры. *Известия Самарского научного центра РАН*, 2016, т. 18, № 4-4, с. 688–693. EDN: YGSQTV
9. **Фомин Д. С., Бальзамов А. В.** Проблематика обработки транзакций при использовании микросервисной архитектуры. *Известия высших учебных заведений. Поволжский регион. Технические науки*, 2021, № 2, с. 15–23. doi:10.21685/2072-3059-2021-2-2
10. **Никонов А. А., Стельмашонок Е. В.** Анализ внедрения современных цифровых технологий в финансовой сфере. *Научно-технические ведомости СПбГПУ. Экономические науки*, 2018, № 4, с. 111–119. doi:10.18721/JE.11408, EDN: UYUPJQ
11. **Chatterjee P.** Cloud-native architecture for high-performance payment system. *TIJER-International Research Journals (TIJER)*, 2023, vol. 10, no. 4, pp. 345–358. doi:10.2139/ssrn.5101076
12. **Редругина Н. М.** Метод вычисления временных характеристик обслуживания в сервисных платформах инфокоммуникационных транзакционных услуг с параллельной обработкой запросов. *Труды учебных заведений связи*, 2023, № 3, с. 82–90. doi:10.31854/1813-324X-2023-9-3-82-90
13. **Редругина Н. М., Зарубин А. А.** Модели и методы расчета временных характеристик слабосвязанных транзакционных услуг. *Наукоемкие технологии в космических исследованиях Земли*, 2024, № 2, с. 4–12. doi:10.36724/2409-5419-2024-16-2-4-12
14. **Nguyen M., Alesawi S., Li N., Che H., Jiang H.** A black-box fork-join latency prediction model for data-intensive applications. *IEEE Transactions on Parallel and Distributed Systems*, 2020, vol. 31, no. 9, pp. 1983–2000. doi:10.1109/TPDS.2020.2982137
15. **Gorbunova A. V., Vishnevsky V. M., Larionov A. A.** Evaluation of the end-to-end delay of a multiphase queuing system using artificial neural networks. *Lecture Notes in Computer Science*, 2020, vol. 12563, pp. 631–642. doi:10.1007/978-3-030-66471-8_48
16. **Кутузов О. И., Татарникова Т. М.** К оцениванию и сопоставлению очередей классических и фрактальных систем массового обслуживания. *Информационно-управляющие системы*, 2016, № 2, с. 48–55. doi:10.15217/issn1684-8853.2016.2.48
17. **Задорожный В. Н., Захаренкова Т. Р.** Методы планирования имитационных экспериментов при моделировании фрактальных очередей. *Омский научный вестник*, 2016, № 3, с. 87–92. EDN: VWXULR
18. **Рыжиков Ю. И.** Теория очередей и распределение Парето. *Труды Военно-космической академии им. А. Ф. Можайского*, 2015, № 648, с. 28–43. EDN: UZMKMF
19. **Горбунова А. В., Вишневский В. М.** Оценка времени отклика среды для вычислений с интенсивным использованием данных. *Информационно-*

управляющие системы, 2022, № 4, с. 12–19. doi:10.31799/1684-8853-2022-4-12-19

20. **Reiser M., Kobayashi H.** Accuracy of the diffusion approximation for some queuing systems. *IBM Journal of Research and Development*, 1974, vol. 18, no. 2, pp. 110–124.
21. **Gelenbe E., Pujolle G.** The behaviour of a single queue in a general queueing network. *Acta Informatica*, 1976, vol. 7, no. 2, pp. 123–136.
22. **Kuhn P.** Analysis of complex queuing networks by decomposition. *Proceedings of the 8th International Teletraffic Congress*, 1976, pp. 1–8.
23. **Kraemer W., Langenbach-Belz M.** Approximate formulae for the delay in the queueing system GI|G|1. *Proceedings of the 8th International Teletraffic Congress*, 1976, vol. 235, pp. 1–8.
24. **Gorbunova A. V., Lebedev A. V.** Nonlinear approximation of characteristics of a fork–join queueing system with Pareto service as a model of parallel structure of data processing. *Mathematics and Computers in Simulation*, 2023, vol. 214, pp. 409–428. doi:10.1016/j.matcom.2023.07.029
25. **Gorbunova A. V., Lebedev A. V.** On the features of service rate control in fork-join queueing system. *Automation and Remote Control*, 2024, vol. 85, no. 12, pp. 1184–1198. doi:10.31857/S0005117924120043
26. **Nelson R., Tantawi A. N.** Approximate analysis of fork/join synchronization in parallel queues. *IEEE Transactions on Computers*, 1988, vol. 37, pp. 739–743. doi:10.1109/12.2213
27. **Gorbunova A. V., Lebedev A. V.** On estimating the characteristics of a fork-join queueing system with Poisson input and exponential service times. *Advances in Systems Science and Applications*, 2023, vol. 23, no. 2, pp. 99–114. doi:10.25728/assa.2023.23.2.1351
28. **Thomasian A.** Analysis of fork/join and related queueing systems. *ACM Computing Surveys (CSUR)*, 2014, vol. 47, pp. 17:1–17:71. doi:10.1145/2628913
29. **Varki E., Merchant A., Chen H.** *The M/M/1 fork-join queue with variable subtasks*. https://www.cs.unh.edu/~varki/publication/2002-nov-open.pdf (дата обращения: 05.05.2024).
30. **Qiu Z., Perez J. F., Harrison P. G.** Beyond the mean in fork-join queues: Efficient approximation for response-time tails. *Performance Evaluation*, 2015, vol. 91, pp. 99–116. doi:10.1016/j.peva.2015.06.007
31. **Wang W., Harchol-Balter M., Jiang H., Scheller-Wolf A., Srikant R.** Delay asymptotics and bounds for multitask parallel jobs. *Queueing Systems*, 2019, vol. 91, pp. 207–239. doi:10.1007/s11134-018-09597-5

UDC 004.032
doi:10.31799/1684-8853-2025-6-42-50
EDN: EGLAUQ

Evaluation of the characteristics of a distributed transactional application model with microservice architecture and fork-join structures

A. V. Gorbunova[a], PhD, Phys.-Math., Senior Researcher, orcid.org/0000-0002-9183-0426, avgorbunova@list.ru
[a]V. A. Trapeznikov Institute of Control Sciences of RAS, 65, Profsoyuznaya St., 117997, Moscow, Russian Federation

Introduction: A microservice architecture, which allows applications to be built as a set of independent microservices that work together to fulfill a common client request, has recently become the basis or even the standard for deploying complex systems that affect multiple physical structures and devices. In addition, the introduction of parallel service scenarios into such systems, especially in the case of high-load systems, makes it possible to increase their efficiency and performance. **Purpose:** To develop a mathematical model of distributed transactional applications with a microservice architecture and parallel nodes and to evaluate such a performance indicator as the average response time. **Results:** We present a mathematical model of distributed transactional applications with a microservice architecture in the form of a queueing network with sequential nodes, one of which has a parallel structure with several subnodes, the number of those is more than two. Based on the decomposition method for analyzing queueing networks, an approach to estimating the average response time of the system under consideration is proposed. We use known results for assessing individual nodes of a G/G/1 type network, as well as nodes with division and with parallel service. The results of the computational experiments allow drawing conclusions about the admissibility of the proposed approach, and obtaining recommendations regarding the applicability of various formulas for different load levels, in particular, for those whose average approximation error does not exceed 10%. **Practical relevance:** The proposed model and the method of its research can be used for the initial assessment and prediction of the average response time of transactional applications with parallel nodes at different load levels and, as a result, can contribute to maintaining the required quality of service for users of transactional applications.

Keywords — transactional applications, microservice architecture, distributed systems, parallel operations, queueing network, average response time, fork-join, G/G/1.

For citation: Gorbunova A. V. Evaluation of the characteristics of a distributed transactional application model with microservice architecture and fork-join structures. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 42–50 (In Russian). doi:10.31799/1684-8853-2025-6-42-50, EDN: EGLAUQ

References

1. Bondarenko A. S., Zaytsev K. S. Using container management systems to build distributed cloud information systems with microservice architecture. *International Journal of Humanities and Natural Sciences,* 2022, vol. 1-1(64). pp. 62–65. (In Russian). doi:10.24412/2500-1000-2022-1-1-62-65
2. Miao K., Li J., Hong W., Chen M. A microservice-based big data analysis platform for online educational applications. *Scientific Programming*, 2020, vol. 2020, pp. 1–13. doi:10.1155/2020/6929750
3. Hao J., Zhao J., Li Y. Research on decompostion method of relational database oriented to microservice refactoring. *2023 24st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, 2023, pp. 282–285.
4. Berardi D., Giallorenzo S., Mauro J., Melis A., Montesi F., Prandini M. Microservice security: A systematic literature review. *PeerJ Computer Science*, 2022, vol. 8, p. :e779. doi:10.7717/peerj-cs.779
5. Velepucha V., Flores P. A survey on microservices architecture: Principles, patterns and migration challenges. *IEEE Access*, 2023, vol. 11, pp. 88339–88358. doi:10.1109/ACCESS.2023.3305687
6. Harrison G., Marshall A., Custer C. *Architecting Distributed Transactional Applications*. O'Reilly Media, Incorporated, 2023. 41 p.
7. Golchevskiy Yu. V., Yermolenko A. V. The relevance of using microservices in the development of information systems. *Vestnik Syktyvkarskogo Universiteta. Seriya 1: Matematika. Mekhanika. Informatika*, 2020, vol. 2(35), pp. 25–36 (In Russian). EDN: MYITJK
8. Artamonov Yu. S., Vostokin S. V. Development of distributed applications for data collection and analysis on the basis of a microservice architecture. *Izvestiya Samarskogo Nauchnogo Centra RAN*, 2016, vol. 18, no. 4-4, pp. 688–693 (In Russian). EDN: YGSQTV
9. Fomin D. S., Bal'zamov A. V. The problem of transaction processing using microservice architecture. *University Proceedings. Volga Region. Engineering Sciences*, 2021, no. 2, pp. 15–23 (In Russian). doi:10.21685/2072-3059-2021-2-2
10. Nikonov A. A., Stelmashonok E. V. Analysis of modern digital technologies' implementation in the financial sphere. *St. Petersburg State Polytechnical University Journal. Economics*, 2018, vol. 11, no. 4, pp. 111–119 (In Russian). doi:10.18721/JE.11408, EDN: UYUPJQ
11. Chatterjee P. Cloud-native architecture for high-performance payment system. *TIJER-International Research Journals (TIJER)*, 2023, vol. 10, no. 4, pp. 345–358. doi:10.2139/ssrn.5101076
12. Redrugina N. M. Method for time characteristics calculating in the service platforms of info-communication transactional services with parallel requests processing. *Proceedings of Telecommunication Universities*, 2023, vol. 9, no. 3, pp. 82–90 (In Russian). doi:10.31854/1813-324X-2023-9-3-82-90
13. Redrugina N. M., Zarubin A. A. Models and methods for calculating the temporal characteristics of loosely coupled transactional services. *High Technologies in Earth Space Research*, 2024, vol. 16, no. 2, pp. 4–12 (In Russian). doi:10.36724/2409-5419-2024-16-2-4-12
14. Nguyen M., Alesawi S., Li N., Che H., Jiang H. A black-box fork-join latency prediction model for data-intensive applications. *IEEE Transactions on Parallel and Distributed Systems*, 2020, vol. 31, no. 9, pp. 1983–2000. doi:10.1109/TPDS.2020.2982137
15. Gorbunova A. V., Vishnevsky V. M., Larionov A. A. Evaluation of the end-to-end delay of a multiphase queuing system using artificial neural networks. *Lecture Notes in Computer Science*, 2020, vol. 12563, pp. 631–642. doi:10.1007/978-3-030-66471-8_48
16. Kutuzov O. I., Tatarnikova T. M. Evaluation and comparison of queues in classical and fractal queuing systems. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2016, no. 2, pp. 48–55 (In Russian). doi:10.15217/issn1684-8853.2016.2.48
17. Zadorozhnyi V. N., Zakharenkova T. R. Methods for planning simulation experiments in modeling fractal queues. *Omsk Scientific Bulletin*, 2016, no. 3(147), pp. 87–92 (In Russian). EDN: VWXULR
18. Ryzhikov Yu. I. Queuing theory and Pareto distribution. *Trudy Voenno-kosmicheskoj akademii im. A. F. Mozhajskogo*, 2015, no. 648, pp. 28–43 (In Russian). EDN: UZMKMF
19. Gorbunova A. V., Vishnevsky V. M. Estimating the response time of a data-intensive computing environment. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2022, no. 4, pp. 12–19 (In Russian). doi:10.31799/1684-8853-2022-4-12-19
20. Reiser M., Kobayashi H. Accuracy of the diffusion approximation for some queuing systems. *IBM Journal of Research and Development*, 1974, vol. 18, no. 2, pp. 110–124.
21. Gelenbe E., Pujolle G. The behaviour of a single queue in a general queueing network. *Acta Informatica*, 1976, vol. 7, no. 2, pp. 123–136.
22. Kuhn P. Analysis of complex queuing networks by decomposition. *Proceedings of the 8th International Teletraffic Congress*, 1976, pp. 1–8.
23. Kraemer W., Langenbach-Belz M. Approximate formulae for the delay in the queueing system GI|G|1. *Proceedings of the 8th International Teletraffic Congress*, 1976, vol. 235, pp. 1–8.
24. Gorbunova A. V., Lebedev A. V. Nonlinear approximation of characteristics of a fork-join queueing system with Pareto service as a model of parallel structure of data processing. *Mathematics and Computers in Simulation*, 2023, vol. 214, pp. 409–428. doi:10.1016/j.matcom.2023.07.029
25. Gorbunova A. V., Lebedev A. V. On the features of service rate control in fork-join queueing system. *Automation and Remote Control*, 2024, vol. 85, no. 12, pp. 1184–1198. doi:10.31857/S0005117924120043
26. Nelson R., Tantawi A. N. Approximate analysis of fork/join synchronization in parallel queues. *IEEE Transactions on Computers*, 1988, vol. 37, pp. 739–743. doi:10.1109/12.2213
27. Gorbunova A. V., Lebedev A. V. On estimating the characteristics of a fork-join queueing system with Poisson input and exponential service times. *Advances in Systems Science and Applications*, 2023, vol. 23, no. 2, pp. 99–114. doi:10.25728/assa.2023.23.2.1351
28. Thomasian A. Analysis of fork/join and related queueing systems. *ACM Computing Surveys (CSUR)*, 2014, vol. 47, pp. 17:1–17:71. doi:10.1145/2628913
29. Varki E., Merchant A., Chen H. *The M/M/1 fork-join queue with variable subtasks*. Available at: https://www.cs.unh.edu/~varki/publication/2002-nov-open.pdf (accessed 5 May 2024).
30. Qiu Z., Perez J. F., Harrison P. G. Beyond the mean in fork-join queues: Efficient approximation for response-time tails. *Performance Evaluation*, 2015, vol. 91, pp. 99–116. doi:10.1016/j.peva.2015.06.007
31. Wang W., Harchol-Balter M., Jiang H., Scheller-Wolf A., Srikant R. Delay asymptotics and bounds for multitask parallel jobs. *Queueing Systems*, 2019, vol. 91, pp. 207–239. doi:10.1007/s11134-018-09597-5

УДК 003.26
doi:10.31799/1684-8853-2025-6-51-63
EDN: AEXSDC

Научные статьи
Articles

Применение метода компактного описания подстановки для модификации схемы цифровой подписи на основе протокола аутентификации Штерна

И. С. Ниткин[а,б], *аспирант, orcid.org/0000-0001-5240-1744, exebopen@gmail.com*
[а]*Университет ИТМО, Кронверкский пр., 49, Санкт-Петербург, 197101, РФ*
[б]*Санкт-Петербургский государственный университет аэрокосмического приборостроения, Б. Морская ул., 67, Санкт-Петербург, 190000, РФ*

Введение: *схема аутентификации без разглашения Штерна — один из востребованных протоколов для построения схемы цифровой подписи на основе корректирующих кодов. Важным недостатком формирования схемы подписи на основе таких протоколов является значительный размер подписи.* ***Цель:*** *разработать оптимизированную по памяти версию квантово-устойчивой схемы электронную подпись, основанную на схеме Штерна.* ***Результаты:*** *исследована схема подписи на основе протокола аутентификации Штерна, размеры блоков элементов структуры подписи, взаимосвязь размеров блоков со значениями открытых параметров схемы подписи. Сделан вывод, что наибольший объем памяти необходим для хранения значений случайных подстановок. Выполнена оценка криптостойкости схемы подписи на основе протокола аутентификации Штерна с использованием модели наилучшей известной атаки. Разработан метод компактного описания подстановки с использованием информационного вектора, который применен для модификации схемы подписи на основе протокола аутентификации Штерна. Выполнена оценка уровня криптографической стойкости в модели наилучшей известной атаки модифицированной схемы подписи на основе протокола аутентификации Штерна. Проведено теоретическое и экспериментальное сравнение функциональных характеристик схемы цифровой подписи на основе протокола аутентификации Штерна и ее модифицированной версии. Предложенная модификация позволяет значительно снизить размеры подписи, при этом, с учетом выбранной модели оценки, общий уровень криптографической стойкости схемы подписи не снижается.* ***Практическая значимость:*** *полученные в рамках исследования результаты могут быть использованы для оптимизации по памяти схем подписи на основе других протоколов аутентификации без разглашения на основе корректирующих кодов, а также для разработки других методов оптимизации по памяти схемы цифровой подписи на основе протокола аутентификации Штерна.*

Ключевые слова *— постквантовая криптография, криптография на основе корректирующих кодов, электронная подпись, схема цифровой подписи, схема Штерна.*

Для цитирования: Ниткин И. С. Применение метода компактного описания подстановки для модификации схемы цифровой подписи на основе протокола аутентификации Штерна. *Информационно-управляющие системы*, 2025, № 6, с. 51–63. doi:10.31799/1684-8853-2025-6-51-63, EDN: AEXSDC

For citation: Nitkin I. S. Permutation compact description method and its application for Stern-based digital signature modification. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 51–63 (In Russian). doi:10.31799/1684-8853-2025-6-51-63, EDN: AEXSDC

Введение

Постквантовая криптография представляет собой раздел криптографии, который изучает алгоритмы и протоколы, устойчивые к атакам при помощи квантового компьютера (квантово-устойчивые криптографические схемы) [1].

Квантовый алгоритм Шора позволяет за полиномиальное время решать задачи факторизации целого числа и дискретного логарифмирования [2]. Вычислительная сложность описанных задач лежит в основе стойкости криптографических систем, как соответствующих общемировым стандартам (RSA, DSA, ECDSA и т.п.), так и описанных в стандартах Российской Федерации [3]. Таким образом, при использовании алгоритма Шора теоретически могут быть осуществлены успешные атаки на большинство существующих на данный момент систем защиты информации.

Для практической реализации таких атак необходим криптографически-релевантный квантовый компьютер (CRQC), который в настоящее время не может быть реализован технологически. Несмотря на то, что технологии квантовых вычислений развиваются стремительно, исследователи утверждают, что появления CRQC по самым оптимистичным оценкам можно ожидать не ранее 2030 г.

При этом специфика применения криптографических средств в системах защиты информации требует разработки и внедрения квантово-устойчивых криптографических схем, не дожидаясь момента появления CRQC. Применение асимметричных алгоритмов шифрования дает возможность злоумышленнику осуществить сбор зашифрованной информации в ожидании появления технологий, позволяющих выполнить быстрое дешифрование без знания секрет-

ных ключей. Поэтому некоторые категории информации, для которых необходимо обеспечение конфиденциальности на длительные сроки, являются потенциально уязвимыми [4].

Аналогично, квантово-устойчивые схемы подписи должны внедряться заблаговременно, потому что появление CRQC потребует не только единовременного внедрения устойчивых алгоритмов в информационные системы, но и обновления долговременных подписей доверенных удостоверяющих центров, дистрибутивов и т.п., что не представляется возможным на практике.

Национальный институт стандартов и технологий США проводит конкурсные исследования на выработку стандартов квантово-устойчивых криптографических схем [4, 5]. По итогам конкурса [4] среди прочих для стандартизации была выбрана схема выработки общего секретного ключа на основе квазициклических кодов [6].

В рамках рабочей группы Технического комитета по стандартизации «Криптографическая защита информации» Росстандарта (ТК 26) также проводятся исследования в этом направлении. Одним из объектов исследования является схема цифровой подписи на основе схемы Штерна, получившая название «Шиповник» (https://kryptonite.ru/articles/how-eds-will-change-in-the-post-quantum-era/). Стойкость данной схемы основана на NP-полной задаче синдромного декодирования произвольного линейного кода [7].

Использование в качестве криптографического примитива корректирующих кодов является одним из основных подходов при построении квантово-устойчивых криптографических схем [8]. Значимым преимуществом такого подхода является возможность обеспечения стойкости криптографической схемы на уровне сложности решения NP-полной задачи синдромного декодирования. Кроме того, корректирующие коды являются одним из наиболее исследованных постквантовых криптографических примитивов.

Первая криптосистема на основе корректирующих кодов была предложена Робертом МакЭлисом в 1978 г. [9]. В 2001 г. Николас Куртуа, Матье Финиаз и Николас Сандрие предложили схему подписи на основе криптосистемы МакЭлиса [10]. Главными недостатками данной схемы являются необходимость многократно повторять алгоритм выработки подписи, а также значительные размеры ключевой пары. Кроме этого, стойкость данной схемы подписи строится на предположении, что поиск порождающего многочлена и вектора локаторов для кода Гоппы по проверочной матрице является вычислительно сложной задачей. Несмотря на многолетние безуспешные попытки опровергнуть это утверждение, на сегодня вычислительная сложность данной задачи не формализована в контексте классов сложности вычислительных задач.

Другим вариантом построения схемы подписи на основе корректирующих кодов является использование протоколов аутентификации без разглашения. Для NP-полной задачи синдромного декодирования произвольного линейного двоичного кода такая схема аутентификации была предложена Жаком Штерном [11] в 1993 г. В качестве развития идей Штерна были предложены другие схемы аутентификации без разглашения на основе произвольных линейных кодов [12–19].

На основе протокола аутентификации без разглашения может быть разработана схема подписи при помощи преобразования Фиата — Шамира [20]. В частности, на основе протокола CROSS ID [15, 16] разработана схема подписи [21], которая прошла отбор первого раунда [22] конкурса [5].

На основе схемы Штерна также разработана квантово-устойчивая схема электронной подписи [23]. Одним из важных недостатков данной схемы является значительный размер формируемой подписи, что определяется необходимостью многократно повторять процедуру аутентификации без разглашения. Кроме того, необходимость проверки веса Хэмминга секретного ключа без разглашения его значения требует хранения в составе подписи произвольных подстановок. Для одной такой подстановки необходимо хранить в памяти вектор значений от 1 до n, где n — это длина подстановки.

В рамках настоящего исследования предложен метод компактного описания подстановки с помощью информационного вектора параметризуемой длины. С использованием данного метода предложена модифицированная схема подписи на основе протокола аутентификации Штерна, позволяющая оптимизировать объем памяти, необходимый для хранения подписи.

Схема подписи на основе протокола аутентификации Штерна

Схема аутентификации Штерна

Протокол аутентификации без разглашения позволяет доказать знание некоторого «секрета» (например, секретного ключа), не раскрывая при этом его значение. Схема Штерна [11] — протокол аутентификации без разглашения, стойкость которого основана на NP-полной задаче синдромного декодирования.

Принцип работы схемы заключается в обмене сообщениями между доказывающим и проверяющим по открытому каналу.

Открытые параметры схемы Штерна:

— значения n, k;

— функция хеширования $h(x)$: $x \to \{0, 1\}^l$;

— проверочная матрица $\mathbf{H}^{n\times k}$ произвольного двоичного линейного кода;

— значение $\mathbf{y} = \mathbf{H}\mathbf{s}^{\mathrm{T}}$, где $\mathbf{s}$ — двоичный вектор длины n заданного веса Хэмминга ω, $\mathrm{wt}(\mathbf{s}) = \omega$.

В качестве секретного ключа в схеме Штерна рассматривается двоичный вектор $\mathbf{s}$ заданного веса ω.

Процесс аутентификации происходит следующим образом.

Шаг 1. Доказывающий выбирает произвольный вектор $\mathbf{u} \in \{0, 1\}^n$ и произвольную подстановку $\sigma \in S_n$, где S_n — симметрическая группа степени n. На основе выбранных значений доказывающий вычисляет значения обязательств $c_0 = h(\sigma \mid \mathbf{H}\mathbf{u}^{\mathrm{T}})$, $c_1 = h(\sigma(\mathbf{u}))$, $c_2 = h(\sigma(\mathbf{u} \oplus \mathbf{s}))$ и передает их проверяющему.

Шаг 2. Проверяющий отправляет доказывающему случайное значение запроса $b \in \{0, 1, 2\}$.

Шаг 3. В зависимости от полученного значения b доказывающий отправляет проверяющему значения ответов r:

если $b = 0$, $r = \sigma \mid \mathbf{u}$;

если $b = 1$, $r = \sigma \mid \mathbf{u} \oplus \mathbf{s}$;

если $b = 2$, $r = \sigma(\mathbf{u}) \mid \sigma(\mathbf{s})$.

Шаг 4. Проверяющий выполняет проверки:

если $b = 0$:

$$c_0 \overset{?}{=} h\left(\sigma \mid \mathbf{H}\mathbf{u}^{\mathrm{T}}\right),\ c_1 \overset{?}{=} h\left(\sigma(\mathbf{u})\right);$$

если $b = 1$:

$$c_0 \overset{?}{=} h\left(\sigma \mid \mathbf{H}(\mathbf{u} \oplus \mathbf{s})^{\mathrm{T}} \oplus \mathbf{y}\right),\ c_2 \overset{?}{=} h\left(\sigma(\mathbf{u} \oplus \mathbf{s})\right);$$

если $b = 2$:

$$c_1 \overset{?}{=} h\left(\sigma(\mathbf{u})\right),\ c_2 \overset{?}{=} h\left(\sigma(\mathbf{u} \oplus \mathbf{s})\right),\ \mathrm{wt}\left(\sigma(\mathbf{s})\right) \overset{?}{=} \omega.$$

Шаги 1–4 повторяются δ раз, где δ — заданный параметр стойкости.

Если все проверки пройдены успешно, знание значения секретного ключа считается подтвержденным.

Вероятность принятия доказательства проверяющей стороной в одном раунде при условии, что доказывающий не обладает знанием значения секрета, составляет 2/3. Если отбросить одно из возможных значений запроса b, знание секрета не является необходимым для выработки обязательства, которое будет принято проверяющей стороной.

Если $b \neq 1$, то злонамеренный доказывающий вместо вектора $\mathbf{s}$ при формировании обязательств c_0, c_1, c_2 использует произвольный вектор $\mathbf{t}_1$: $\mathrm{wt}(\mathbf{t}_1) = \omega$.

Если $b \neq 2$, при формировании обязательств вместо $\mathbf{s}$ используется вектор $\mathbf{t}_2$: $\mathbf{y} = \mathbf{H}\mathbf{t}_2^{\mathrm{T}}$.

Если $b \neq 0$, вместо $\mathbf{s}$ используется вектор $\mathbf{t}_1$, а $c_0 = h(\sigma \mid \mathbf{H}(\mathbf{u} \oplus \mathbf{t}_1)^{\mathrm{T}} \oplus \mathbf{y})$.

Преобразование Фиата — Шамира [20] позволяет на основе протоколов аутентификации без разглашения формировать схемы подписи. В том числе на основе схемы Штерна предложена схема подписи [23].

Схема цифровой подписи не может быть реализована в интерактивном виде, поэтому вместо генерации случайного значения b оно вычисляется при помощи троичной хеш-функции $f(x)$: $x \to \{0, 1, 2\}^{\delta}$.

При формировании подписи в первую очередь вычисляется δ наборов обязательств по схеме Штерна ($c = c_0 \mid \ldots \mid c_{\delta-1}$). После этого сформированные обязательства вместе с подписываемым сообщением подаются на вход троичной хеш-функции $f(x)$, и на основе полученного значения вектора вызовов ($b = f(c \mid m) = b_0 \mid \ldots \mid b_{\delta-1}$) формируется блок ответов ($r = r_0 \mid \ldots \mid r_{\delta-1}$) в соответствии со схемой Штерна. Значение подписи представляет собой конкатенацию блоков c и r ($Sig = c \mid r$).

При проверке подписи в первую очередь вычисляется значение вектора вызовов, после чего осуществляется δ проверок обязательств по схеме Штерна. Подпись принимается, если все проверки пройдены успешно.

Подробное описание схемы цифровой подписи на основе протокола аутентификации Штерна (СП СШ) представлено в [23].

Исследование размеров блоков подписи на основе схемы Штерна

Для достижения цели оптимизации по памяти подписи, основанной на схеме Штерна, проведена оценка размеров блоков структуры подписи исходя из значений открытых параметров схемы подписи:

— размер блока c_i равен $3l$ бит, где l — битовая длина значения хеш-функции $h(x)$;

— размер блока r_i зависит от значения b_i. Если $b_i \in \{0, 1\}$, размер блока r_i составляет $n + n \cdot \log_2 n$ бит. При $b_i = 2$ размер блока r_i составляет $2n$ бит.

Конкретные значения системных параметров для СП СШ подбираются из соображений необходимости обеспечения криптографической стойкости [24]. В рамках исследования рассматриваются два набора значений системных параметров схемы подписи на основе схемы Штерна, которые приводятся в научных источниках. В ч. 1 табл. 1 представлены значения системных параметров из указанных наборов.

Для набора № 1 [25] значение l не приведено в источнике, поэтому $l = 112$ (с точностью до одного байта) вычислено пропорционально значению n на основе значений набора № 2 [23].

Размеры элементов структуры СП СШ для исследуемых наборов конкретных значений си-

■ ***Таблица 1.*** Системные параметры и характеристики схемы цифровой подписи на основе протокола аутентификации Штерна

■ ***Table 1.*** Stern-based digital signature open parameters and characteristics

Значение	Формула	Набор № 1	Набор № 2
Часть 1. Значения системных параметров			
n	–	620	2896
k	–	310	1448
ω	–	68	318
δ	–	137	137
l	–	(112)	512
Часть 2. Размеры блоков формируемой подписи			
$\lvert c_i \rvert$	$3l$	336 бит	1536 бит
$\lvert r_{i,\mathbf{u}} \rvert$	$2n$	1240 бит	5792 бит
$\lvert r_{i,\sigma} \rvert$	$n + n \cdot \log_2 n$	6820 бит	37648 бит
$\lvert c \rvert$	$\delta \cdot \lvert c_i \rvert$	5,62 Кбайт	25,69 Кбайт
$\lvert r \rvert$	$\delta \cdot \lvert r_{i,\sigma} \rvert$	114,06 Кбайт	629,61 Кбайт
$\lvert Sig \rvert$	$\lvert c \rvert + \lvert r \rvert$	119,67 Кбайт	655,30 Кбайт
Часть 3. Оценка уровня криптографической стойкости			
λ_δ	$-\log_2\left(\frac{2}{3}\right)^\delta$	80 бит	80 бит
λ_{HC}	$\frac{l}{2}$	56 бит	256 бит
$\lambda_{\mathbf{s}}$	$\log_2\binom{n}{\omega}$	305 бит	1440 бит
λ_{SD}	$0{,}0885 \cdot n$	54 бита	256 бит
$\lambda_{\mathbf{u}}$	n	620 бит	2896 бит
λ_σ	$\log_2\binom{n}{\omega}$	305 бит	1440 бит
λ	$\min(\lambda_\delta, \lambda_{HC}, \lambda_{\mathbf{s}}, \lambda_{SD}, \lambda_{\mathbf{u}}, \lambda_\sigma)$	54 бита	80 бит

стемных параметров представлены в ч. 2 табл. 1. Наибольший объем памяти в структуре СП СШ, значительно превосходящий объем памяти, требуемый для других значений в составе подписи, необходим для хранения случайных подстановок.

В табл. 1, 2 использованы следующие обозначения:

– $\lvert c_i \rvert$ – размер блока c_i;
– $\lvert r_{i,\mathbf{u}} \rvert$ – размер блока r_i при условии $b_i = 2$;
– $\lvert r_{i,\sigma} \rvert$ – размер блока r_i при условии $b_i \in \{0, 1\}$;
– $\lvert c \rvert$ – размер блока c;
– $\lvert r \rvert$ – размер блока r;
– $\lvert Sig \rvert$ – размер формируемой подписи;
– λ_δ – уровень криптостойкости при атаке на подделку подписи;
– λ_{HC} – уровень криптостойкости при атаках, основанных на поиске коллизии хеш-функции;
– $\lambda_{\mathbf{s}}$ – уровень криптостойкости при атаке перебора возможных значений секретного ключа;
– λ_{SD} – уровень криптостойкости при атаке с использованием декодирования по информационным совокупностям;
– $\lambda_{\mathbf{u}}$ – уровень криптостойкости при атаке перебора возможных значений $\mathbf{u}_i$;
– λ_σ – уровень криптостойкости при подборе значения $\mathbf{s}$ по известному значению $\sigma_i(\mathbf{s})$;
– λ – общий уровень криптостойкости схемы подписи.

Оценка уровня криптографической стойкости схемы цифровой подписи на основе протокола аутентификации Штерна

В рамках настоящего исследования для оценки криптографической стойкости применяется *модель наилучшей известной атаки*, в которой в качестве показателя избран уровень криптографической стойкости, измеренный в битах. В качестве общего уровня стойкости криптографической схемы принимается наименьшее значение среди оценок уровня криптографической стойкости для атак на исследуемую схему.

При проведении оценки криптографической стойкости СП СШ рассматриваются атаки с использованием детерминированной машины Тьюринга. В рамках настоящего исследования оценка уровня криптографической стойкости СП СШ при атаках с использованием квантового компьютера не проводилась. Применение квантового компьютера позволяет получить значительное ускорение при решении проблемы синдромного декодирования [26]. Последние публикации предлагают алгоритмы, которые позволяют декодировать произвольный линейный код за $2^{0{,}0508n}$ вычислительных операций [27]. Кроме того, алгоритм Гровера [28] может обеспечивать квадратичное ускорение перебора.

При проведении оценки криптографической стойкости схемы цифровой подписи на основе протокола аутентификации Штерна рассмотрены атаки на подделку подписи и атаки на раскрытие значения секретного ключа.

При реализации *атаки на подделку подписи* злоумышленник должен предъявить пару (m, Sig) – произвольное сообщение и подпись данного сообщения, выработанную без знания секретного ключа, которая успешно пройдет проверку.

Алгоритм проведения атаки на подделку подписи.

Шаг 1. Сформировать δ наборов обязательств $c = c_0 \mid \ldots \mid c_{\delta-1}$, каждый из которых пройдет проверку по схеме Штерна с вероятностью $\frac{2}{3}$.

Шаг 2. Подобрать такое значение m, при котором набор значений запросов $b_0 \mid \ldots \mid b_{\delta-1} = f(c \mid m)$ совпадет с номерами проверок, прохождение которых заложено для каждого набора обязательств.

Уровень криптостойкости СП СШ при атаке на подделку подписи в данном случае определяется по формуле $\lambda_\delta = -\log_2\left(\frac{2}{3}\right)^\delta$ [бит].

Другие атаки на подделку подписи связаны с поиском коллизии хеш-функции. Лучший алгоритм поиска коллизии хеш-функции основан на парадоксе дней рождения и требует $2^{\frac{l}{2}}$ операций хеширования, следовательно, уровень криптостойкости СП СШ при *атаках, основанных на поиске коллизии хеш-функции*, определяется по формуле $\lambda_{HC} = \frac{l}{2}$ [бит].

Подбор секретного ключа *по публичному ключу и известным параметрам системы* может быть осуществлен либо посредством полного перебора возможных значений секретного ключа, либо посредством решения NP-полной задачи синдромного декодирования произвольного линейного кода.

Уровень криптостойкости при *атаке с использованием перебора возможных значений секретного ключа* определяется исходя из мощности множества допустимых значений секретного ключа, которая вычисляется как число сочетаний из n по ω. Следовательно, уровень криптостойкости при данной атаке определяется по формуле $\lambda_{\mathbf{s}} = \log_2\binom{n}{\omega}$ [бит].

Лучший из известных алгоритмов решения задачи синдромного декодирования основан на декодировании по информационным совокупностям и требует порядка $2^{0{,}0885n}$ битовых операций [29]. Таким образом, уровень криптостойкости при *атаке с использованием декодирования по информационным совокупностям* определяется по формуле $\lambda_{SD} = 0{,}0885n$ [бит].

Важно отметить, что эффективность алгоритмов декодирования по информационным совокупностям постоянно улучшается. Одним из актуальных подходов является сведение данной задачи к вычислительно сложным задачам на алгебраических решетках [30].

Для атак по *подбору значения секретного ключа на основе значений блока r* рассмотрены нижеописанные варианты.

При $b_i = 1$ может быть осуществлен *подбор значения* $\mathbf{s}$ *по известному значению* $\mathbf{u}_i \oplus \mathbf{s}$. Вычислительная сложность такой атаки эквивалентна перебору возможных значений $\mathbf{u}_i$. Уровень криптографической стойкости схемы подписи на основе схемы Штерна при реализации данной атаки определяется по формуле $\lambda_{\mathbf{u}} = n$ [бит].

При $b_i = 2$ может быть осуществлен подбор значения $\mathbf{s}$ по известному значению $\sigma_i(\mathbf{s})$. Для реализации атаки необходимо выполнить проверку:

$$\mathbf{y} \stackrel{?}{=} \mathbf{H}\left(\sigma_s\left(\sigma_i\left(\mathbf{s}\right)\right)\right)^{\mathrm{T}}, \qquad (1)$$

где $\sigma_{\mathbf{s}}$ — случайно выбранная подстановка длины n.

Если проверка пройдена успешно, значение секретного ключа восстановлено. Количество подстановок $\sigma_{\mathbf{s}}$ таких, что $\sigma_{\mathbf{s}}^{-1}\left(\mathbf{s}\right) = \sigma_i\left(\mathbf{s}\right)$, составляет $n!(n-\omega)!$.

Вероятность успешной атаки вычисляется по формуле

$$P_S = \frac{Q_1 Q_0}{Q} = \frac{\omega!(n-\omega)!}{n!} = \binom{n}{\omega}^{-1}, \qquad (2)$$

где Q_1 — количество способов поместить ω единиц двоичного вектора $\mathbf{s}$ на ω позиций; Q_0 — количество способов поместить $n - \omega$ нулей двоичного вектора $\mathbf{s}$ на $n - \omega$ позиций; Q — общее количество возможных подстановок длины n.

Тогда уровень криптографической стойкости при *атаке на подбор значения секретного ключа по известному значению* $\sigma_i(\mathbf{s})$ определяется по формуле

$$\lambda_\sigma = -\log_2 P_S = \log_2\binom{n}{\omega}. \qquad (3)$$

Общий уровень криптографической стойкости СП СШ в рамках модели наилучшей известной атаки вычисляется по формуле

$$\lambda = \min(\lambda_\delta, \lambda_{HC}, \lambda_{\mathbf{s}}, \lambda_{SD}, \lambda_{\mathbf{u}}, \lambda_\sigma). \qquad (4)$$

В части 3 табл. 1 приведены расчеты уровня криптографической стойкости СП СШ при значениях системных параметров из наборов № 1 и 2.

В работе [23] приводится оценка уровня доказуемой криптографической стойкости для набора № 2 значений системных параметров на уровне 70 бит. Данный результат не противоречит значению, полученному в модели наилучшей известной атаки.

Модифицированная схема цифровой подписи на основе протокола аутентификации Штерна

Метод компактного описания подстановки при помощи информационного вектора

Для достижения цели оптимизации по памяти квантово-устойчивой подписи на основе схемы Штерна следует сократить размеры элементов подписи. Установлено, что наибольший объем памяти в структуре подписи необходим для хранения значений случайных подстановок. Для сокращения этого объема разработан метод компактного задания подстановки с использованием регистров сдвига с линейной обратной связью (РСЛОС).

Регистры сдвига с линейной обратной связью применяются для генерации псевдослучайных битовых последовательностей. Один такт работы РСЛОС представляет собой генерацию одного бита выходной последовательности, расчет значения входного бита и обновление состояния регистра.

Последовательность, генерируемая РСЛОС, определяется характеристическим многочленом и начальным состоянием (вектором инициализации). Максимальный период последовательности для регистра сдвига длины L составляет $2^L - 1$ и достигается при условии, что характеристический многочлен РСЛОС является примитивным многочленом поля $GF(2^L)$. В этом случае состояние регистра для каждого такта не повторяется в течение одного периода работы, и РСЛОС принимает все возможные состояния, кроме нулевого.

Таким образом, при помощи РСЛОС может быть сгенерирована подстановка любой длины $N \le 2^L - 1$. Для этого необходимо записывать в вектор, описывающий подстановку, состояния регистра сдвига, не превышающие значения N, в течение одного периода работы в порядке их возникновения. Полученный вектор значений описывает подстановку заданной длины N.

Оптимальным является выбор значения L, рассчитанного по формуле $L = \lfloor \log_2 N \rfloor$.

Если зафиксировать конкретное значение характеристического многочлена в качестве открытого параметра системы, подстановка длины N может быть компактно задана описанным способом при помощи значения вектора инициализации РСЛОС. Количество различных подстановок, которые могут быть заданы таким способом, составляет N.

Для увеличения мощности множества подстановок, которые могут быть заданы компактным способом при помощи РСЛОС, в настоящем исследовании предлагается метод, названный *расширением подстановки*.

Расширение подстановки — это генерация на основе подстановки длины N подстановки длины $N + 1$ путем дополнения значением $N + 1$ вектора, описывающего подстановку длины N. При этом значение $N + 1$ размещается на произвольной позиции от 0 до N с последующим сдвигом на одну позицию всех значений вектора, находящихся на позициях с номерами, не меньше чем у выбранной.

Таким образом, при помощи двух значений (вектора инициализации (от 1 до N) и номера позиции для расширения (от 0 до N)) может быть задана подстановка длины $N + 1$ из множества мощностью $N(N + 1)$.

В рамках исследования введен параметр γ, который называется *степенью расширения подстановки*. Он описывает количество расширений заданной компактным способом с использованием РСЛОС подстановки. С использованием γ-кратного расширения подстановки на основе подстановки длины $n - \gamma$ может быть выработана подстановка длины n. Подстановка длины $n - \gamma$ может быть компактно описана при помощи одного значения. При этом значение вектора инициализации для РСЛОС может быть объединено с вектором значений длины γ, задающим подстановку длины n на основе подстановки длины $n - \gamma$, в единый информационный вектор.

Таким способом может быть сгенерирована подстановка длины n, заданная при помощи $\gamma + 1$ значений.

Пример.

На основе РСЛОС с характеристическим многочленом $p(x) = x^3 + x + 1$ *и вектором инициализации* IV = 4= 100_2 *может быть сгенерирована подстановка длины 5:* (4, 5, 3, 1, 2).

После этого могут быть выполнены четыре расширения полученной подстановки, которые задаются значениями: [2, 1, 0, 6]. *В результате будет получена подстановка длины 9:* (8, 4, 7, 5, 6, 3, 9, 1, 2).

На основе описанных результатов сформулировано понятие информационного вектора подстановки и разработан алгоритм формирования подстановки по информационному вектору подстановки.

Понятие информационного вектора подстановки определено следующим образом: *информационный вектор подстановки длины n со степенью расширения* γ — это вектор длины $\gamma + 1$ следующего вида:

$$\mathbf{v}_{n,\gamma} = \Big[\big[0..(n-\gamma)\big),\ \big[0..(n-\gamma+1)\big),$$
$$\big[0..(n-\gamma+2)\big),\ \ldots,\ \big[0..(n-1)\big),\ \big[0..n\big)\Big],$$

где $[0\ ..\ a)$ — произвольное целое число x: $0 \le x < a$.

Обозначим $V_{n,\gamma}$ множество всех различных векторов $\mathbf{v}_{n,\gamma}$.

Ниже представлен алгоритм формирования вектора подстановки по информационному вектору подстановки.

PERMUTATION($\mathbf{v}_{n,\gamma}$, $p(x)$: $\deg(p(x)) = \lfloor \log_2(n - \gamma) \rfloor$):

Шаг 1. Построить РСЛОС на основе примитивного многочлена $p(x)$;

Шаг 2. Инициализировать РСЛОС значением $\mathbf{v}_{n,\gamma}[0] + 1$;

Шаг 3. Для i от 0 до $2^{\deg(p(x))} - 1$:

3.1. Считать значение регистра t;

3.2. Если $t \le n - \gamma$ — записать t на i-ю позицию вектора *perm*;

3.3. Выполнить один такт работы РСЛОС.

Шаг 4. Для i от 1 до $\gamma + 1$ получить новое значение вектора *perm* следующим образом:

4.1. Добавить $\mathbf{v}_{n,\gamma}[i]$ первых значений вектора *perm*;

4.2. Добавить значение $\text{len}(perm) + 1$;

4.3. Добавить оставшиеся значения вектора функции перестановки *perm*.

Результат: *perm* — подстановка длины n.

При проведении исследования сформулирована и доказана **теорема** о вложении множества информационных векторов подстановок во множество подстановок.

Отображение множества $V_{n,\gamma}$ в множество элементов симметрической группы S_n с функцией отображения **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$) инъективно.

Доказательство:

1. Результатом выполнения алгоритма **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$) является подстановка.

Работа РСЛОС детерминирована, следовательно, результатом выполнения шага 3 алгоритма является вектор из $n - \gamma$ различных значений от 1 до $n - \gamma$.

Выполнение шага 4 алгоритма представляет собой последовательное дополнение вектора значений, полученного в результате выполнения шага 3, значениями от $n - \gamma + 1$ до n.

Таким образом, результатом работы алгоритма является вектор длины n, содержащий все различные значения от 1 до n, который по определению описывает подстановку длины n.

2. Если $\mathbf{v}_{n,\gamma} \ne \mathbf{v}'_{n,\gamma}$, то **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$) $\ne$ $\ne$ **PERMUTATION**($\mathbf{v}'_{n,\gamma}$, $p(x)$).

Назовем i-ядром подстановки порядок следования элементов от 1 до i данной подстановки.

Если $\mathbf{v}_{n,\gamma}[0] \ne \mathbf{v}'_{n,\gamma}[0]$, то подстановки **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$), **PERMUTATION**($\mathbf{v}'_{n,\gamma}$, $p(x)$) имеют различные $(n - \gamma)$-ядра.

Если $\mathbf{v}_{n,\gamma}[i] \ne \mathbf{v}'_{n,\gamma}[i]$: $1 \le i \le \gamma + 1$, то при выполнении i-й итерации шага 4 алгоритма будет получен различный результат.

Если i-ядро подстановки **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$) не совпадает с i-ядром подстановки **PERMUTATION**($\mathbf{v}'_{n,\gamma}$, $p(x)$), то **PERMUTATION**($\mathbf{v}_{n,\gamma}$, $p(x)$) $\ne$ **PERMUTATION**($\mathbf{v}'_{n,\gamma}$, $p(x)$).

3. Из 1 и 2 следует, что каждому информационному вектору подстановки соответствует подстановка, притом только одна. *Что и требовалось доказать*.

Модифицированная схема цифровой подписи на основе протокола аутентификации Штерна

На основе вышеописанного метода компактного описания подстановки разработана модифицированная схема подписи на основе протокола аутентификации Штерна (МСП СШ). Модификация предлагает генерацию и хранение в составе блока ответов информационных векторов подстановок. Ниже представлено описание данной схемы подписи.

Системные параметры:

— значения n, k;

— значение ω, которое определяет вес Хэмминга секретного ключа;

— двоичная матрица полного ранга $\mathbf{H}^{n \times k}$;

— функция хеширования $h(x)$: $x \to \{0, 1\}^l$;

— значение параметра стойкости δ;

— троичная хеш-функция $f(x)$: $x \to \{0, 1, 2\}^\delta$;

— значение γ, которое определяет степень расширения подстановки;

— примитивный над полем $GF(2)$ многочлен $p(x)$: $\deg(p(x)) = \lfloor \log_2(n - \gamma) \rfloor$.

Ключевая пара:

$sk = \mathbf{s}$: $\{0, 1\}^n$, $\text{wt}(\mathbf{s}) = \omega$;

$pk = \mathbf{y}$: $\mathbf{y} = \mathbf{H}\mathbf{s}^{\text{T}}$.

Алгоритм формирования подписи.

В качестве входных значений передается значение секретного ключа $\mathbf{s}$ и подписываемое сообщение m.

Шаг 1. Повторить δ раз:

1.1. Сгенерировать $\mathbf{u}_i \in \{0, 1\}^n$, $(\mathbf{v}_{n,\gamma})_i \in V_{n,\gamma}$.

1.2. Вычислить $\sigma_i = \mathbf{PERMUTATION}((\mathbf{v}_{n,\gamma})_i, p(x))$.

1.3. Вычислить $c_{0i} = h(\sigma_i \mid \mathbf{H}\mathbf{u}_i^{\text{T}})$.

1.4. Вычислить $c_{1i} = h(\sigma(\mathbf{u}_i))$.

1.5. Вычислить $c_{2i} = h(\sigma(\mathbf{u}_i \oplus \mathbf{s}))$.

1.6. Вычислить $c_i = c_{0i} \mid c_{1i} \mid c_{2i}$.

Шаг 2. Вычислить $c = c_0 \mid \ldots \mid c_{\delta-1}$.

Шаг 3. Вычислить $b = f(c \mid m) = b_0 \mid \ldots \mid b_{\delta-1}$.

Шаг 4. Повторить δ раз:

Для b_i, $(\mathbf{v}_{n,\gamma})_i$, $\mathbf{u}_i$, σ_i вычислить r_i:

если $b_i = 0$, $r_i = \mathbf{u}_i \mid (\mathbf{v}_{n,\gamma})_i$;

если $b_i = 1$, $r_i = \mathbf{u}_i \oplus \mathbf{s} \mid (\mathbf{v}_{n,\gamma})_i$;

если $b_i = 2$, $r_i = \sigma_i(\mathbf{u}_i) \mid \sigma_i(\mathbf{s})$.

Шаг 5. Вычислить $r = r_0 \mid \ldots \mid r_{\delta-1}$.

Шаг 6. Вычислить значение подписи $Sig = c \mid r$.

Алгоритм проверки подписи.

В качестве входных значений передается значение открытого ключа $\mathbf{y}$, сообщение m и значение подписи $Sig = c \mid r$.

Шаг 1. Вычислить значение $b = f(c \mid m) = b_0 \mid \ldots \mid b_{\delta-1}$.

Шаг 2. Для каждого b_i:

Если $b_i = 0$:

– вычислить $\sigma_i = \mathbf{PERMUTATION}((\mathbf{v}_{n,\gamma})_i, p(x))$;

– выполнить проверки

$$\mathbf{c}_{i0} \stackrel{?}{=} h\left(\sigma_i|\mathbf{H}\mathbf{u}_i^{\mathrm{T}}\right),\ c_{i1} \stackrel{?}{=} h\left(\sigma_i\left(\mathbf{u}_i\right)\right).$$

Если $b_i = 1$:

– вычислить $\sigma_i = \mathbf{PERMUTATION}((\mathbf{v}_{n,\gamma})_i, p(x))$;

– выполнить проверки

$$\mathbf{c}_{i0} \stackrel{?}{=} h\left(\sigma_i|\mathbf{H}\left(\mathbf{u}_i \oplus \mathbf{s}\right)^{\mathrm{T}} \oplus \mathbf{y}\right),\ c_{i2} \stackrel{?}{=} h\left(\sigma_i\left(\mathbf{u}_i \oplus \mathbf{s}\right)\right).$$

Если $b_i = 2$, выполнить проверки

$$c_{i1} \stackrel{?}{=} h\left(\sigma_i\left(\mathbf{u}_i\right)\right),\ c_{i2} \stackrel{?}{=} h\left(\sigma_i\left(\mathbf{u}_i \oplus \mathbf{s}\right)\right),\ \mathrm{wt}\left(\sigma(\mathbf{s})\right) \stackrel{?}{=} \omega.$$

Если все проверки пройдены успешно — подпись принимается. Иначе — отклоняется.

Оценка уровня криптографической стойкости модифицированной схемы цифровой подписи на основе протокола аутентификации Штерна

На основе оценки уровня криптографической стойкости СП СШ может быть произведена оценка уровня криптографической стойкости МСП СШ. Произведенная модификация оказывает влияние только на оценку уровня криптографической стойкости при атаке на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$.

Для уточнения оценки уровня криптографической стойкости МСП СШ относительно уровня оценки криптографической стойкости базовой версии в рамках модели наилучшей известной атаки необходимо рассчитать значения вероятности успешной атаки и уровня криптографической стойкости при атаке на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$ способом, аналогичным представленному в формулах (2), (3).

Для проведения оценки уровня криптографической стойкости рассматривается выполнение проверки (1) при условии, что $\sigma_{\mathbf{s}} = (\sigma')^{-1}$, где $\sigma' = \mathbf{PERMUTATION}(\mathbf{v}_{n,\gamma}, p(x))$, $\mathbf{v}_{n,\gamma}$ — случайный информационный вектор из множества $V_{n,\gamma}$.

Необходимость вычисления обратной подстановки обусловлена тем, что в отличие от S_n множество подстановок, соответствующих $V_{n,\gamma}$, в общем случае не является группой.

Вероятность успешной атаки не превышает следующего значения:

$$P_{SM} \le \frac{Q_1 Q_0}{Q}, \qquad (5)$$

где Q_1 — количество подстановок, соответствующих $V_{n,\gamma}$, размещающих ω единиц двоичного вектора $\mathbf{s}$ на ω заданных позиций; Q_0 — количество подстановок, соответствующих $V_{n,\gamma}$, размещающих $n - \omega$ нулей двоичного вектора $\mathbf{s}$ на $n - \omega$ заданных позиций; Q — общее количество возможных подстановок, соответствующих $V_{n,\gamma}$.

В числителе дроби (5) вычисляется предельное возможное количество подстановок, соответствующих $V_{n,\gamma}$, которые переводят вектор $\mathbf{s}$ в вектор $\sigma_i(\mathbf{s})$.

Общее количество возможных подстановок по теореме о вложении множества информационных векторов подстановок во множество подстановок вычисляется как мощность множества $V_{n,\gamma}$ по формуле

$$Q = \prod_{j=0}^{\gamma}(n-j) = \frac{n!}{(n-\gamma-1)!}. \qquad (6)$$

Чтобы вычислить значение Q_1 для МСП СШ, необходимо определить значение q_1 как количество единиц в первых $n - \gamma$ разрядах вектора $\mathbf{s}$.

$(n - \gamma)$-ядро подстановки в МСП СШ определяется РСЛОС. В симметрической группе S_n для q_1 значений $(n - \gamma)$-ядра найдутся подстановки, которые соответствуют $q_1!$ различным способам разместить q_1 значений вектора, описывающего функцию перестановки. Для подстановок, которые соответствуют множеству информационных векторов подстановок $V_{n,\gamma}$, необходимо исключить все такие способы, кроме циклических.

Таким образом, количество способов поместить ω единиц двоичного вектора $\mathbf{s}$ на ω позиций для МСП СШ вычисляется по формуле

$$Q_1 = \frac{\omega!}{\frac{q_1!}{q_1}} = \frac{\omega!}{(q_1-1)!}. \qquad (7)$$

Аналогично, при вычислении Q_0 для МСП СШ необходимо определить значение q_0 как количество нулей в первых $n - \gamma$ разрядах вектора $\mathbf{s}$.

Количество нулей в первых $n - \gamma$ разрядах вектора $\mathbf{s}$ вычисляется по формуле $q_0 = n - \gamma - q_1$.

По аналогии с вычислением значения Q_1, количество способов поместить $n - \omega$ нулей двоичного вектора $\mathbf{s}$ на $n - \omega$ позиций для МСП СШ определяется по формуле

$$Q_0 = \frac{(n-\omega)!}{(q_0-1)!} = \frac{(n-\omega)!}{(n-\gamma-q_1-1)!}. \qquad (8)$$

Оценка, вычисляемая по формулам (5)–(8), является завышенной, так как формируется из следующих предположений:

— среди подстановок, соответствующих $V_{n,\gamma}$, найдутся Q_1 способов размещений единиц на

ω произвольных позиций для любого значения вектора **s**;

– среди подстановок, соответствующих $V_{n,\gamma}$, найдутся Q_0 способов размещения нулей вектора **s** для каждого возможного способа размещения единиц.

Вероятность успешной атаки на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$ для МСП СШ не превышает

$$P_{SM} \le \frac{Q_1 Q_0}{Q} = \frac{\frac{\omega!}{(q_1-1)!}\cdot\frac{(n-\omega)!}{(n-\gamma-q_1-1)!}}{\frac{n!}{(n-\gamma-1)!}} =$$

$$= \frac{\omega!(n-\omega)!(n-\gamma-1)!}{n!(q_1-1)!(n-\gamma-q_1-1)!} =$$

$$= P_S \cdot \frac{(n-\gamma-1)!}{(q_1-1)!(n-\gamma-q_1-1)!},$$

где P_S – вероятность успешной атаки на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$ для СП СШ, вычисленная по формуле (2).

Тогда уровень криптографической стойкости при атаке на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$ для МСП СШ определяется по формуле

$$\lambda_\sigma = -\log_2 P_{SM} = \log_2 \frac{n!(q_1-1)!(n-\gamma-q_1-1)!}{\omega!(n-\omega)!(n-\gamma-1)!}. \quad (9)$$

По формуле (9) значение системного параметра γ может быть подобрано исходя из выбранного значения λ_σ.

Общий уровень криптографической стойкости, измеренный в битах, для МСП СШ может быть вычислен по формуле (4).

Сравнительный анализ модифицированной и базовой схем подписи на основе протокола аутентификации Штерна

В ходе исследования был проведен сравнительный анализ функциональных характеристик СП СШ и предложенной модифицированной версии. Под функциональными характеристиками понимается размер формируемой подписи и производительность алгоритмов схемы подписи.

Для выполнения сравнения размера подписи необходимо вычислить размеры блоков структуры МСП СШ и сравнить их с соответствующими значениями для СП СШ, приведенными в табл. 1.

Чтобы вычислить размеры блоков, необходимо определить значения системных параметров для МСП СШ на основе исследуемых наборов значений системных параметров [25, 23]. Для этого по формулам (6)–(9) подобраны наименьшие значения степени расширения подстановки. Эти значения вычислены исходя из требования обеспечения уровня криптографической стойкости при атаке на подбор значения секретного ключа по известному значению $\sigma_i(\mathbf{s})$ не ниже общего уровня криптографической стойкости для каждого набора, представленного в табл. 1.

Для набора № 1 исходя из $\lambda_\sigma = \lambda = 54$ определено значение $\gamma = 268$. Для набора № 2, аналогично, ($\lambda_\sigma = \lambda = 80$) определено значение $\gamma = 478$.

В отличие от СП СШ для модифицированной версии схемы подписи размер блока r_i при $b_i \in \{0, 1\}$ составляет $n + (\gamma + 1) \cdot \log_2 n$ [бит].

Сопоставлены размеры блоков структуры подписи и значения характеристик криптографической стойкости, претерпевшие изменения в результате модификации (см. табл. 2). В графе «Коэффициент» табл. 2 рассчитаны относительные значения показателей, в качестве базовых рассматриваются значения характеристик СП СШ.

Из табл. 2 следует, что разработанная модификация позволяет значительно сократить размеры затронутых модификацией блоков структуры подписи и, как следствие, размеры подписи в целом. Для набора значений системных параметров № 1 в результате модификации размер подписи уменьшился в 1,96 раза, для набора зна-

■ ***Таблица 2.*** Сравнение показателей СП СШ и МСП СШ

■ ***Table 2.*** Basic and modified versions Stern-based digital signature comparison

Показатель	СП СШ	МСП СШ	Коэффициент
Набор № 1			
γ	–	268	–
$\lvert r_i, \sigma\rvert$, бит	6820	**3310**	**0,485**
$\lvert r\rvert$, Кбайт	114,06	**55,36**	**0,485**
$\lvert Sig\rvert$, Кбайт	119,67	**60,97**	**0,510**
λ_σ, бит	305	54	–
λ, бит	54	54	–
Набор № 2			
γ	–	478	–
$\lvert r_i, \sigma\rvert$, бит	37648	**8644**	**0,230**
$\lvert r\rvert$, Кбайт	629,61	**144,56**	**0,230**
$\lvert Sig\rvert$, Кбайт	655,30	**170,25**	**0,260**
λ_σ, бит	1440	80	–
λ, бит	80	80	–

чений системных параметров № 2 — в 3,85 раза соответственно.

При этом предложенная модификация не влияет на общий уровень криптографической стойкости схемы подписи с учетом выбранной модели оценки.

Для сравнения времени выполнения алгоритма формирования подписи и алгоритма проверки подписи выполнена программная реализация СП СШ и МСП СШ. В качестве среды реализации выбрана система компьютерной алгебры SageMath 9.3.

По результатам проведения серии экспериментов для набора значений системных параметров № 1, представленного в табл. 1, наблюдается увеличение времени работы алгоритмов МСП СШ относительно алгоритмов СП СШ: в 1,11 раза для алгоритма формирования подписи и в 1,32 раза — для алгоритма проверки.

Для набора значений системных параметров № 2, представленного в табл. 1, была проведена ограниченная серия экспериментов. В результате модификации для набора № 2 наблюдается увеличение времени работы алгоритма формирования подписи в 1,08 раза, алгоритма проверки подписи — в 1,38 раза.

Заключение

Уменьшение размеров формируемой подписи для модифицированной версии схемы цифровой подписи (в 1,96 раза для набора значений системных параметров, обеспечивающих криптографическую стойкость на уровне 54 бита, в 3,85 раза для криптографической стойкости на уровне 80 бит соответственно) является значимым в контексте недостатков схемы подписи на основе протокола аутентификации Штерна, а также абсолютного значения снижения размеров подписи. При этом снижение производительности алгоритмов схемы подписи не является значимым в связи с особенностями применения схем цифровой подписи в информационных системах. Необходимо отметить, что увеличение времени работы алгоритмов схемы подписи связано с особенностями инструментов программной реализации и использованием высокоуровневого языка программирования. При выполнении программной реализации на языке системного программирования данное отставание может быть нивелировано.

При этом для практического использования предложенной модифицированной версии необходимо определить уровень ее стойкости в модели доказуемой стойкости.

Для дальнейших исследований в рамках предметной области могут быть выбраны следующие направления:

— уточнение оценки уровня криптостойкости модифицированной схемы цифровой подписи на основе протокола аутентификации Штерна (в том числе с использованием модели доказуемой стойкости);

— разработка других методов оптимизации по памяти для схемы подписи на основе протокола аутентификации Штерна;

— применение предложенного метода компактного описания подстановки для оптимизации по памяти схем подписи на основе других протоколов аутентификации без разглашения на основе произвольных линейных кодов.

Финансовая поддержка

Работа выполнена в рамках государственного задания (проект FSER20250003).

Благодарности

Автор выражает благодарность Вадиму Валерьевичу Давыдову, Андрею Андреевичу Голованову и Сергею Валентиновичу Беззатееву за помощь при проведении исследования, Ивану Владимировичу Чижову за предоставленный отзыв на результаты исследования, Юлии Викторовне Ниткиной за помощь при работе с текстом статьи.

Литература

1. **Boudot F., Gaudry P., Guillevic A., Heninger N., Thomé E., Zimmermann P.** The state of the art in integer factoring and breaking public-key cryptography. *IEEE Security & Privacy*, 2022, vol. 20, no. 2, pp. 80–86. doi:10.1109/MSEC.2022.3141512
2. **Shor P. W.** Algorithms for quantum computation: Discrete logarithms and factoring. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.
3. ГОСТ 34.10-2018. *Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи*. М., Стандартинформ, 2018. 21 с.
4. *Post-Quantum Cryptography | CSRC*. https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization/Call-for-Proposals (дата обращения: 01.06.2025)
5. *Post-Quantum Cryptography: Additional Digital Signature Schemes | CSRC*. https://csrc.nist.gov/projects/

pqc-dig-sig/standardization/call-for-proposals (дата обращения: 01.06.2025)
6. **Alagic G., Bros M., Ciadoux P., Cooper D., Dang Q., Dang T., Kelsey J., Lichtinger J., Liu Y.-K., Miller C., Moody D., Peralta R., Perlner R., Robinson A., Silberg H., Smith-Tone D., Waller N.** Status report on the fourth round of the NIST post-quantum cryptography standardization process. *NIST IR 8545*, 2025. doi:10.6028/NIST.IR.8545
7. **Chailloux A., Etinski S.** On the (in)security of optimized Stern-like signature schemes. *Designs, Codes and Cryptography*, 2024, no. 92, pp. 803–832. doi:10.1007/s10623-023-01329-y
8. **Weger V., Gassner N., Rosenthal J.** *A Survey on code-based cryptography*. 2022. https://arxiv.org/pdf/2201.07119 (дата обращения: 01.06.2025). doi:10.48550/arXiv.2201.07119
9. **McEliece R. J.** A public-key cryptosystem based on algebraic coding theory. *DSN Progress Report, Jet Propulsion Laboratory, Pasadena*, 1978, pp. 114–116.
10. **Courtois N., Finiasz M., Sendrier N.** How to achieve a McEliece-based digital signature scheme. *Advances in Cryptology — ASIACRYPT 2001*, 2001, pp. 157–174. doi:10.1007/3-540-45682-1_8
11. **Stern J.** A new identification scheme based on syndrome decoding. *Advances in Cryptology — CRYPTO'93*, 1993, pp. 13–21. doi:10.1007/3-540-48329-2_2
12. **Véron P.** Improved identification schemes based on error-correcting codes. *Applicable Algebra in Engineering, Communication and Computing*, 1996, no. 8, pp. 57–69. doi:10.1007/BF01190881
13. **Cayrel P., Véron P., El Y. A. S. M.** A zero-knowledge identification scheme based on the q-ary syndrome decoding problem. *17th International Workshop, SAC 2010*, 2010, pp. 171–186. doi:10.1007/978-3-642-19574-7_12
14. **Jain A., Krenn S., Pietrzak K., Tentes A.** Commitments and efficient zero-knowledge proofs from learning parity with noise. *Advances in Cryptology — ASIACRYPT 2012*, 2012, pp. 663–680. doi:10.1007/978-3-642-34961-4_40
15. **Feneuil T., Joux A., Rivain M.** Syndrome decoding in the head: Shorter signatures from zero-knowledge proofs. *Advances in Cryptology — CRYPTO 2022*, 2022, pp. 541–572. doi:10.1007/978-3-031-15979-4_19
16. **Feneuil T., Joux A., Rivain M.** Shared permutation for syndrome decoding: new zero-knowledge protocol and code-based signature. *Designs, Codes and Cryptography*, 2023, no. 91, pp. 563–608. doi:10.1007/s10623-022-01116-1
17. **Baldi M., Bitzer S., Pavoni A., Santini P., Wachter-Zeh A., Weger V.** Zero knowledge protocols and signatures from the restricted syndrome decoding problem. *14th International Conference on Post-Quantum Cryptography (PQCrypto 2023)*, 2023.
18. **Baldi M., Battaglioni M., Chiaraluce F., Horlemann-Trautmann A.-L., Persichetti E., Santini P., Weger V.** A new path to code-based signatures via identification schemes with restricted errors. *Advances in Mathematics of Communications*, 2025, no. 19(5), pp. 1360–1381. doi:10.3934/amc.2024058
19. **Manganiello F., Slaughter F.** Generic error SDP and generic error CVE. *Code-Based Cryptography — CBCrypto 2023*, 2023, pp. 125–143. doi:10.1007/978-3-031-46495-9_7
20. **Fiat A., Shamir A.** How to prove yourself: Practical solutions to identification and signature problems. *Advances in Cryptology — CRYPTO'86*, 1986, pp. 186–194. doi:10.1007/3-540-47721-7_12
21. **Bidoux L., Gaborit P., Kulkarni M., Mateu V.** Code-based signatures from new proofs of knowledge for the syndrome decoding problem. *Designs, Codes and Cryptography*, 2023, no. 91, pp. 497–544. doi:10.1007/s10623-022-01114-3
22. **Alagic G., Bros M., Ciadoux P., Cooper D., Dang Q., Dang T., Kelsey J., Lichtinger J., Liu Y.-K., Miller C., Moody D., Peralta R., Perlner R., Robinson A., Silberg H., Smith-Tone D., Waller N.** Status report on the first round of the additional digital signature schemes for the NIST post-quantum cryptography standardization process. *NIST IR 8528*, 2024. doi:10.6028/NIST.IR.8528
23. **Vysotskaya V. V., Chizhov I. V.** The security of the code-based signature scheme based on the stern identification protocol. *Прикладная дискретная математика*, 2022, № 57, с. 67–90. doi:10.17223/20710410/57/5
24. **Esser A., Bellini E.** Syndrome decoding estimator. *Public-Key Cryptography — PKC 2022*, 2022, pp. 112–141. doi:10.1007/978-3-030-97121-2_5
25. **Roy P. S., Morozov K., Fukushima K., Kiyomoto S.** Evaluation of code-based signature schemes. *Cryptology ePrint Archive*, 2019. https://eprint.iacr.org/2019/544 (дата обращения: 01.06.2025).
26. **Bernstein D. J.** Grover vs. McEliece. *Post-Quantum Cryptography (PQCrypto 2010)*, 2010, pp. 73–80. doi:10.1007/978-3-642-12929-2
27. **Kirshanova E.** Improved quantum information set decoding. *Post-Quantum Cryptography (PQCrypto 2018)*, 2018, pp. 507–527. doi:10.1007/978-3-319-79063-3_24
28. **Grover L. K.** A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, 1996, pp. 212–219.
29. **Both L., May A.** Decoding linear codes with high error rate and its impact for LPN security. *Post-Quantum Cryptography (PQCrypto 2018)*, 2018, pp. 25–46. doi:10.1007/978-3-319-79063-3_2
30. **Debris-Alazard T., Ducas L.,van Woerden W. P. J.** An algorithmic reduction theory for binary codes: LLL and more. *IEEE Transactions on Information Theory*, 2022, vol. 68, no. 5, pp. 3426–3444. doi:10.1109/TIT.2022.3143620

UDC 003.26
doi:10.31799/1684-8853-2025-6-51-63
EDN: AEXSDC

Permutation compact description method and its application for Stern-based digital signature modification

I. S. Nitkin[a,b], Post-Graduate Student, orcid.org/0000-0001-5240-1744, exebopen@gmail.com
[a]ITMO University, 49, Kronverksky Pr., 197101, Saint-Petersburg, Russian Federation
[b]Saint-Petersburg State University of Aerospace Instrumentation, 67, B. Morskaia St., 190000, Saint-Petersburg, Russian Federation

Introduction: The Stern authentication scheme is foundational zero-knowledge protocols for constructing quantum-resistant code-based digital signatures. However, the key limitation of this approach is the significant size of the generated signature. **Purpose:** To develop a memory-optimized Stern-based digital signature. **Results:** The study investigates the Stern-based digital signature, analyzing the sizes of its structural elements, and signature system parameters to its block size relation. We conclude that the largest memory consumption is required for storing random permutation values. We perform the cryptographic security analysis using the best-known attack model for Stern-based digital signature. We develop a method of permutation compact description using an information vector and apply it to Stern-based digital signature modification. The cryptographic security of the modified Stern-based digital signature using the same model is evaluated. A theoretical and experimental comparative analysis of the functional characteristics between the original and modified signature schemes demonstrates that the proposed modification significantly reduces generated signature size while maintaining the overall cryptographic strength of the scheme according to the chosen evaluation model. **Practical relevance:** The results can be applied to memory optimization for signature schemes using other code-based zero-knowledge protocols, as well as to the development of further optimization methods for Stern-based digital signatures.

Keywords — post-quantum cryptography, code-based cryptography, digital signature, Stern scheme, Stern-based signature.

For citation: Nitkin I. S. Permutation compact description method and its application for Stern-based digital signature modification. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 51–63 (In Russian). doi:10.31799/1684-8853-2025-6-51-63, EDN: AEXSDC

Financial support
The work was performed within the framework of the state assignment (project FSER-2025-0003).

References

1. Boudot F., Gaudry P., Guillevic A., Heninger N., Thomé E., Zimmermann P. The state of the art in integer factoring and breaking public-key cryptography. *IEEE Security & Privacy*, 2022, vol. 20, no. 2, pp. 80–86. doi:10.1109/MSEC.2022.3141512
2. Shor P. W. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science (FOCS'94)*, 1994, pp. 124–134.
3. State Standard 34.10-2018. *Informatsionnaia tekhnologiia. Kriptograficheskaia zashchita informatsii. Protsessy formirovaniia i proverki elektronnoi tsifrovoi podpisi* [Information technology. Cryptographic data security. Processes of digital signature generation and verification]. Moscow, Standartinform Publ., 2018. 21 p. (In Russian).
4. *Post-Quantum Cryptography | CSRC*. Available at: https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization/Call-for-Proposals (accessed 1 June 2025)
5. *Post-Quantum Cryptography: Additional Digital Signature Schemes | CSRC*. Available at: https://csrc.nist.gov/projects/pqc-dig-sig/standardization/call-for-proposals (accessed 1 June 2025)
6. Alagic G., Bros M., Ciadoux P., Cooper D., Dang Q., Dang T., Kelsey J., Lichtinger J., Liu Y.-K., Miller C., Moody D., Peralta R., Perlner R., Robinson A., Silberg H., Smith-Tone D., Waller N. Status report on the fourth round of the NIST post-quantum cryptography standardization process. *NIST IR 8545*, 2025. doi.org/10.6028/NIST.IR.8545
7. Chailloux A., Etinski S. On the (in)security of optimized Stern-like signature schemes. *Designs, Codes and Cryptography*, 2024, no. 92, pp. 803–832. doi:10.1007/s10623-023-01329-y
8. Weger V., Gassner N., Rosenthal J. A survey on code-based cryptography, 2022. Available at: https://arxiv.org/pdf/2201.07119 (accessed 1 June 2025). doi:10.48550/arXiv.2201.07119
9. McEliece R. J. A public-key cryptosystem based on algebraic coding theory. *DSN Progress Report, Jet Propulsion Laboratory, Pasadena*, 1978, pp. 114–116.
10. Courtois N., Finiasz M., Sendrier N. How to achieve a McEliece-based digital signature scheme. *Advances in Cryptology — ASIACRYPT 2001*, 2001, pp. 157–174. doi:10.1007/3-540-45682-1_8
11. Stern J. A new identification scheme based on syndrome decoding. *Advances in Cryptology — CRYPTO'93*, 1993, pp. 13–21. doi:10.1007/3-540-48329-2_2
12. Véron P. Improved identification schemes based on error-correcting codes. *Applicable Algebra in Engineering, Communication and Computing*, 1996, no. 8, pp. 57–69. doi:10.1007/BF01190881
13. Cayrel P., Véron P., El Y. A. S. M. A zero-knowledge identification scheme based on the q-ary syndrome decoding problem. *17th International Workshop: Selected Areas in Cryptography — SAC 2010*, 2010, pp. 171–186. doi:10.1007/978-3-642-19574-7_12
14. Jain A., Krenn S., Pietrzak K., Tentes A. Commitments and efficient zero-knowledge proofs from learning parity with noise. *Advances in Cryptology — ASIACRYPT 2012*, 2012, pp. 663–680. doi:10.1007/978-3-642-34961-4_40
15. Feneuil T., Joux A., Rivain M. Shared permutation for syndrome decoding: New zero-knowledge protocol and code-based signature. *Designs, Codes and Cryptography*, 2023, no. 91, pp. 563–608. doi:10.1007/s10623-022-01116-1
16. Feneuil T., Joux A., Rivain M. Syndrome decoding in the head: Shorter signatures from zero-knowledge proofs. *Advances in Cryptology — CRYPTO 2022*, 2022, pp. 541–572. doi:10.1007/978-3-031-15979-4_19
17. Baldi M., Bitzer S., Pavoni A., Santini P., Wachter-Zeh A., Weger V. Zero knowledge protocols and signatures from the restricted syndrome decoding problem. *14th International Conference on Post-Quantum Cryptography — PQCrypto 2023*, 2023.
18. Baldi M., Battaglioni M., Chiaraluce F., Horlemann-Trautmann A.-L., Persichetti E., Santini P., Weger V. A new path to code-based signatures via identification schemes with restricted errors. *Advances in Mathematics of Communications*, 2025, no. 19(5), pp. 1360–1381. doi:10.3934/amc.2024058
19. Manganiello F., Slaughter F. Generic error SDP and generic error CVE. *Code-Based Cryptography — CBCrypto 2023*, 2023, pp. 125–143. doi:10.1007/978-3-031-46495-9_7
20. Fiat A., Shamir A. How to prove yourself: Practical solutions to identification and signature problems. *Advances in Cryptology — CRYPTO'86*, 1986, pp. 186–194. doi:10.1007/3-540-47721-7_12
21. Bidoux L., Gaborit P., Kulkarni M., Mateu V. Code-based signatures from new proofs of knowledge for the syn-

drome decoding problem. *Designs, Codes and Cryptography*, 2023, no. 91, pp. 497–544. doi:10.1007/s10623-022-01114-3
22. Alagic G., Bros M., Ciadoux P., Cooper D., Dang Q., Dang T., Kelsey J., Lichtinger J., Liu Y.-K., Miller C., Moody D., Peralta R., Perlner R., Robinson A., Silberg H., Smith-Tone D., Waller N. Status report on the first round of the additional digital signature schemes for the NIST post-quantum cryptography standardization process, *NIST IR 8528*, 2024. doi:10.6028/NIST.IR.8528
23. Vysotskaya V. V., Chizhov I. V. The security of the code-based signature scheme based on the Stern identification protocol. *Prikladnaya diskretnaya matematika*, 2022, no. 57, pp. 67–90. doi:10.17223/20710410/57/5
24. Esser A., Bellini E. Syndrome decoding estimator. *Public-Key Cryptography – PKC 2022*, 2022, pp. 112–141. doi:10.1007/978-3-030-97121-2_5
25. Roy P. S., Morozov K., Fukushima K., Kiyomoto S. Evaluation of code-based signature schemes. *Cryptology ePrint Archive*, 2019. Available at: https://eprint.iacr.org/2019/544 (accessed 1 June 2025)
26. Bernstein D. J. Grover vs. McEliece. *Post-Quantum Cryptography – PQCrypto 2010*, 2010, pp. 73–80. doi:10.1007/978-3-642-12929-2_6
27. Kirshanova E. Improved quantum information set decoding. *Post-Quantum Cryptography – PQCrypto 2018*, 2018, pp. 507–527. doi:10.1007/978-3-319-79063-3_24
28. Grover L. K. A fast quantum mechanical algorithm for database search. *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, 1996, pp. 212–219.
29. Both L., May A. Decoding linear codes with high error rate and its impact for LPN security. *Post-Quantum Cryptography – PQCrypto 2018*, 2018, pp. 25–46. doi:10.1007/978-3-319-79063-3_2
30. Debris-Alazard T., Ducas L.,van Woerden W. P. J. An algorithmic reduction theory for binary codes: LLL and more. *IEEE Transactions on Information Theory*, 2022, vol. 68, no. 5, pp. 3426–3444. doi:10.1109/TIT.2022.3143620

УДК 519.72
doi:10.31799/1684-8853-2025-6-64-73
EDN: ESMBYH

Научные статьи
Articles

Методика построения информационных совокупностей с неравномерным разбиением для исправления пакетов ошибок

М. Н. Исаева[a], *старший преподаватель, orcid.org/0009-0007-6228-0617, imn@guap.ru*
[a]*Санкт-Петербургский государственный университет аэрокосмического приборостроения, Б. Морская ул., 67, Санкт-Петербург, 190000, РФ*

Введение: *задача исправления ошибок является актуальной для современных систем связи и хранения данных, особенно в каналах, для которых характерно группирование ошибок в пакеты. Малоизученной остается задача исправления более чем одного пакета ошибок, образующегося при передаче блока данных.* ***Цель:*** *разработать и проанализировать методику генерации множества информационных совокупностей для исправления двукратных пакетов ошибок.* ***Результаты:*** *в ходе исследования установлено, что плотные информационные совокупности, исследовавшиеся ранее для исправления однократных пакетов ошибок, применимы для исправления двукратных пакетов только для кодов с малой скоростью. Для кодов с более высокими скоростями проанализированы методики построения множества информационных совокупностей с различными критериями разбиения и предложена методика, использующая динамическое неравномерное разбиение. Полученные с ее помощью информационные совокупности делают возможным исправление любых комбинаций из не более чем двукратных пакетов ошибок, длина которых находится в рамках модифицированной границы Рейгера.* ***Практическая значимость:*** *результаты работы имеют практическое значение в проектировании систем связи для передачи по каналам с памятью, где частота появления пакетов достаточно велика для образования множественных пакетов за время передачи одного кодового слова. Предложенная методика позволяет повысить помехозащищенность таких каналов и может быть использована для разработки вычислительно эффективных декодеров.* ***Обсуждение:*** *результаты работы получены в предположении, что любые последовательные позиции кодового слова образуют информационную совокупность, а также что длины исправляемых пакетов лежат на модифицированной границе Рейгера. Можно ожидать, что эти два эффекта в некоторой степени компенсируют друг друга с точки зрения требований ко множествам информационных совокупностей для конкретных кодов, однако оценка параметров множеств информационных совокупностей и построение декодеров на их основе для отдельных классов кодов является направлением дальнейших исследований.*

Ключевые слова *— декодирование по информационным совокупностям, граница Рейгера, исправление многократных пакетов ошибок, плотные информационные совокупности, каналы с памятью.*

Для цитирования: Исаева М. Н. Методика построения информационных совокупностей с неравномерным разбиением для исправления пакетов ошибок. *Информационно-управляющие системы*, 2025, № 6, с. 64–73. doi:10.31799/1684-8853-2025-6-64-73, EDN: ESMBYH

For citation: Isaeva M. N. Methodology for constructing information sets with non-uniform partitioning for error burst correction. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 64–73 (In Russian). doi:10.31799/1684-8853-2025-6-64-73, EDN: ESMBYH

Введение

Задача повышения надежности передачи данных в каналах связи, подверженных ошибкам, является одной из классических в теории помехоустойчивого кодирования. В то время как для каналов с независимыми ошибками разработаны и хорошо изучены методы помехоустойчивого кодирования, многие реальные системы сталкиваются с более сложной структурой ошибок. Каналы беспроводной связи, каналы с замираниями, системы хранения данных характеризуются наличием памяти, что приводит к коррелированному возникновению ошибок, которые группируются в пакеты ошибок.

Задача исправления однократных пакетов ошибок может решаться на основе кодов со специальной алгебраической структурой, таких как циклические коды Файра, коды Рида — Соломона, итеративные конструкции на их основе [1–5]. Однако во многих практических сценариях за время передачи одного кодового слова может возникать несколько независимых пакетов ошибок [6], этот случай менее изучен по сравнению с исправлением однократных пакетов. Проблема кодирования в каналах с памятью активно исследуется в современной литературе.

В работе [7] исследуются современные подходы к построению композиции кодов с малой плотностью проверок на четность (МППЧ-коды), в [8] решается задача детектирования пакетов в кодовых словах МППЧ-кодов в специальном случае канала, образованного марковской цепью с гауссовыми каналами в состо-

яниях цепи. В [9] предлагается декодер исправления одиночных пакетов для произвольных линейных кодов, однако имеющий практически реализуемую сложность лишь для кодов с разреженными проверочными матрицами с блоковой структурой. В [10, 11] затрагиваются вопросы кодирования полярными кодами для различных каналов, в том числе с памятью. В [12] предлагается декодер полярных кодов для классического канала с памятью, задаваемого моделью Гилберта — Эллиота с неизвестными параметрами. В [13] рассматривается исправление пакетов ошибок, расположенных всегда в начале слова. В [14, 15] предлагаются методы исправления пакетов (в том числе множественных) ошибок и стираний для случая их фиксированного расположения (фазированные пакеты). В [16] ставятся и решаются вопросы построения метрик для каналов с памятью с двумя состояниями. В [17, 18] исследуется передача по каналам специального вида, память в которых обуславливается межсимвольной интерференцией и замираниями. В [19] разрабатывается подход к исправлению однократных пакетов ошибок на основе информационных совокупностей и показано, что этот метод может обладать достаточно низкой вычислительной сложностью за счет использования специальным образом построенного множества информационных совокупностей.

Целью настоящей работы является расширение и обобщение подхода с использованием информационных совокупностей на случай исправления двукратных пакетов ошибок. В статье предлагается методика построения множества информационных совокупностей для исправления двух пакетов ошибок, проводится анализ возможности уменьшения мощности полученного множества.

Статья использует следующую структуру. Вводятся основные понятия помехоустойчивого кодирования, в том числе при исправлении пакетов ошибок, используемые в дальнейшем в статье. Дается определение Δ-плотных информационных совокупностей. Рассматривается возможность использования однократных плотных информационных совокупностей, равномерного разбиения информационных совокупностей, а также неравномерного разбиения для исправления двукратных пакетов. Анализируются границы применимости каждой из методик. На основе полученных результатов предлагается методика построения множества информационных совокупностей для исправления двукратных пакетов ошибок. Проводится оценка количества требуемых информационных совокупностей для исправления пакетов ошибок в пределах границы Рейгера.

Основные понятия

Для дальнейшего изложения введем ряд формальных определений [20]. Линейный двоичный (n, k)-код определяется как k-мерное подпространство в n-мерном векторном пространстве всех векторов над полем $GF(2)$. Для задания линейных кодов используются порождающая и проверочная матрицы. Порождающая $(k \times n)$-матрица $\mathbf{G}$ ранга k используется для построения кодовых слов из информационных векторов $\mathbf{m}$ в соответствии с правилом $\mathbf{a} = \mathbf{mG}$. Для проверки корректности принятых из канала данных используется проверочная $(r \times n)$-матрица $\mathbf{H}$, где $r = n - k$. Она связана с $\mathbf{G}$ условием ортогональности $\mathbf{GH}^{\mathrm{T}} = \mathbf{0}$ и позволяет вычислить для любого принятого вектора $\mathbf{b}$ его синдром $\mathbf{s} = \mathbf{bH}^{\mathrm{T}}$. В рамках стандартной аддитивной модели канала принятый вектор можно представить в виде $\mathbf{b} = \mathbf{a} \oplus \mathbf{e}$, где $\mathbf{e}$ — вектор ошибок, в котором единицы соответствуют ошибочным битам. Если синдром равен нулю, вектор $\mathbf{b}$ является кодовым словом и считается, что ошибок не произошло. В противном случае обнаруживается ошибка и запускается процесс декодирования. Структура вектора ошибок $\mathbf{e}$ определяется статистическими свойствами канала связи.

Для каналов без памяти, например двоичного симметричного канала, ошибки являются независимыми событиями, в этом случае ненулевые компоненты вектора $\mathbf{e}$ распределены равномерным образом по символам принятого слова. Однако, как уже было сказано, для многих практических каналов с памятью ошибки имеют тенденцию группироваться в пакеты ошибок. Вектор $\mathbf{e}$ называется однократным пакетом ошибок длиной b, если все его ненулевые компоненты содержатся только в b последовательно расположенных позициях. Двукратный пакет ошибок длиной b может быть определен как вектор $\mathbf{e}$, который можно представить в виде суммы по модулю 2 двух одиночных пакетов ошибок $\mathbf{e}_1$ и $\mathbf{e}_2$ длиной b каждый: $\mathbf{e} = \mathbf{e}_1 \oplus \mathbf{e}_2$.

Корректирующая способность кода при рассмотрении случайных независимых ошибок определяется минимальным расстоянием Хэмминга d_0, способность кода исправлять любую комбинацию до t ошибок оценивается неравенством $d_0 \geq 2t + 1$. При декодировании группированных ошибок в качестве корректирующей способности кода может быть использована минимальная длина исправляемого пакета ошибок b_0, которая может быть определена с полиномиальной сложностью согласно процедуре, описанной в [21]. Для оценки предельной способности кода исправлять пакеты ошибок используется граница Рейгера, которая устанавливает необходимое условие на количество проверочных сим-

волов $r = n - k$ для гарантированного исправления любого однократного пакета ошибок длиной до b_0 включительно: $2b_0 \le r$. Следует отметить, что данная граница является необходимым, но недостаточным условием, и многие кодовые конструкции лежат ниже этой границы. В случае исправления двукратных пакетов может быть сформулирована модифицированная граница Рейгера, в соответствии с которой длина исправляемого двукратного пакета не может превысить $b_0 \le \lfloor (n-k)/4 \rfloor$. Можно заметить, что если код исправляет все двукратные пакеты длиной до $r/4$, то он также исправляет одиночные пакеты длиной до $r/2$, так как такой пакет является конкатенацией двух более коротких пакетов. Таким образом, алгоритмы декодирования двукратных пакетов длиной на границе Рейгера исправляют также любые однократные пакеты.

Информационной совокупностью линейного (n, k)-кода называется любое множество из k позиций $\chi = \{1 \le i_1 < i_2 < \ldots < i_k \le n\}$, однозначно задающее кодовое слово. Для того чтобы χ являлось информационной совокупностью, необходимо и достаточно, чтобы соответствующие столбцы порождающей матрицы **G** были линейно независимы. Если ошибочные символы не попали на позиции информационной совокупности, то такая информационная совокупность называется свободной от ошибок, и с ее помощью можно корректно исправить принятый вектор. Поскольку места ошибок неизвестны, стратегия декодирования может заключаться в переборе по некоторому множеству X информационных совокупностей, заранее построенному так, чтобы для любых исправляемых векторов ошибки нашлась не искаженная информационная совокупность. Для уменьшения сложности декодирования размер множества X необходимо по возможности минимизировать.

В [19] исследуется применение информационных совокупностей для исправления однократных пакетов ошибок. Для минимизации количества информационных совокупностей, необходимых для декодирования, вводится понятие Δ-плотной информационной совокупности (при $\Delta = 0$ символ «Δ» опускается), т. е. из $k + \Delta$ подряд идущих позиций.

Обратим внимание, что позиции плотной информационной совокупности выбираются циклически, с учетом этого максимальное множество плотных информационных совокупностей имеет размерность n. Отметим, что в литературе иногда используется понятие циклического пакета, для исправления которого используются циклические коды. В данной статье циклический пакет не рассматривается, так как исследование не ограничивается циклическими кодами и, кроме того, однократный циклический пакет описывается моделью с двумя пакетами.

В [19] предложена методика построения множества плотных информационных совокупностей, позволяющая снизить их количество с $O(n)$ до $O(1)$ с гарантированным исправлением пакетов в рамках корректирующей способности кода. Однако следует заметить, что конкретные значения этого количества зависят от скорости кода $R = k/n$.

Необходимо отметить, что не любые k подряд идущих позиций образуют информационную совокупность. Для сохранения подхода, основанного на использовании информационных совокупностей ограниченного диаметра, понятие плотной информационной совокупности может быть расширено до $k + \Delta$ позиций. Вероятность нахождения таких информационных совокупностей в зависимости от Δ и параметров используемого кода оценена в [22]. В данной статье используется упрощенное предположение о том, что любые k позиций образуют информационную совокупность.

Использование плотных информационных совокупностей для исправления двукратных пакетов ошибок

Изучим возможность применения подхода, основанного на плотных информационных совокупностях, описанного в предыдущем разделе, для исправления двукратных пакетов ошибок. С учетом цикличности выбора позиций информационных совокупностей рассмотрим схематическое изображение пакетов (рис. 1).

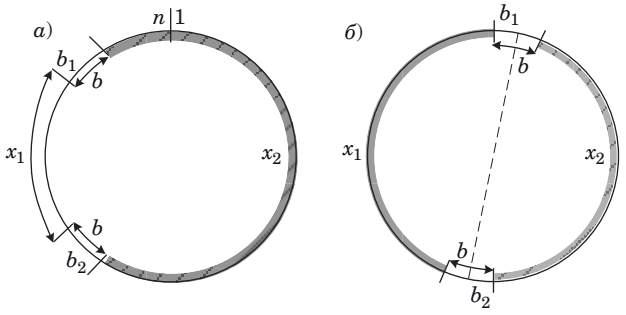
Пусть b_1, b_2 — начальные позиции пакетов, при этом $1 \le b_1 < b_2 \le n - b + 1$, где $b = \lfloor (n-k)/4 \rfloor$ — длина пакета. Пакеты занимают $2b$ из n позиций кодового слова, разбивая множество остальных позиций на два подмножества (с учетом цикличности) из x_1 и x_2 элементов (рис. 1, *а*), при этом

$$x_1 + x_2 + 2b = n. \qquad (1)$$

Оценим параметры кода, при которых построение плотной информационной совокупности будет невозможно. Пусть пакеты лежат «напротив» друг друга (рис. 1, *б*), или, другими словами, $|x_1 - x_2| \le 1$, но для простоты будем считать, что $x_1 = x_2 = x$. С учетом $b = \lfloor (n-k)/4 \rfloor$ из (1) имеем

$$2x + (n-k)/2 = n. \qquad (2)$$

Плотная информационная совокупность не может быть построена, если $x_1 < k$ и $x_2 < k$, т. е. $x < k$. Тогда из (2) получим $n < 3k$; с учетом, что



■ ***Рис. 1.*** Размеры интервалов для поиска информационных совокупностей при $x_2 \neq x_1$ (*а*) и при $x_2 = x_1$ (*б*)

■ ***Fig. 1.*** Interval sizes for searching information sets for $x_2 \neq x_1$ (*а*) and for $x_2 = x_1$ (*б*)

$R = k/n$, получим $R > 1/3$. Таким образом, использование только плотных информационных совокупностей возможно лишь для низкоскоростных кодов со скоростью $R < 1/3$.

Оценим количество необходимых для декодирования плотных информационных совокупностей при $R < 1/3$. Будем использовать следующую методику. Построим таблицу L из двух столбцов, в каждой строке которой содержатся начальные позиции двух пакетов (b_1, b_2), $b_1 < b_2$.

Строки (b_1, b_2) в таблице L будем располагать в лексикографическом порядке. Например, при $n = 10$, $R = 0{,}2$ имеем $b = 2$. Тогда таблица L будет выглядеть, как на рис. 2, *а*. Такая таблица описывает конфигурации пакетов, графически представленных на рис. 2, *б*, где точки соответствуют позициям пакетов. Количество строк N_L в таблице L не более $C_n^2 = O\left(n^2\right)$. Будем говорить, что пакеты покрываются некоторой информационной совокупностью, если позиции информационной совокупности не пересекаются с позициями пакетов, т. е. интервалами $[b_1, b_1 + b - 1]$ и $[b_2, b_2 + b - 1]$.

Задача построения множества X состоит в том, чтобы информационные совокупности из X покрывали всю таблицу L и множество X имело как можно меньшее число элементов $N = |\mathrm{X}|$. Как было показано ранее, множество из n всевозможных плотных информационных совокупностей покроет все строки таблицы L. Оценим возможность уменьшения этого количества.

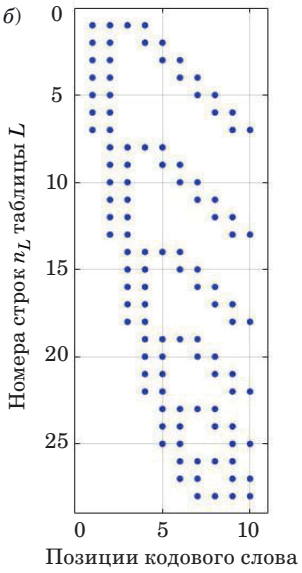
Рассмотрим $n = 50$ и разные значения скоростей R. Множество всех плотных информационных совокупностей содержит $N = 50$ элементов. Оценим экспериментально минимальное количество T плотных информационных совокупностей, покрывающих всю таблицу L.

Для однократных пакетов и кодов с низкими скоростями возможна ситуация, когда двух плотных информационных совокупностей достаточно для исправления всех пакетов длиной

а)

$L =$

n_L	b_1	b_2
1	1	3
2	1	4
3	1	5
4	1	6
5	1	7
6	1	8
7	1	9
8	2	4
9	2	5
10	2	6
11	2	7
12	2	8
13	2	9
14	3	5
15	3	6
16	3	7
17	3	8
18	3	9
19	4	6
20	4	7
21	4	8
22	4	9
23	5	7
24	5	8
25	5	9
26	6	8
27	6	9
28	7	9



■ ***Рис. 2.*** Позиции двукратных пакетов ошибок в табличном (*а*) и графическом (*б*) виде

■ ***Fig. 2.*** Positions of double error bursts in table (*а*) and in graphical (*б*) form

до b_0 [17]. Очевидно, для двукратных пакетов нижняя граница количества необходимых плотных информационных совокупностей T равна трем, так как любые две информационные совокупности могут быть затронуты двукратным пакетом. Эксперименты показывают, что для скоростей $R \leq 0{,}2$ эта граница может быть достигнута с равенством, с дальнейшим ростом скорости до 1/3 количество минимально необходимых информационных совокупностей растет, но не превышает n. Для скоростей кода выше 1/3 построение множества, состоящего только из плотных информационных совокупностей, невозможно.

Построение множества информационных совокупностей с равномерным разбиением для исправления двукратных пакетов ошибок

Как было показано в предыдущем разделе, при $R > 0{,}33$ всегда будут существовать такие расположения пакетов (b_1, b_2), для которых невозможно построение плотной информационной совокупности.

Разобьем плотную информационную совокупность на две равные части по $k/2$ позиций (возможностью нечетных значений k пренебрежем). Оценим условия, при которых такую информационную совокупность построить невозможно. Согласно обозначениям рис. 1, это соответствует случаю $x_1 < k/2$, $x_2 < k$ (или $x_1 < k$, $x_2 < k/2$). Тогда $x_1 + x_2 < k/2 + k$. Из (1) получим $k/2 + k + (n - k)/2 > n$, отсюда $R > 1/2$. Таким образом, равномерное разбиение информационных совокупностей для исправления двукратных пакетов ошибок имеет смысл для кодов со скоростями $R \in (1/3, 1/2)$.

Рассмотрим следующий подход к построению множества X информационных совокупностей для скоростей $R \in (1/3, 1/2)$. Для оценки исправляемых пакетов будем использовать таблицу L. При R из указанного диапазона множество, состоящее только из плотных информационных совокупностей, гарантированно оставит непокрытые пакеты. Однако начнем процедуру построения множества X с перебора всех n плотных информационных совокупностей и учета покрываемых ими пакетов.

Будем далее перебирать информационные совокупности, состоящие из двух частей по $k/2$ позиций. Методику перебора опишем следующим образом (рис. 3). Пусть (b_1, b_2) — первая непокрытая строка таблицы L (после перебора по плотным информационным совокупностям). Выберем информационную совокупность, состоящую из двух равных частей, начинающихся в позициях $((b_1 - 1) - k/2) \bmod n + 1$ и $b_2 - k/2$ с учетом $b_1 < b_2$, возможной цикличности позиций информационной совокупности и используемой нумерации позиций от 1 до n, и отметим, какие двукратные пакеты (строки таблицы L) покрываются новой информационной совокупностью. Будем повторять описанную процедуру до тех пор, пока все строки L не окажутся покрытыми.

Оценим экспериментально описанную методику для кода с параметрами $n = 50$, $R = 0{,}4$. Эксперименты показывают, что 97 % расположений двукратных пакетов покрываются плотными информационными совокупностями (их число $N = 50$). Для того чтобы покрыть оставшиеся расположения пакетов, достаточно три дополнительные информационные совокупности, состоящие из равных частей. Общее число информационных совокупностей для кода с такими параметрами $N_{all} = 53$.

Рассмотрим еще один пример со скоростью $R = 0{,}4$ и $n = 80$. Плотными информационными совокупностями покрывается 88 % расположений двукратных пакетов ($N = 80$), для полного покрытия требуется четыре дополнительные информационные совокупности из двух равных частей, соответственно, $N_{all} = 84$.

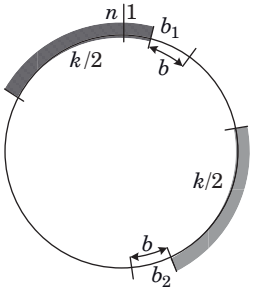
Для $n = 80$ и $R = 0{,}5$ плотными информационными совокупностями покрывается 69 % расположений пакетов, а для полного покрытия требуется 20 дополнительных информационных совокупностей, $N_{all} = 100$.

Таким образом, эксперименты показывают, что для кодов со скоростью $R \in (1/3, 1/2)$ количество информационных совокупностей, необходимых для исправления двукратных пакетов в пределах границы Рейгера, лишь незначительно превосходит длину кода.

Построение множества информационных совокупностей с неравномерным разбиением для исправления двукратных пакетов ошибок

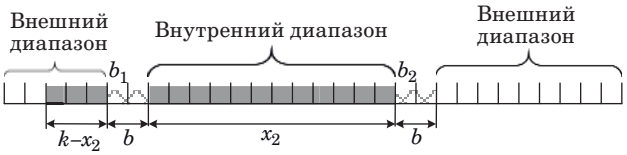
Для кодов со скоростями, превышающими 0,5, описанные ранее методики построения множества информационных совокупностей не позволят исправлять все двукратные пакеты ошибок в рамках модифицированной границы Рейгера. Используем подход с неравномерным разбиением плотных информационных совокупностей (рис. 4).

Как и ранее, сначала будем строить плотные информационные совокупности. После того как в таблице L отмечены все покрываемые ими пакеты, рассмотрим первый непокрытый двукратный пакет (b_1, b_2).



■ ***Рис. 3.*** Выбор расположения двух частей информационной совокупности

■ ***Fig. 3.*** Selection the location of the halves of the information set



■ ***Рис. 4.*** Выбор расположений неравных долей информационных совокупностей

■ ***Fig. 4.*** Selection the locations for unequal shares of information sets

Возьмем в информационную совокупность все позиции «внутреннего» диапазона, т. е. лежащие между пакетами в интервале $[b_1 + b, b_2 - 1]$, оставшиеся $k - b_2 + b_1 + b$ позиций («внешний» диапазон) возьмем, при необходимости циклически, слева от первого пакета, т. е. начиная с позиций $(b_2 - b - k - 1) \bmod n + 1$ (см. рис. 4).

Оценим применение этой методики для кодов с $R \in [0{,}6; 0{,}9]$, число плотных информационных совокупностей для всех случаев $N = n$ (табл. 1). Приведена доля двукратных пакетов, покрываемых первоначальным множеством X из $N = n$ плотных информационных совокупностей, в процентах. Параметр N_1 показывает количество информационных совокупностей, которые потребовалось добавить к множеству X, состоящему из плотных совокупностей, чтобы покрыть все двукратные пакеты. Наконец, параметр N_{all} показывает общую итоговую мощность множества X.

Как можно видеть, с увеличением скорости кода увеличивается количество информационных совокупностей, требуемых для покрытия всех возможных расположений двукратных пакетов ошибок. Тем не менее общее количество информационных совокупностей N_{all} растет пропорционально длине кода, хотя и с повышающим коэффициентом, что отличается от ситуации с исправлением независимых ошибок, при которых число информационных совокупностей экспоненциально.

Указанная методика, использующая вначале внутренний диапазон позиций между пакетами, не учитывает то, что фактически при циклическом выборе позиций в информационные совокупности позиции слева от первого пакета и справа от второго, т. е. внешний диапазон, также могут рассматриваться как интервал между пакетами. Таким образом, размеры частей, на которые разбивается изначально плотная информационная совокупность, варьируются в зависимости от размера внутреннего диапазона, который изменяется в процессе построения непредсказуемо. Воспользуемся эвристикой, в соответствии с которой из двух диапазонов между пакетами, внутренним и внешним, вначале будет выбираться больший. Это обусловлено тем, чтобы полученные части были по возможности более неравномерны и, таким образом, более близки к плотной информационной совокупности, что позволяет надеяться покрыть больше расположений пакетов меньшим числом совокупностей.

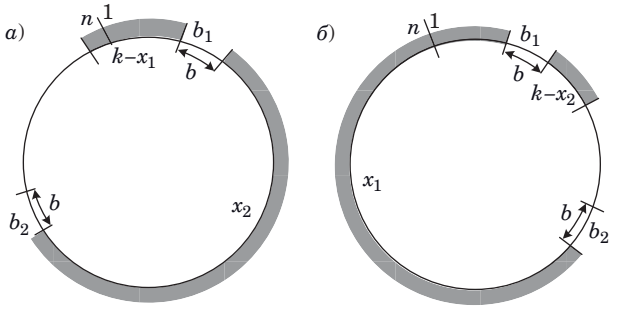
Основываясь на этих рассуждениях, приведем методику построения множества информационных совокупностей для исправления двукратных пакетов ошибок, в некотором смысле обобщающую предыдущие. Для заданных начальных позиций пакетов (b_1, b_2) определим число циклических позиций между ними (см. рис. 1, *а*). Получим $x_1 = n + b_1 - b_2 - b$ и $x_2 = b_2 - b_1 - b$ позиций, используя обозначения из рис. 1, *а*. Пусть $x_1 \le x_2$, тогда выберем x_2 позиций между пакетами в информационную совокупность, оставшиеся $k - x_2$ позиций, как и ранее, выберем слева (при необходимости — циклически) от пакета с началом b_1 (рис. 5, *а*). При $x_1 > x_2$ выполним зеркальные действия (рис. 5, *б*).

Заметим, что при $x_1 > k$ или $x_2 > k$ вследствие использования такой методики будут получены

■ ***Таблица 1.*** Количество информационных совокупностей при использовании методики с неравномерным разбиением

■ ***Table 1.*** Number of information sets when using the non-uniform partitioning method

Параметры кода		Доля расположений, покрытых плотными совокупностями, %	Количество дополнительных информационных совокупностей N_1	Количество всех информационных совокупностей N_{all}
Длина n	Скорость R			
50	0,6	54	49	99
	0,7	44	75	125
	0,8	30	130	180
	0,9	16	275	325
80	0,6	52	79	159
	0,7	38	146	226
	0,8	25	259	339
	0,9	13	553	633
100	0,6	52	99	199
	0,7	40	178	278
	0,8	24	329	429
	0,9	14	618	718



■ ***Рис. 5.*** Динамический выбор расположений неравных долей информационной совокупности при $x_1 \le x_2$ (*а*) и при $x_1 > x_2$ (*б*)

■ ***Fig. 5.*** Dynamic selection the locations for unequal shares of the information set for $x_1 \le x_2$ (*а*) and for $x_1 > x_2$ (*б*)

плотные информационные совокупности. Это позволяет исключить как отдельный этап построение первоначального множества плотных информационных совокупностей. Результаты использования такой методики приведены в табл. 2.

Как видно из таблицы, количество информационных совокупностей, построенных по данной методике, меньше, чем в случае первоначального выбора внутреннего диапазона (см. табл. 1). Кроме того, особенностью данной методики является то, что ее без изменений возможно использовать и для низкоскоростных кодов, так как при ее использовании могут быть построены плотные информационные совокупности. Так, число информационных совокупностей для $R \le 0{,}2$ соответствует нижней границе количества необходимых плотных информационных совокупностей T, которые были получены ранее. Заметим также, что полученные результаты для $n = 80$, $R = 0{,}4$ и 0,5 лучше, чем при использовании описанного в предыдущем разделе равномерного разбиения (общее число информационных совокупностей составляет 63 и 79 против 84 и 100 соответственно).

Напомним (см. рис. 2), что позиции начала пакетов при их перечислении в таблице L во всех случаях выбирались в лексикографическом порядке. Оценим влияние порядка построения таблицы L на размер множества X для последней методики. Для сравнения будем выбирать строки таблицы L в случайном порядке, результаты приведены в табл. 2.

Как видно из таблицы, для случайного порядка размер множества информационных совокупностей заметно увеличивается. Вероятно, это связано с тем, что при лексикографическом порядке в первую очередь генерируются информационные совокупности, которые покрывают большее количество расположений. При таком порядке сначала строятся плотные информационные совокупности, так как расположения пакетов находятся близко друг к другу. При случайном порядке появляется тенденция построения информационных совокупностей с более равномерным разбиением на части, что приводит к уменьшению количества покрываемых пакетов отдельной информационной совокупностью. Таким образом, влияние порядка просмотра расположений пакетов приводит к новой оптимизационной задаче поиска оптимального порядка для минимизации числа информационных совокупностей, что может являться направлением дальнейшей работы.

Заключение

В статье решается задача построения множества информационных совокупностей для исправления двукратных пакетов ошибок. Исследовано несколько методик для построения такого множества, получены границы применимости с точки зрения кодовых скоростей.

Показано, что известный подход на основе плотных информационных совокупностей в случае исправления двукратных пакетов применим только для кодов со скоростью $R < 1/3$, для кодов с более высокими скоростями рассмотрены и проанализированы методики построения множества информационных совокупностей с равномерным (для $R \in (1/3, 1/2)$) и неравномерным (для $R > 1/2$) разбиением. Экспериментальная оценка для конкретных значений параметров (длин и скоростей) кодов показала, что размеры полученных множеств для кодов со скоростями выше 1/2 пропорциональны длине кода и позволяют исправлять любые расположения двукратного пакета ошибок.

■ ***Таблица 2.*** Количество информационных совокупностей при использовании динамической методики с неравномерным разбиением при разном порядке строк в таблице L

■ ***Table 2.*** Number of information sets when using the dynamic non-uniform partitioning method with different row order in table L

Длина n	Общее количество N_{all} при скорости R								
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
Лексикографический порядок									
50	3	3	15	41	48	80	102	158	312
80	3	3	27	63	79	128	198	313	593
100	3	3	33	79	97	160	205	392	697
Случайный порядок									
50	5	6	25	78	116	148	193	276	461
80	6	6	43	140	196	276	396	560	976
100	7	12	49	208	275	340	509	776	1216

Предложена общая методика построения множества информационных совокупностей для кодов любых скоростей. С помощью данной методики возможно сократить размер множества информационных совокупностей по сравнению с предыдущими методиками.

Необходимо отметить, что при используемых в работе предположениях об исправляемых длинах пакетов на основе модифицированной границы Рейгера построенные множества информационных совокупностей позволяют исправлять как двукратные, так и однократные пакеты ошибок двойной длины.

Предложенная методика может быть использована для разработки вычислительно эффективных декодеров однократных и двукратных пакетов ошибок для конкретных кодовых конструкций.

Финансовая поддержка

Работа выполнена при финансовой поддержке Российского научного фонда, грант № 22-19-00305 «Пространственно-временные стохастические модели беспроводных сетей с большим числом абонентов».

Литература

1. **Moon T. K.** *Error Correction Coding: Mathematical Methods and Algorithms*. Second edition. Hoboken, NJ, Wiley, 2021. 992 p.
2. **Smeshko A., Ivanov F., Zyablov V.** Theoretical estimates of burst error probability for convolutional codes. *2020 International Symposium on Information Theory and Its Applications (ISITA)*, Kapolei, HI, USA, 2020, pp. 136–140.
3. **Kuvshinov A., Timokhin I., Ivanov F.** On the concatenation of superposition and polar codes. *2024 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, Novosibirsk, Russian Federation, 2024, pp. 52–57. doi:10.1109/SIBIRCON63777.2024.10758448
4. **Liu H., Xiao L., Wang T., Li J.** Error detection and correction codes for memories with enhanced detection abilities. *2024 IEEE 7th International Conference on Electronics and Communication Engineering (ICECE)*, Xi'an, China, 2024, pp. 165–168. doi:10.1109/ICECE63871.2024.10976832
5. **Kim C., Kim J.-W., No J.-S.** New design of error control codes resilient to single burst error or two random bit errors using constacyclic codes. *IEEE Access*, 2022, vol. 10, pp. 131101–131108. doi:10.1109/ACCESS.2022.3229427
6. **Liu H., Xiao L., Wang T., Li J., Li J.** Error correction codes for double burst errors correction in memories. *IEEE Access*, 2025, vol. 13, pp. 116621–116631. doi:10.1109/ACCESS.2025.3586226
7. **Simegn D., Andreev K., Rybin P., Frolov A.** On the design of LDPC-based error-reducing codes. *2024 19th International Symposium on Wireless Communication Systems (ISWCS)*, Rio de Janeiro, Brazil, 2024, pp. 1–6. doi:10.1109/ISWCS61526.2024.10639052
8. **Li L., Lv J., Li Y., Dai X., Wang X.** Burst error identification method for LDPC coded systems. *IEEE Communications Letters*, 2024, vol. 28, no. 7, pp. 1489–1493. https://doi.org/10.1109/LCOMM.2024.3391826
9. **Ovchinnikov A. A.**, **Veresova A. M.**, **Fominykh A. A.** Decoding of linear codes for single error bursts correction based on the determination of certain events, *Информационно-управляющие системы*, 2022, no. 6, pp. 41–52. doi:10.31799/1684-8853-2022-6-41-52, EDN: UWXZHN
10. **Karakchieva L., Trifonov P.** A recursive soft-input soft-output decoding algorithm. *IEEE Transactions on Communications*, 2024, vol. 72, no. 3, pp. 1290–1302. doi:10.1109/TCOMM.2023.3334812
11. **Aharoni Z., Huleihel B., Pfister H. D., Permuter H. H.** Data-driven polar codes for unknown channels with and without memory. *2023 IEEE International Symposium on Information Theory (ISIT)*, Taipei, IEEE, 2023, pp. 1890–1895. doi:10.1109/ISIT54713.2023.10206663
12. **Fang Y., Chen J.** Decoding polar codes for a generalized Gilbert — Elliott channel with unknown parameter. *IEEE Transactions on Communications*, 2021, vol. 69, no. 10, pp. 6455–6468. https://doi. org/10.1109/TCOMM.2021.3095195
13. **Yang M., Pan Z., Djordjevic I. B.** FPGA-based burst-error performance analysis and optimization of regular and irregular SD-LDPC codes for 50G-PON and beyond. *Opt. Express*, 2023, vol. 31, no. 6, pp. 10936–10946. doi:10.1364/OE.477546
14. **Song L., Huang Q., Wang Z.** Construction of multipleburst-correction codes in transform domain and its relation to LDPC codes. *IEEE Trans. Commun.*, 2020, vol. 68, no. 1, pp. 40–54. doi:10.1109/TCOMM.2019.2948341
15. **Xiao X., Vasic B., Lin S., Li J., Abdel-Ghaffar K.** Quasi-cyclic LDPC codes with parity-check matrices of column weight two or more for correcting phased bursts of erasures. *IEEE Transactions on Communications*, 2021, vol. 69, no. 5, pp. 2812–2823. doi:10.1109/TCOMM. 2021.3059001
16. **Вересова А. М.** Оценка эффективности использования марковской метрики при декодировании в каналах с памятью. *Информационно-управляющие системы*, 2025, № 1, с. 29–41. doi:10.31799/1684-8853-2025-1-29-41, EDN: JQIBMZ
17. **Трофимов А. Н.**, **Таубин Ф. А.** Улучшенная граница вероятности ошибки при оптимальном приеме в канале с межсимвольной интерференцией. *Ин-*

формационно-управляющие системы, 2023, № 5, с. 33–42. doi:10.31799/1684-8853-2023-5-33-42, EDN: MDHOXU

18. **Kandhway K.** Modeling burst errors in a fading channel. *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)*, Indore, India, 2022, pp. 409–414. doi:10.1109/CSNT54456.2022.9787652
19. **Исаева М. Н.** Декодирование одиночных пакетов ошибок по минимуму длины пакета на основе информационных совокупностей. *Информационно-управляющие системы*, 2025, № 2, с. 68–77. doi:10.31799/1684-8853-2025-2-68-77, EDN: MAHCAY
20. **Lin S., Li J.** *Fundamentals of Classical and Modern Error-Correcting Codes*. Cambridge, Cambridge University Press, 2022. 840 p. doi:10.1017/9781009067928
21. **Veresova A. M.**, **Isaeva M. N.**, **Ovchinnikov A. A.** Estimation of independent errors and bursts correction capability of linear codes. *2024 Conference of Young Researchers in Electrical and Electronic Engineering (ElCon)*, Saint Petersburg, IEEE, 2024, pp. 23–27. doi:10.1109/ElCon61730.2024.10468456
22. **Исаева М. Н.** Разработка и анализ методики построения множества плотных информационных совокупностей для исправления пакетов ошибок. *T-Comm: Телекоммуникации и транспорт*, 2024, т. 18, № 10, с. 20–26. doi:10.36724/2072-8735-2024-18-10-20-26

UDC 519.72
doi:10.31799/1684-8853-2025-6-64-73
EDN: ESMBYH

Methodology for constructing information sets with non-uniform partitioning for error burst correction

M. N. Isaeva[a], Senior Lecturer, orcid.org/0009-0007-6228-0617, imn@guap.ru
[a]Saint-Petersburg State University of Aerospace Instrumentation, 67, B. Morskaia St., 190000, Saint-Petersburg, Russian Federation

Introduction: Error correction is a relevant problem for modern communication and data storage systems, especially in channels characterized by error bursts. The task of correcting more than one error burst occurring during the transmission of a data block remains poorly studied. **Purpose:** To develop and analyze methods for generating a set of information sets for correcting double error bursts. **Results:** The study has shown that dense information sets previously studied for correcting single error bursts are applicable for correcting double error bursts only for low-rate codes. For codes with higher rates, methods for constructing a set of information sets of a more general type are proposed and analyzed: with uniform and dynamic non-uniform partitioning. The proposed methodology allows correcting any combinations of double error bursts whose length is within the modified Reiger bound. **Practical relevance:** The results of this work are of practical importance for the design of communication systems for transmission over channels with memory, where the frequency of bursts occurrence is high enough to form multiple bursts during the transmission of a single code word. The proposed methods allow improving the noise resistance of such channels and can be used to develop computationally efficient decoders. **Discussion:** The results have been obtained assuming that any consecutive positions of the code word form an information set, and that the lengths of the corrected bursts lie on the modified Reiger bound. It can be expected that these two effects will compensate each other to some extent in terms of the requirements for sets of information sets for specific codes, but the estimation of the parameters of sets of information sets and the construction of decoders based on them for individual classes of codes is a direction for further research.

Keywords — information set decoding, Reiger bound, correction of multiple error bursts, dense information sets, channels with memory.

For citation: Isaeva M. N. Methodology for constructing information sets with non-uniform partitioning for error burst correction. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 64–73 (In Russian). doi:10.31799/1684-8853-2025-6-64-73, EDN: ESMBYH

Financial support
The paper was prepared with the financial support of the Russian Science Foundation, project No. 22-19-00305 “Spatial-temporal stochastic models of wireless networks with a large number of users”.

References

1. Moon T. K. *Error Correction Coding: Mathematical Methods and Algorithms*. Second edition. Hoboken, NJ, Wiley, 2021. 992 p.
2. Smeshko A., Ivanov F., Zyablov V. Theoretical estimates of burst error probability for convolutional codes. *2020 International Symposium on Information Theory and Its Applications (ISITA)*, Kapolei, HI, USA, 2020, pp. 136–140.
3. Kuvshinov A., Timokhin I., Ivanov F. On the concatenation of superposition and polar codes. *2024 IEEE International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, Novosibirsk, Russian Federation, 2024, pp. 52–57. doi:10.1109/SIBIRCON63777.2024.10758448
4. Liu H., Xiao L., Wang T., Li J. Error detection and correction codes for memories with enhanced detection abilities. *2024 IEEE 7th International Conference on Electronics and Communication Engineering (ICECE)*, Xi’an, China, 2024, pp. 165–168. doi:10.1109/ICECE63871.2024.10976832
5. Kim C., Kim J.-W., No J.-S. New design of error control codes resilient to single burst error or two random bit errors using constacyclic codes. *IEEE Access*, 2022, vol. 10, pp. 131101–131108. doi:10.1109/ACCESS.2022.3229427
6. Liu H., Xiao L., Wang T., Li J., Li J. Error correction codes for double burst errors correction in memories. *IEEE Access*, 2025, vol. 13, pp. 116621–116631. doi:10.1109/ACCESS.2025.3586226
7. Simegn D., Andreev K., Rybin P., Frolov A. On the design of LDPC-based error-reducing codes. *2024 19th International Symposium on Wireless Communication Systems (ISWCS)*, Rio de Janeiro, Brazil, 2024, pp. 1–6. doi:10.1109/ISWCS61526.2024.10639052

8. Li L., Lv J., Li Y., Dai X., Wang X. Burst error identification method for LDPC coded systems. *IEEE Communications Letters*, 2024, vol. 28, no. 7, pp. 1489–1493. https://doi.org/10.1109/ LCOMM.2024.3391826
9. Ovchinnikov A. A., Veresova A. M., Fominykh A. A. Decoding of linear codes for single error bursts correction based on the determination of certain events, *Информационно-управляющие системы*, 2022, no. 6, pp. 41–52. doi:10.31799/1684-8853-2022-6-41-52, EDN: UWXZHN
10. Karakchieva L., Trifonov P. A recursive soft-input soft-output decoding algorithm. *IEEE Transactions on Communications*, 2024, vol. 72, no. 3, pp. 1290–1302. doi:10.1109/TCOMM.2023.3334812
11. Aharoni Z., Huleihel B., Pfister H. D., Permuter H. H. Data-driven polar codes for unknown channels with and without memory. *2023 IEEE International Symposium on Information Theory (ISIT)*, Taipei, IEEE, 2023, pp. 1890–1895. doi:10.1109/ISIT54713.2023.10206663
12. Fang Y., Chen J. Decoding polar codes for a generalized Gilbert – Elliott channel with unknown parameter. *IEEE Transactions on Communications*, 2021, vol. 69, no. 10, pp. 6455–6468. https://doi. org/10.1109/TCOMM.2021.3095195
13. Yang M., Pan Z., Djordjevic I. B. FPGA-based burst-error performance analysis and optimization of regular and irregular SD-LDPC codes for 50G-PON and beyond. *Opt. Express*, 2023, vol. 31, no. 6, pp. 10936–10946. doi:10.1364/OE.477546
14. Song L., Huang Q., Wang Z. Construction of multiple-burst-correction codes in transform domain and its relation to LDPC codes. *IEEE Trans. Commun.*, 2020, vol. 68, no. 1, pp. 40–54. doi:10.1109/TCOMM.2019.2948341
15. Xiao X., Vasic B., Lin S., Li J., Abdel-Ghaffar K. Quasi-cyclic LDPC codes with parity-check matrices of column weight two or more for correcting phased bursts of erasures. *IEEE Transactions on Communications*, 2021, vol. 69, no. 5, pp. 2812–2823. doi:10.1109/TCOMM. 2021.3059001
16. Veresova A. M. Performance evaluation of decoding in channels with memory with the use of a Markov metric. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 1, pp. 29–41 (In Russian). doi:10.31799/1684- 8853-2025-1-29-41, EDN: JQIBMZ
17. Trofimov A. N., Taubin F. A. Improved bound on optimal reception error probability for an intersymbol interference channel. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2023, no. 5, pp. 33–42 (In Russian). doi:10.31799/1684- 8853-2023-5-33-42, EDN: MDHOXU
18. Kandhway K. Modeling burst errors in a fading channel. *2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT)*, Indore, India, 2022, pp. 409–414. doi:10.1109/CSNT54456.2022.9787652
19. Isaeva M. N. Decoding of single error bursts using minimal burst length criteria and information sets. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 2, pp. 68–77 (In Russian). doi:10.31799/1684-8853-2025-2-68-77, EDN: MAHCAY
20. Lin S., Li J. *Fundamentals of Classical and Modern Error-Correcting Codes*. Cambridge, Cambridge University Press, 2022. 840 p. doi:10.1017/9781009067928
21. Veresova A. M., Isaeva M. N., Ovchinnikov A. A. Estimation of independent errors and bursts correction capability of linear codes. *2024 Conference of Young Researchers in Electrical and Electronic Engineering (ElCon)*, Saint Petersburg, IEEE, 2024, pp. 23–27. doi:10.1109/ElCon61730.2024.10468456
22. Isaeva M. N. Development and analysis of a method for constructing dense information sets for error bursts correction. *T-Comm*, 2024, vol. 18, no. 10, pp. 20–26 (In Russian). doi:10.36724/2072-8735-2024-18-10-20-26

ПАМЯТКА ДЛЯ АВТОРОВ

Поступающие в редакцию статьи проходят обязательное рецензирование.

При наличии положительной рецензии статья рассматривается редакционной коллегией. Принятая в печать статья направляется автору для согласования редакторских правок. После согласования автор представляет в редакцию окончательный вариант текста статьи.

Процедуры согласования текста статьи могут осуществляться как непосредственно в редакции, так и по e-mail (ius.spb@gmail.com).

При отклонении статьи редакция представляет автору мотивированное заключение и рецензию, при необходимости доработать статью — рецензию.

Редакция журнала напоминает, что ответственность за достоверность и точность рекламных материалов несут рекламодатели.

UDC 004.728.3.057.4
doi:10.31799/1684-8853-2025-6-74-84
EDN: EHEWUI

Articles

Comparative analysis of ALOHA based algorithms with early feedback

A. A. Burkov[a], *PhD, Tech., orcid.org/0000-0002-0920-585X*
R. O. Rachugin[a], *Post-Graduate Student, orcid.org/0000-0001-5813-3867*
A. M. Turlikov[a], *Dr. Sc., Tech., Professor, orcid.org/0000-0001-7132-094X, turlikov@guap.ru*
[a]*Saint-Petersburg State University of Aerospace Instrumentation, 67, B. Morskaia St., 190000, Saint-Petersburg, Russian Federation*

Introduction: *One option for increasing the throughput of a random multiple access system is to use early feedback. Early feedback refers to a rapid response from the base station after receiving the preambles. In this paper, early feedback is considered for multiple access systems using different ALOHA-based algorithms.* ***Purpose:*** *To conduct a comparative analysis of the dependence of the maximum throughput on the number of unique preambles of random access algorithms based on the ALOHA algorithm with early feedback.* ***Results:*** *The paper considers the ALOHA algorithm with an exploration phase, the 2-step ALOHA algorithm and a combination of these two algorithms. For these three algorithms we carry out a comparative analysis for the variant with a known number of unique users and the variant with an estimation of the number of active users. The study shows that the use of the procedure for estimating the number of active users allows achieving similar values of the dependence of the maximum throughput on the number of unique preambles as for the first variant with a known number of active users. In addition, it is shown that the use of a fixed parameter affecting the estimation procedure, with the value of this parameter equal to its optimal for infinite numbers of preambles, leads to a loss that does not exceed 6% for any numbers of preambles, and does not exceed 0.2% for 30 or more preambles.* ***Practical relevance:*** *New algorithm based on ALOHA with early feedback is proposed, which allows increasing the maximum throughput of the system as compared to previously known algorithms. This algorithm can be used in the random access channel of future generation networks.* ***Discussion:*** *The analysis does not take into account the influence of the number of channels used in the system, which could be a further direction of research.*

Keywords *— ALOHA, grant-free random access, preamble-based exploration, estimation, throughput, ergodicity, Markov chain.*

For citation: Burkov A. A., Rachugin R. O., Turlikov A. M. Comparative analysis of ALOHA based algorithms with early feedback. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 74–84. doi:10.31799/1684-8853-2025-6-74-84, EDN: EHEWUI

Introduction

The random access (RA) procedure in 4G (LTE) networks is implemented using a 4-step scheme [1, 2]. Its mechanism is as follows: first, the user selects and sends a unique preamble. The base station then attempts to decode the preambles and sends a response allocating resources for data transmission. Then the user uses this resource to send his data. A collision occurs when multiple users select the same preamble. At the final stage, the base station informs all users about an event in the channel after frame of RA procedure via feedback channel. The distribution of these resources is called grants, and the RA principle itself is called the grant-based principle [3].The main disadvantage of this scheme becomes apparent when a lot of devices are connected: the number of collisions increases sharply and, as a consequence, the time it takes to resolve them increases, which causes significant delays.

To solve this problem, the transmission decision in the first stage is applied randomly by each user, which helps to reduce the number of collisions, this mechanism is called access class barring (ACB) [4, 5]. The essence of this mechanism is to apply a probabilistic approach to the transmission of the preamble at the first stage of the RA procedure. By allowing users to transmit the preamble not in every iteration, but only with a given probability, the total number of collisions and average delay are reduced. To minimize latency in the ACB method, the transmission probability should be related to the number of users attempting to gain access [6]. With precise knowledge of this quantity allows you to select the optimal probability. However, in practice, determining the exact number of users is a non-trivial task.

The development of modern wireless communication networks is actively linked to the development of the Internet of Things (IoT) technology. Within this technology, one of the key scenarios is massive Machine-Type Communication (mMTC), which is characterized by the operation of a large number of user devices, to which different requirements apply, which affects the operation of networks. The use of the 4-step scheme in the conditions of mMTC is characterized by two main disadvantages: a high

ratio of service traffic to payload and an increase in transmission time due to delayed collision detection. In 5G NR, an alternative 2-step procedure has been introduced [7, 8], the key feature of which is the sending of a single combined message (preamble and user data) in the first step. This reduces the number of signaling messages and the required base station computing resources. The 2-step RA scheme is classified as a grant-free random access procedure.

The random access schemes described begin with the selection and transmission of one of the unique preambles to users. Typically, preambles are generated based on the Zadoff – Chu sequence. The properties of these sequences are well studied, and their detailed description is discussed in [9, 10].

In [11] and [12], the efficiency improvement of the grant-free RA scheme by means of early feedback is described and investigated. Early feedback can be achieved by early detection of preamble collisions [13]. Thus, in the paper [12] an algorithm based on the multi-channel ALOHA algorithm is discussed, which uses two phases: the exploration phase (EP), during which preambles are transmitted, and the data transmission phase (DTP), during which user messages are transmitted. The analysis of this algorithm is carried out under the assumption that there are no retransmissions, the number of unique preambles is unlimited, and the number of independent channels tends to infinity. In addition, it is shown that the maximum throughput per channel can reach $e^{-1}\left(2-e^{-1}\right)$.

The paper [14] presents an analysis of the same system and investigates the dependence of the maximum channel throughput on the number of unique preambles. A system with retransmission is also considered and an approach to stabilizing this algorithm with a limited number of unique preambles and a single channel is described. Also, in [14] a system is analyzed that uses a specific approach to estimate the number of active users. In the paper [15] a system with one channel and repeated transmissions is considered. For such a system, a 2-step ALOHA algorithm using an early feedback approach is analyzed. In addition, for this algorithm shows the dependence of throughput on the number of unique preambles.

Increasing the throughput for the random multiple access procedure and the use of switching with duplicate routes [16, 17] together can lead to an increase in the probability of message delivery in the system.

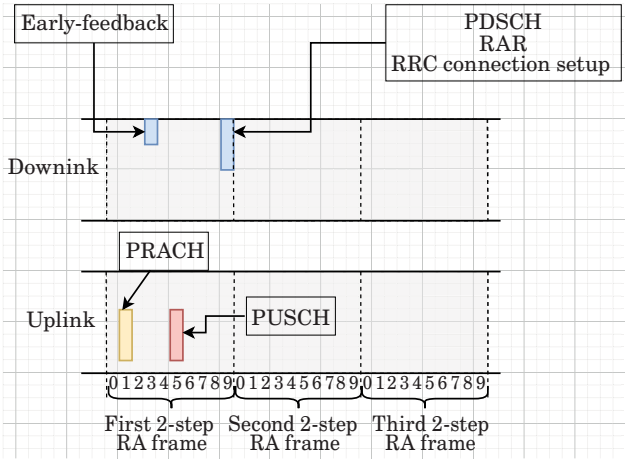
This paper describes a combined random multiple access algorithm that allows increasing the throughput compared to [12], but with a smaller number of channels in the system. Taking into account the use of an estimation procedure similar to that in [14], it is shown that the functioning of the system is described by a two-dimensional Markov chain. The use of this approach allows us to eliminate the necessary condition of an exact number of active users at the beginning of each frame of the RA procedure. In addition, a comparison is made of the dependence of the throughput on the number of unique preambles and the parameter of the procedure for estimating the number of active users for three different algorithms.

Description of the early feedback approach

The idea of early feedback

As was written earlier in the work [12], an approach called early feedback is defined. This approach implies a quick response from the base station after receiving preambles from all users and before users begin transmitting their data. Thus, it is important to note that the frame duration of the RA procedure remains unchanged. Let us consider the application of this approach to a 2-step RA procedure. In this case, RA frame consists of transmitting two messages: message A and message B. Message A represents the transmission of the preamble and the user data (yellow and red blocks shown in Fig. 1, respectively). Message B is a combined response for the received preamble and data (the second blue block shown in Fig. 1). An example of a time-frequency diagram for this approach is shown in Fig. 1.

In addition, in the work [12] the author considers the application of the early feedback approach to the multi-channel RA algorithm ALOHA with an infinite number of channels and an infinite number of preambles.



■ ***Fig. 1.*** Time-frequency diagram of a 2-step RA procedure with early feedback

Using early feedback for the ALOHA-based algorithms

In [14], a modification of the algorithm from [12] is considered for a system with one channel, a limited number of unique preambles and repeated transmissions (users leave the system only after successful transmission of their data). The modified algorithm's frame consists of two phases: an EP and a DTP. During the exploration phase, users transmit preambles, and during the data transmission phase, users transmit their's data.

In [15], the authors consider the application of the early feedback approach to the 2-step ALOHA algorithm. This algorithm is considered for a system with 1 channel, a limited number of preambles and the presence of repeated transmissions. It is important to note that in this case the base station does not need to determine the exact number of preambles received, so simpler preambles can be used.

In this paper, we will consider a combined RA algorithm based on ALOHA with a exploration phase. This combination consists of using a EP similar to the algorithm and the DTP of the algorithm from [14], as well as changing the operation of the RA procedure in the case of the "Empty" event after the EP similar to the algorithm from [15].

The operation of system for these algorithms presented in the Table 1.

In the Table 1 we introduce G — an algorithm parameter that affects the transmission probability during the EP; M_t — the exact number of active users at the beginning of the RA algorithm frame; U — the number of preambles received an EP.

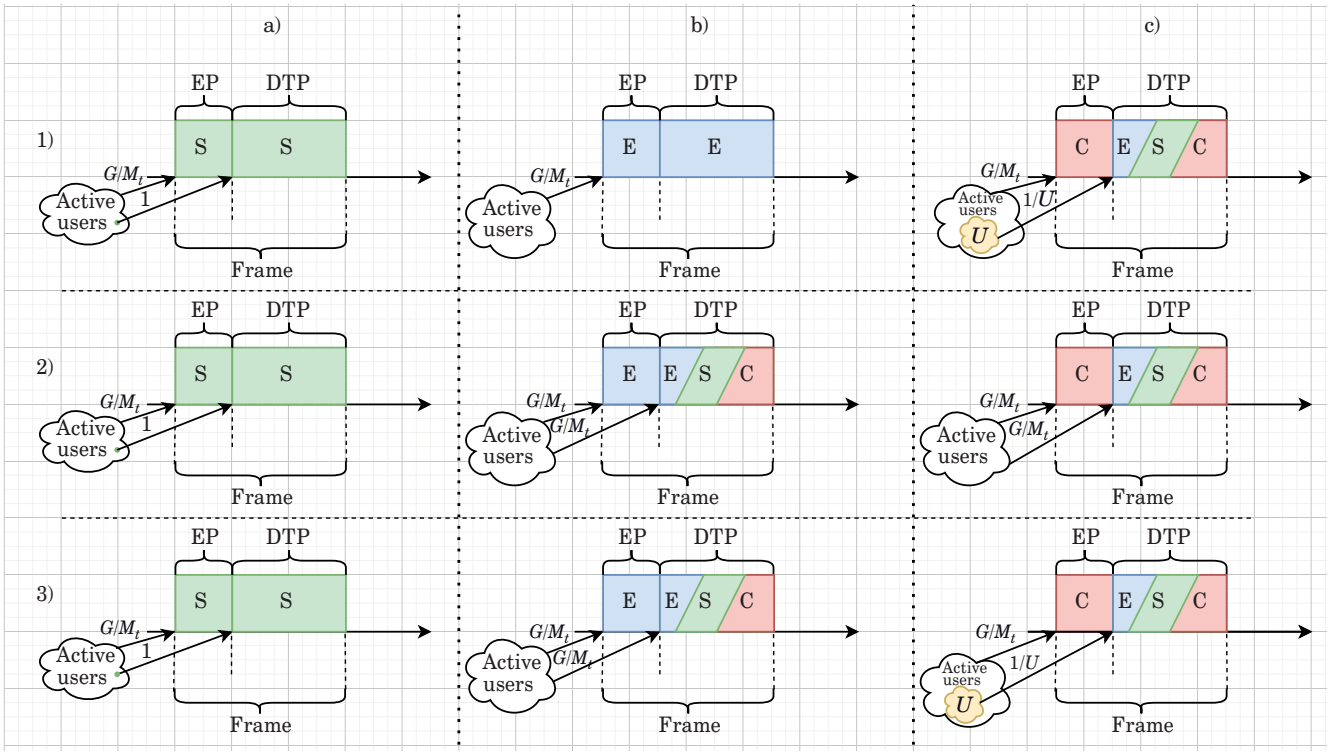
Steps 1 and 2 defined an EP and steps 3 and 4 define DTP.

Possible situations when using the 3 described algorithms are illustrated in Fig. 2: 1) refers to the ALOHA algorithm with a EP, 2) refers to the 2-step ALOHA algorithm, and 3) refers to the algorithm considered in this paper. In addition, a) means the "Success" event in the EP, b) means the "Empty" event in the EP, and c) means the "Conflict" event in the EP.

According to the possible events presented in Fig. 2, if the "Success" event occurs on the EP, the

■ ***Table 1.*** Algorithm of operation the system with ALOHA-based algorithms with early feedback

Number of step	ALOHA with EP	2-step ALOHA	Combined ALOHA
Step 1	With probability p_{EP} each user decides whether to transmit the preamble and with probability $1 - p_{EP}$ decides not to transmit p_{EP} can be calculated using the formula: $P_{EP} = \min\left(1, \frac{G}{M_t}\right)$ Users who decide to transmit randomly select and transmit one of the L preambles		
Step 2	The base station informs users about the number of detected preambles via the feedback channel	Through the feedback channel, the base station informs all users about the event that occurred in the channel (one of the following events is possible: "Success" — 1 preamble received, "Empty" — 0 preambles received, "Conflict" — more than 1 preamble received)	The base station informs users about the number of detected preambles via the feedback channel
Step 3 (if Success on EP)	The user transmits during DTP with probability $p_{DTP} = 1$ for 1 detected preamble in the channel		
Step 3 (if Empty on EP)	Nobody transmit on DTP	The user will attempt to transmit data during the DTP phase with the following probability: $p_{DTP} = \min\left\{1, \frac{G}{M_t}\right\}$	All users with a message ready to transmit during DTP will transmit with probability: $p_{DTP} = \min\left(1, \frac{G}{M_t}\right)$
Step 3 (if Conflict on EP)	The user transmits during DTP with a probability of following probability: $p_{DTP} = \frac{1}{U}$	The user will attempt to transmit data during the DTP phase with the following probability: $p_{DTP} = \min\left\{1, \frac{G}{M_t}\right\}$	The user transmits during DTP with a probability of following probability: $p_{DTP} = \frac{1}{U}$
Step 4	At the end of each frame, the base station notifies all users via the feedback channel about the event that occurred on the channel. Users exit the system only if the data transmission is successful; otherwise, they return to step 1 of the algorithm		



■ ***Fig. 2.*** Frame diagram of RA algorithms for various events after EP

"Success" event will occur for all algorithms on the DTP. According to column "b" of Fig. 2, in the case of the "Empty" event on the EP for the ALOHA algorithm with the EP on the DTP, the "Empty" event will occur. With the same event on the EP for the other two algorithms on the DTP, any event can occur ("Empty", "Success", "Conflict"). If the "Conflict" event occurs on the EP, one of the following events may occur on the DTP in all algorithms: "Empty", "Success", "Conflict". For each of these events, an arrow from the entire set of users, from a specific user, or from a subset of users indicates at which phase the decision to transmit is made, and the inscription above the arrow indicates the probability of making such a decision.

To further analyze the combined ALOHA algorithm, we describe the system of assumptions.

System model

In [14], a system model is introduced that is defined by five assumptions. This model assumes the following: the entire system operation time is divided into frames of equal length, consisting of a exploration phase and data transmission phase. The number of active users at the beginning of each frame is known. The system operates with a fixed number of unique preambles. The time required to transmit each preamble is uniform. It is important to note that information about the number of received preambles is transmitted without errors via the feedback channel. The base station reliably determines the number of received preambles. It is assumed that the number of users entering the system during a frame is a random variable distributed according to the Poisson law with parameter λ. Each user has one message ready to be transmitted and after successful transmission leaves the system. Data transmitted by multiple users during DTP cannot be reliably detected at the base station. Each phase ends with one of three events: "Empty", "Success", "Conflict".

This model is fully characterized by two parameters and the users operation algorithm: L is the number of preambles, λ is the input arrival rate.

This model is based on the model from [12], but has some differences. The differences are presented in Table 2.

Throughput with knowledge of the exact number of active users

For further comparison of the algorithm under consideration in this paper with the algorithms presented in [14] and [15], we obtain a formula for calculating the throughput with exact knowledge of the number of active users. It is important to note that similar formulas for the ALOHA algorithms

■ ***Table 2.*** *Differences in model assumption systems*

Number of difference	System from this work	System discussed in [12]
1	Number of unique preamble is finite	Unlimited number of unique preambles
2	System has one channel	System has large number of independent channels
3	Users transmit the preamble with some probability different from one	Users always transmit the preamble with probability one
4	Users are exits the system only if the transfer is successful. In this case, the system can have retransmissions	Users are exits the system after the first data transmission

with exploration phase and the 2-step ALOHA algorithm were obtained in the papers [14] and [15] regarding.

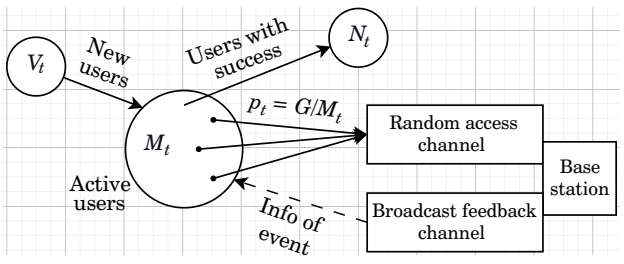
Figure 3 shows the system operation diagram for the case when all users know the total number of active users. In each frame with number t, new users enter the system, the number of which is equal to V_t. Active users, the number of which is equal to M_t, using information about the number of active users, independently of each other in the EP phase make a decision on transmission with probability $p_t = G/M_t$. The decision to transfer to DTP is then made in accordance with the algorithm. Users who successfully transmit during the DTP phase leave the system. The number of such users is N_t.

The process of changing the number of active users is described by the following equality:

$$M_{t+1} = M_t - N_t + V_t.$$

It follows from this expression that the sequence $\{M_1, \ldots, M_t\}$ is homogeneous irreducible aperiodic Markov chain. Based on the results from [14], we will understand the throughput as the upper bound of the input arrival rate, up to which the Markov chain is ergodic:

$$T(G, L) = \sup_{\lambda} \{\lambda : Markov\,chain\,ergodic\}.$$



■ ***Fig. 3.*** Scheme of the system operation with a known number of active users

Using approaches from the works [14], it can be shown that the value of $T(G, L)$ can be calculated using the following equation:

$$T(G, L) = \lim_{n\to\infty} \Pr\{B_t \mid M_t = n\}. \quad (1)$$

Let us show how to calculate the probability $\Pr\{B_t|M_t = n\}$. Based on the definition given earlier, this probability can be calculated as follows:

$$\Pr\{B_t \mid M_t = n\} = np_{EP}\left(1 - p_{Ep}\right)^{n-1} +$$
$$+ \sum_{i=2}^{n} C_n^i p_{EP}^i \left(1 - p_{EP}\right)^{n-1} \Pr\{B_t \mid i\} +$$
$$+ \left(1 - p_{EP}\right)^n np_{DTP}\left(1 - p_{DTP}\right)^{n-1},$$

where $\Pr\{B_t|\ i\}$ — is the probability of the event "Success" for i users and L preambles.

Probability $\Pr\{B_t|\ i\}$ was calculated in [14]. In this case, substituting the values $p_{EP} = p_{DTP} = G/M_t$, formula (1) will have the following form:

$$T(G,L) = \lim_{n\to\infty} \Pr\{B_t \mid M_t = n\} = Ge^{-G} +$$
$$+ \sum_{i=2}^{\infty} \frac{G^i}{i!} e^{-G} \sum_{j=2}^{\min(i,L)} C_L^{L-j} \sum_{v=0}^{j} (-1)^v \times$$
$$\times C_j^v \left(1 - \frac{L-j+v}{L}\right)^i i \frac{1}{j}\left(1 - \frac{1}{j}\right)^{i-1} + Ge^{-2G}.$$

From the above formula it follows that for a given fixed L, the value of G affects the throughput. The function under consideration is monotone and unimodal for all $G \in (0; \infty)$. Thus, the throughput can be maximized for a given number of preambles L by the parameter G. Let us introduce the following notations: $T_{\max}(L) = \max_G T(G, L)$ and also $G_{\text{opt}}(L) = \arg\max_G T(G, L)$. The results of calculating these values for different numbers of preambles will be presented below in Table 6.

Throughput using the procedure for estimating the number of active users

In real systems, there is no way to reliably know the number of active users at the beginning of each frame to determine the probability of transmission during EP. To get closer to real conditions, one can use some procedure for estimating the number of active users in a frame, based on the approach proposed in [18]. In this case, assumption 6 is excluded from the system of assumptions.

Figure 4 shows the system operation diagram for the case when the number of active users is unknown. In each frame with number t, new users enter the system, the number of which is equal to V_t. The base station calculates the value of S_t according to a certain rule and transmits it via the feedback channel. Active users, the number of which is equal to M_t, using S_t, independently of each other in the EP phase make a decision on transmission with probability $p_t = G/S_t$. The decision to transfer to DTP is then made in accordance with the algorithm. Users who successfully transmit during the DTP phase leave the system. The number of such users is N_t.

The process of changing the estimate of the number of active users is described by the following equality (where $S_1 = 1$):

$$S_{t+1} = \max\left[1,\ S_t + aI\{A_t\} + bI\{B_t\} + cI\{C_t\}\right],$$

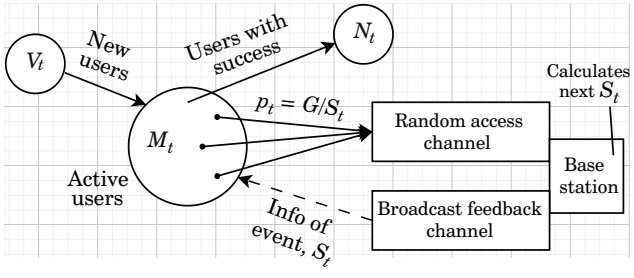
where $I\{*\}$ — is the indicator function. The coefficients a, b and c are defined as follows:

$$a = Q - 1;\ b = Q - 1;\ c = \frac{2}{(e-2)} + Q,$$

where Q — is an estimation parameter that depends on the number of unique preambles.

Next, the probability of p_{EP} is calculated using the following formula:

$$P_{EP} = \min\left(1,\ \frac{G}{S_t}\right).$$



■ ***Fig. 4.*** Scheme of the system operation with an estimated number of active users

Based on the above, the sequence of pairs (M_t, S_t) is a two-dimensional Markov chain.

Using the results of work [18], to determine the ergodicity conditions of a given Markov chain, we calculate the average drift for each component of a given Markov chain:

$$E[M_{t+1} - M_t \mid S_t = s,\ M_t = n] = \lambda - \alpha_i;$$
$$E[S_{t+1} - S_t \mid S_t = s,\ M_t = n] = c + (a - c)\beta_i + (b - c)\alpha_i,$$

where α_i — conditional probability of the event "Success" on the DTP with n active users and an estimate s of the number of active users in the frame and β_i — conditional probability of the event "Empty" on the DTP with n active users and an estimate s of the number of active users in the frame.

Remark: The approach to estimating the number of active users for the ALOHA algorithm with a EP (algorithm 1) was considered in [14]. Also in [14] the coefficients a, b and c were proposed, and the optimal coefficient Q was calculated for different numbers of preambles. This paper examines in detail the application of the approach to estimating the number of active users for the 2-step ALOHA algorithm (algorithm 2), the combined ALOHA algorithm (algorithm 3), and provides the main results from [14] for algorithm 1 for comparison.

Table 3 presents the final formula for calculating the coefficients α_i and β_i for algorithm 1 and provides the derivation of formulas for calculating these coefficients for algorithms 2 and 3. In this case, the fact is taken into account that the number of users who decided to transmit to the EP is distributed according to the binomial law with parameters p_{EP} and n.

Following the work [18], to determine the ergodicity conditions, we introduce the following functions $\mu_n(\lambda, k)$ and $\mu_s(k)$:

$$\mu_n(\lambda, k) = \lambda - \lim_{\substack{n\to\infty \\ s\to\infty \\ k=n/s}} \alpha_i;$$

$$\mu_s(k) = c + (a - c) \lim_{\substack{n\to\infty \\ s\to\infty \\ k=n/s}} \beta_i + (b - c) \lim_{\substack{n\to\infty \\ s\to\infty \\ k=n/s}} \alpha_i. \qquad (2)$$

Taking into account that during the limit transition $n \to \infty$, $s \to \infty$ and maintaining a constant relationship $k = n/s$ the binomial distribution transforms into a Poisson distribution with parameter $k = n/s$, and the limits defined above can be calculated using the formulas presented in Table 4.

It follows from [18] that in order to determine whether a Markov chain (M_t, S_t) is ergodic for some value λ, it is necessary to perform the following steps:

■ ***Table 3.*** *Formulas for calculating the coefficients* α_i *and* β_i

α_i	$i=1$	$$\frac{nG}{s}\left(1-\frac{G}{s}\right)^{n-1}+\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\sum_{j=2}^{\min(i,L)}C_L^{L-j}\sum_{v=0}^{j}(-1)^v C_j^v\left(1-\frac{L-j+v}{L}\right)^i\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}$$
	$i=2$	$$np_{EP}(1-p_{EP})^{n-1}+np_{DTP}(1-p_{DTP})^{n-1}\left(\sum_{i=2}^{n}C_n^i p_{EP}^i(1-p_{EP})^{n-i}\left(1-\left(\frac{1}{L}\right)^i\right)+(1-p_{EP})^n\right)=$$ $$=\frac{nG}{s}\left(1-\frac{G}{s}\right)^{n-1}+\frac{nG}{s}\left(1-\frac{G}{s}\right)^{n-1}\left(\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\left(1-\left(\frac{1}{L}\right)^i\right)+\left(1-\frac{G}{s}\right)^n\right)$$
	$i=3$	$$np_{EP}(1-p_{EP})^{n-1}+\sum_{i=2}^{n}C_n^i p_{EP}^i(1-p_{EP})^{n-i}\Pr\{B_t\mid i\}+(1-p_{EP})^n np_{DTP}(1-p_{DTP})^{n-1}=$$ $$=\frac{nG}{s}\left(1-\frac{G}{s}\right)^{n-1}+\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\sum_{j=2}^{\min(i,L)}C_L^{L-j}\sum_{v=0}^{j}(-1)^v C_j^v\left(1-\frac{L-j+v}{L}\right)^i\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}+\left(1-\frac{G}{s}\right)^n\frac{nG}{s}\left(1-\frac{G}{s}\right)^{n-1}$$
β_i	$i=1$	$$\left(1-\frac{G}{s}\right)^n+\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\sum_{j=2}^{\min(i,L)}C_L^{L-j}\sum_{v=0}^{j}(-1)^v C_j^v\left(1-\frac{L-j+v}{L}\right)^i\left(1-\frac{1}{j}\right)^i$$
	$i=2$	$$\left(\sum_{i=2}^{n}C_n^i p_{EP}^i(1-p_{EP})^{n-i}\left(1-\left(\frac{1}{L}\right)^i\right)+(1-p_{EP})^n\right)(1-p_{DTP})^n=\left(\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\left(1-\left(\frac{1}{L}\right)^i\right)+\left(1-\frac{G}{s}\right)^n\right)\left(1-\frac{G}{s}\right)^n$$
	$i=3$	$$\sum_{i=2}^{n}C_n^i p_{EP}^i(1-p_{EP})^{n-i}\Pr\{A_t\mid i\}+(1-p_{EP})^n(1-p_{DTP})^n=$$ $$=\sum_{i=2}^{n}C_n^i\left(\frac{G}{s}\right)^i\left(1-\frac{G}{s}\right)^{n-i}\sum_{j=2}^{\min(i,L)}C_L^{L-j}\sum_{v=0}^{j}(-1)^v C_j^v\left(1-\frac{L-j+v}{L}\right)^i\left(1-\frac{1}{j}\right)^i+\left(1-\frac{G}{s}\right)^n\left(1-\frac{G}{s}\right)^n$$

■ ***Table 4.*** *Formulas for calculating limits of the coefficients* α_i *and* β_i

$\lim\limits_{\substack{n\to\infty\\ s\to\infty\\ k=n/s}}\alpha_i$	$i=1$	$$e^{-Gk}\left(Gk+\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j\mid i\}\right)$$
	$i=2$	$$Gke^{-Gk}\left(1+e^{-Gk}\left(\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\left(1-\left(\frac{1}{L}\right)^i\right)+1\right)\right)$$
	$i=3$	$$e^{-Gk}\left(Gk+\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j\mid i\}+Gke^{-Gk}\right)$$
$\lim\limits_{\substack{n\to\infty\\ s\to\infty\\ k=n/s}}\beta_i$	$i=1$	$$e^{-Gk}\left(1+\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum_{j=2}^{\min(i,L)}\left(1-\frac{1}{j}\right)^i\Pr\{j\mid i\}\right)$$
	$i=2$	$$e^{-2Gk}\left(\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\left(1-\left(\frac{1}{L}\right)^i\right)+1\right)$$
	$i=3$	$$e^{-Gk}\left(\sum_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum_{j=2}^{\min(i,L)}\left(1-\frac{1}{j}\right)^i\Pr\{j\mid i\}+e^{-Gk}\right)$$

1) for a fixed value of λ, solve the equation $\mu_n(\lambda, k) = k\mu_s(k)$ for the unknown k;

2) check that the following condition is satisfied for all roots of the equation:

$$\mu_n(\lambda, k_i) < 0,$$
$$\mu_s(k_i) < 0.$$

3) if the condition is satisfied, then the Markov chain for a given value of λ is ergodic.

Taking into account formula (2) and the formula from Table 4, the final formulas for calculating the functions $\mu_n(\lambda, k)$ and $\mu_s(k)$ are presented in Table 5.

Let the number of unique preambles L be given and the throughput value for the exact number of active users be known $T_{\max}(L)$. Then one can numerically find Q for which the two-dimensional Markov chain is ergodic at $\lambda = T_{\max}(L)$. The value of the estimation parameter Q at which the throughput value equal to $T_{\max}(L)$ is achieved will be denot-

■ ***Table 5.*** *Formulas for calculating coefficients* $\mu_n(\lambda, k)$ and $\mu_s(k)$

$\mu_n(\lambda, k)$	$i = 1$	$\lambda - e^{-Gk}\left(Gk + \sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j \mid i\}\right)$
	$i = 2$	$\lambda - Gke^{-Gk}\left(1 + e^{-Gk}\left(\sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\left(1-\left(\frac{1}{L}\right)^i\right)+1\right)\right)$
	$i = 3$	$\lambda - e^{-Gk}\left(Gk + \sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j \mid i\} + Gke^{-Gk}\right)$
$\mu_s(k)$	$i = 1$	$c + (a-c)e^{-Gk}\left(1 + \sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\left(1-\frac{1}{j}\right)^i\Pr\{j \mid i\}\right) + (b-c)e^{-Gk}\left(Gk + \sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j \mid i\}\right)$
	$i = 2$	$c + (a-c)e^{-2Gk}\left(\sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\left(1-\left(\frac{1}{L}\right)^i\right)+1\right) + (b-c)Gke^{-Gk}\left(1 + e^{-Gk}\left(\sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\left(1-\left(\frac{1}{L}\right)^i\right)+1\right)\right)$
	$i = 3$	$c + (a-c)e^{-Gk}\left(\sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\left(1-\frac{1}{j}\right)^i\Pr\{j \mid i\} + e^{-Gk}\right) +$ $+(b-c)e^{-Gk}\left(Gk + \sum\limits_{i=2}^{\infty}\frac{(Gk)^i}{i!}\sum\limits_{j=2}^{\min(i,L)}\frac{i}{j}\left(1-\frac{1}{j}\right)^{i-1}\Pr\{j \mid i\} + Gke^{-Gk}\right)$

■ ***Table 6.*** *Maximum throughput and the corresponding parameter values G and Q for combination algorithm of ALOHA*

L	$T_{\max}(L)$	$G_{\text{opt}}(L)$	$Q_{\text{opt}}(L)$
1	0.522	0.757	0.020
2	0.572	0.900	0.103
4	0.601	1.000	0.188
8	0.617	1.050	0.249
16	0.625	1.080	0.286
32	0.628	1.100	0.305
64	0.630	1.100	0.316
128	0.631	1.100	0.321

■ ***Table 7.*** *Maximum throughput and the corresponding parameter values G and Q for 2-step algorithm of ALOHA*

L	$T_{\max}(L)$	$G_{\text{opt}}(L)$	$Q_{\text{opt}}(L)$
1	0.521	0.76	0.020
2	0.562	0.90	0.247
4	0.582	0.96	0.329
8	0.591	0.98	0.354
16	0.596	0.99	0.362
32	0.598	1.00	0.366
64	0.599	1.00	0.367
128	0.599	1.00	0.367

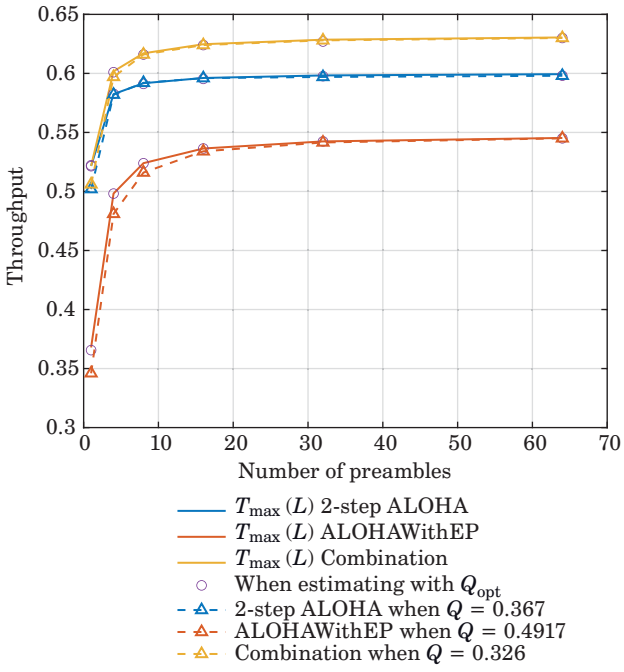
ed as $Q_{opt}(L)$. The values for the combined ALOHA approach are presented in Table 6. Similar values for the 2-step ALOHA algorithm are presented in Table 7.

Impact of the estimation procedure parameter on throughput

The dependences of the throughput on the number of preambles L for the considered algorithms are presented in Fig. 5. The solid lines show the dependence of the throughput on the number of preambles L for a known number of active users. The "circle" marker shows the throughput values for different values of the number of unique preambles L, an unknown number of active users and the use of $Q_{opt}(L)$ in the estimation procedure. The dotted lines with the "triangle" marker mark the corresponding values of the throughput when using a fixed value of Q in the estimation procedure for different numbers of preambles L. This value of Q for each of the algorithms under consideration is chosen as the value of $Q_{opt}(L)$ when the number of preambles L tends to infinity.

The following conclusions can be drawn from the presented dependencies.

1. With an unknown number of active users and using the $Q_{opt}(L)$ value in the estimation procedure, the same values of maximum throughput can be achieved as with exact knowledge of the number of active users.



■ ***Fig. 5.*** Throughput versus number of unique preambles

■ ***Table 8.*** *Values of T(L) with non-optimal value Q*

L	$T_1(L)$	$T_2(L)$	$T_3(L)$
1	0.346	0.502	0.506
2	0.423	0.555	0.564
4	0.481	0.582	0.597
8	0.516	0.592	0.616
16	0.534	0.596	0.624
32	0.542	0.597	0.628
64	0.545	0.598	0.63
128	0.547	0.599	0.631

2. When the number of active users is unknown and a fixed value of Q is used in the estimation procedure, the maximum throughput values are lower than $T_{max}(L)$. However, this difference rapidly decreases as the number of preambles increases. Next, we will show how to quantitatively evaluate the value of the relative gain.

The quantitative value of the relative loss can be calculated using the following formula:

$$\Delta_i = \frac{T_{max,i}(L) - T_i(L)}{T_{max,i}(L)},$$

where i — index of the algorithm; $T_{max,i}(L)$ — maximum throughput for the algorithm with index i, the values of $T_i(L)$ for each of the algorithms are presented in Table 8.

The relative loss when using a non-optimal Q (in our work, we take Q for an infinite number of preambles) does not exceed 6% for any number of preambles, and starting with $L = 30$, the loss is no more than 0.2%.

Note that for algorithm 1 the values in Table 8 were obtained with $Q = 0.491$, for algorithm 2 the values in Table 8 were obtained with $Q = 0.367$ and for algorithm 3 the values in Table 8 were obtained with $Q = 0.326$.

Conclusion

The paper presents a comparative analysis of random multiple access algorithms with early feedback based on ALOHA. A model of a system is considered, built on the basis of a model of a system with random multiple access and early feedback, proposed in [12]. Unlike work [12], the system has only one channel, a limited number of preambles, and to avoid message losses due to conflicts in the multiple access channel, retransmissions are carried out in accordance with a certain algorithm. We briefly describe two previously considered algo-

rithms based on ALOHA with early feedback from [14] and [15] and introduce a new algorithm, which is a combination of the works [14, 15]. The operation of these algorithms is described from a unified point of view. For all three algorithms presented in this paper, two variants are considered: for the first variant, the number of active users is considered known, for the second variant, the number of active users is unknown and some estimation procedure is used. For each of the three algorithms, a procedure for estimating the number of active users is described, which is specified by one parameter Q. For some values of the number of preambles, the optimal parameter Q is determined to maximize the system throughput. For each algorithm, the dependence of the throughput on the number of unique preambles is investigated for the first and second variants. It is shown that when using the optimal value of Q, the throughput of the second variant is equal to the throughput of the first variant, that is, the considered estimation procedure allows, in the absence of information on the number of active users, to obtain the same throughput as in the presence of such information. It is also shown that when using a fixed value of the parameter Q for any number of preambles, equal to the optimal Q for an infinite number of preambles, the throughput loss is no more than 6% and decreases rapidly with increasing number of preambles. Starting from $L = 30$ the loss does not exceed 0.2%.

The obtained results show that the combined algorithm proposed for the first time in this paper, obtained on the basis of algorithms from works [14] and [15], allows obtaining the highest value of throughput for any number of unique preambles. In addition, this modification of the algorithm allows exceeding the maximum value of the throughput $e^{-1}(2 - e^{-1}) \approx 0.6004$ for systems with early feedback, first presented in [12] for a system without retransmissions and an unlimited number of unique preambles.

The considered combined algorithm based on ALOHA with early feedback can be used in the RA channel of future generations of wireless networks.

Financial support

The paper was prepared with the financial support of the Russian Science Foundation, project No. 22-19-00305-P “Spatial-temporal stochastic models of wire less networks with a large number of users”, https://rscf.ru/project/22-19-00305/.

References

1. Kodheli O. Random access procedure over non-terrestrial networks: From theory to practice. *IEEE Access*, 2021, vol. 9, pp. 109130–109143. doi:10.1109/ACCESS.2021.3101291
2. Amatetti C. A novel two fold approach to enhance NB-IoT MAC procedure in NTN. *IEEE Journal on Selected Areas in Communications*, 2024, vol. 42, no. 5, pp. 1453–1464. doi:10.1109/JSAC.2024.3365868
3. Zhou H. Novel random access schemes for small data transmission. *IEEE International Conference on Communications (ICC)*, IEEE, 2022, pp. 1992–1997. doi:10.1109/ ICC45855.2022.9839227
4. Song S., Seo J. B., Jin H. Online control of two-step random access: A step towards uMTC. *2023 IEEE Wireless Communications and Networking Conference (WCNC)*, IEEE, 2023, pp. 1–6. doi:10.1109/WCNC55385.2023.10118949
5. Alvi M. Mini-slot based access barring scheme for IoT networks. *Human-Centric Computing and Information Sciences*, 2023, no. 13. doi:10.22967/HCIS.2023.13.032
6. Zheng K. Dynamic beam-based random access scheme for M2M communications in massive MIMO systems. *IEEE Transactions on Vehicular Technology*, 2023, vol. 72, no. 11, pp. 14531–14542. doi:10.1109/TVT.2023.3286660
7. Pennanen H. 6G: The intelligent network of everything. *IEEE Access*, 2024, vol. 13, pp. 1319–1421. doi:10.1109/ACCESS.2024.3521579
8. Dao N. N., Tu N. H., Hoang T. D., Nguyen T. H., Nguyen L. V., Lee K. A review on new technologies in 3GPP standards for 5G access and beyond. *Computer Networks*, 2024, vol. 245, pp. 110370. doi:10.1016/j.comnet.2024.110370
9. Gregoratti D., Arteaga X., Broquetas J. Mathematical properties of the Zadoff – Chu sequences. *arXiv preprint*, 2023. arXiv:2311.01035. doi:10.48550/arXiv.2311.01035
10. Andrews J. G. A primer on Zadoff – Chu sequences. *arXiv preprint*, 2022. arXiv.2211.05702. doi:10.48550/arXiv.2211.05702
11. Choi J. Random access techniques for machine – type communication. *Next Generation Multiple Access*, 2024, pp. 259–285. doi:10.1002/9781394180523.ch11
12. Choi J. On improving throughput of multichannel ALOHA using preamble-based exploration. *Journal of Communications and Networks*, 2020, vol. 22, no. 5, pp. 380–389. doi:10.1109/JCN.2020.000024
13. Plastras S. Non-terrestrial networks for energy-efficient connectivity of remote IoT devices in the 6G era: A survey. *Sensors*, 2024, vol. 24, no. 4, pp. 1227. doi:10.3390/s24041227
14. Burkov A. A., Rachugin R. O., Turlikov A. M. The impact of the number of unique preambles on the stability region of the ALOHA algorithm with early feed-

back. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2024, no. 6, pp. 58–65. doi:10.31799/1684-8853-2024-6-58-65, EDN: FKCCQN

15. Burkov A., Rachugin R., Turlikov A. Throughput two-step random multiple access based on ALOHA with early-feedback. *2025 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*, IEEE, 2025, pp. 1–5. doi:10.1109/WECONF65186.2025.11017015

16. Bogatyrev V. A., Bogatyrev S. V., Bogatyrev A. V. Control of multipath transmissions in the nodes of switching segments of reserved paths. *2022 International Conference on Information, Control, and Communication Technologies (ICCT)*, 2022, pp. 1–5. doi:10.1109/ICCT56057.2022.9976839

17. Arustamov S. A., Bogatyrev V. A., Polyakov V. I. Back up data transmission in real-time duplicated computer systems. *Advances in Intelligent Systems and Computing*, 2016, vol. 451, pp. 103–109. doi:10.1007/978-3-319-33816-3_11

18. Mikhailov V. A. Geometrical analysis of the stability of Markov chains in R_+^n and its application to throughput evaluation of the adaptive random multiple access algorithm. *Problemy of Information Transmission*, 1988, vol. 24, no. 1, pp. 61–73 (In Russian). doi:10.31799/1684-8853-2020-3-79-85

УДК 004.728.3.057.4
doi:10.31799/1684-8853-2025-6-74-84
EDN: EHEWUI

Сравнительный анализ алгоритмов случайного множественного доступа с ранней обратной связью, построенных на базе АЛОХА

А. А. Бурков[a], канд. техн. наук, orcid.org/0000-0002-0920-585X
Р. О. Рачугин[a], аспирант, orcid.org/0000-0001-5813-3867
А. М. Тюрликов[a], доктор техн. наук, профессор, orcid.org/0000-0001-7132-094X, turlikov@guap.ru
[a]Санкт-Петербургский государственный университет аэрокосмического приборостроения, Б. Морская ул., 67, Санкт-Петербург, 190000, РФ

Введение: одним из вариантов повышения пропускной способности системы случайного множественного доступа является использование ранней обратной связи, под которой понимается быстрый ответ базовой станции после приема преамбул. В работе ранняя обратная связь рассматривается для систем множественного доступа, использующих разные алгоритмы на базе АЛОХА. **Цель:** провести сравнительный анализ зависимости максимальной пропускной способности от числа уникальных преамбул алгоритмов случайного доступа на основе алгоритма АЛОХА с ранней обратной связью. **Результаты:** рассмотрены алгоритм АЛОХА с фазой исследования, 2-шаговый алгоритм АЛОХА и комбинация двух этих алгоритмов. Проведен сравнительный анализ алгоритма АЛОХА с фазой исследования, 2-шагового алгоритма АЛОХА и комбинированного алгоритма для варианта с известным числом уникальных абонентов и варианта с оценкой числа активных абонентов. Показано, что использование процедуры оценки числа активных абонентов позволяет достичь аналогичных значений зависимости максимальной пропускной способности от числа уникальных преамбул, что и для первого варианта с известным числом активных абонентов. Также демонстрируется, что использование фиксированного параметра, влияющего на процедуру оценки, равного оптимальному значению этого параметра для бесконечного числа преамбул, имеет проигрыш, не превышающий 6 % для любого числа преамбул и 0,2 % при 30 и более преамбулах. **Практическая значимость:** предложен новый алгоритм на базе АЛОХА с ранней обратной связью, позволяющий повысить максимальную пропускную способность системы по сравнению с ранее известными алгоритмами этого класса. Данный алгоритм может быть использован в канале случайного доступа сетей будущих поколений. **Обсуждение:** в рамках проведенного анализа не учитывалось влияние числа каналов, используемых в системе, что может быть дальнейшим направлением исследования.

Ключевые слова — АЛОХА, случайный доступ без выделения грантов, фаза исследования на основе преамбул, оценка, пропускная способность, эргодичность, марковская цепь.

Для цитирования: Burkov A. A., Rachugin R. O., Turlikov A. M. Comparative analysis of ALOHA based algorithms with early feedback. *Информационно-управляющие системы*, 2025, № 6, с. 74–84. doi:10.31799/1684-8853-2025-6-74-84, EDN: EHEWUI

For citation: Burkov A. A., Rachugin R. O., Turlikov A. M. Comparative analysis of ALOHA based algorithms with early feedback. *Informatsionno-upravliaiushchie sistemy* [Information and Control Systems], 2025, no. 6, pp. 74–84. doi:10.31799/1684-8853-2025-6-74-84, EDN: EHEWUI

АКИМОВ Андрей Анатольевич

Доцент кафедры инструментального и прикладного программного обеспечения Института информационных технологий МИРЭА – Российского технологического университета, Москва.
В 1997 году окончил Стерлитамакский государственный педагогический институт по специальности «Математика и информатика».
В 2006 году защитил диссертацию на соискание ученой степени кандидата физико-математических наук.
Является автором 50 научных публикаций и девяти свидетельств о регистрации программ для ЭВМ.
Область научных интересов – дифференциальные уравнения, методы оптимального управления, прикладная информатика.
Эл. адрес: akimov_a@mirea.ru

БУРКОВ Артем Андреевич

Доцент кафедры инфокоммуникационных технологий и систем связи Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 2017 году окончил Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Инфокоммуникационные технологии и системы связи».
В 2023 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором 25 научных публикаций.
Область научных интересов – беспроводные системы передачи данных, алгоритмы случайного множественного доступа, системы с гибридной решающей обратной связью и др.
Эл. адрес: a.burkov@k36.org

ГОРБУНОВА Анастасия Владимировна

Старший научный сотрудник Института проблем управления им. В. А. Трапезникова РАН, Москва.
В 2010 году окончила магистратуру Российского университета дружбы народов по специальности «Прикладная математика и информатика».
В 2017 году защитила диссертацию на соискание ученой степени кандидата физико-математических наук.
Является автором 39 научных публикаций.
Область научных интересов – математическая теория телетрафика, теория массового обслуживания, прикладная теория вероятностей, имитационное моделирование и др.
Эл. адрес: avgorbunova@list.ru

БОЛБАКОВ Роман Геннадьевич

Доцент, заведующий кафедрой инструментального и прикладного программного обеспечения Института информационных технологий МИРЭА – Российского технологического университета, Москва.
В 2008 году окончил Московский государственный технический университет радиотехники, электроники и автоматики по специальности «Программная инженерия».
В 2013 году защитил диссертацию на соискание ученой степени кандидата технических наук.
Является автором 205 научных публикаций и двух свидетельств о регистрации программ для ЭВМ.
Область научных интересов – системный анализ, программное обеспечение, оптимальные задачи, прикладные методы обработки информации.
Эл. адрес: bolbakov@mirea.ru

ГНАТЕНКО Юлия Ахнафовна

Доцент кафедры математического моделирования Стерлитамакского филиала Уфимского университета науки и технологий.
В 2002 году окончила Стерлитамакский государственный педагогический институт по специальности «Математика и информатика».
В 2006 году защитила диссертацию на соискание ученой степени кандидата физико-математических наук.
Является автором 39 научных публикаций.
Область научных интересов – математическое моделирование, численные методы, разработка программных комплексов.
Эл. адрес:
y.a.gnatenko@struust.ru

ЖУКОВА Наталия Александровна

Доцент, ведущий научный сотрудник лаборатории технологий больших данных социокиберфизических систем Санкт-Петербургского Федерального исследовательского центра РАН.
В 2005 году окончила Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В. И. Ульянова (Ленина) по специальности «Прикладная математика и информатика».
В 2019 году защитила диссертацию на соискание ученой степени доктора технических наук.
Является автором более 100 научных публикаций.
Область научных интересов – интеллектуальные системы, анализ данных.
Эл. адрес: nazhukova@mail.ru

ИСАЕВА Мария Николаевна

Старший преподаватель кафедры инфокоммуникационных технологий и систем связи Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 2020 году окончила Санкт-Петербургский государственный университет аэрокосмического приборостроения по специальности «Информационная безопасность».
Является автором 27 научных публикаций.
Область научных интересов — теория кодирования, кодовая криптография, цифровая стеганография.
Эл. адрес: imn@guap.ru

ИСАЕВА Ольга Сергеевна

Ведущий научный сотрудник отдела Регионального научно-образовательного математического центра «Красноярский математический центр» Института вычислительного моделирования СО РАН.
В 1998 году окончила Красноярский государственный университет по специальности «Прикладная математика».
В 2022 году защитила диссертацию на соискание ученой степени доктора технических наук.
Является автором более 100 научных публикаций и 13 свидетельств о регистрации программ для ЭВМ.
Область научных интересов — методы искусственного интеллекта, анализ данных, цифровые двойники.
Эл. адрес: isaeva@icm.krasn.ru

КОВАЛЕВСКИЙ Владислав Эдуардович

Аспирант, младший научный сотрудник лаборатории технологий больших данных социокиберфизических систем Санкт-Петербургского Федерального исследовательского центра РАН.
В 2010 году окончил магистратуру Санкт-Петербургского государственного политехнического университета по направлению «Информатика и вычислительная техника».
Является автором десяти научных публикаций и двух патентов на изобретения.
Область научных интересов — автоматизированное машинное обучение, мета-обучение, оптимизация машинного обучения.
Эл. адрес: darkeol@mail.ru

НИТКИН Иван Сергеевич

Аспирант факультета безопасности информационных технологий Университета ИТМО, ассистент Института киберфизических систем Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 2024 году окончил магистратуру Университета ИТМО по направлению «Информационная безопасность» со специализацией «Криптографические средства защиты информации».
Является автором восьми научных публикаций.
Область научных интересов — постквантовая криптография, криптография на основе корректирующих кодов, теория кодирования.
Эл. адрес: exebopen@gmail.com

РАЧУГИН Роман Олегович

Аспирант кафедры инфокоммуникационных технологий и систем связи Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 2023 году окончил магистратуру Санкт-Петербургского государственного университета аэрокосмического приборостроения по специальности «Инфокоммуникационные технологии и системы связи».
Является автором двух научных публикаций.
Область научных интересов — теория разрядных вычислений, методы проектирования спецпроцессоров для систем контроля и управления, оптико-информационные системы.
Эл. адрес: rro1699@gmail.com

ТЮРЛИКОВ Андрей Михайлович

Профессор, заведующий кафедрой инфокоммуникационных технологий и систем связи Санкт-Петербургского государственного университета аэрокосмического приборостроения.
В 1980 году окончил Ленинградский институт авиационного приборостроения по специальности «Информационные системы управления».
В 2011 году защитил диссертацию на соискание ученой степени доктора технических наук.
Является автором более 150 научных публикаций.
Область научных интересов — многоабонентные системы связи, системы дистанционного обучения, протоколы передачи данных в реальном масштабе времени, алгоритмы сжатия видеоинформации.
Эл. адрес: turlikov@guap.ru

СОДЕРЖАНИЕ ЖУРНАЛА «ИНФОРМАЦИОННО-УПРАВЛЯЮЩИЕ СИСТЕМЫ» ЗА 2025 г. [№ 1–6]

Уважаемые авторы!

При подготовке рукописей статей необходимо руководствоваться следующими рекомендациями.

Статьи должны содержать изложение новых научных результатов. Название статьи должно быть кратким, но информативным. В названии недопустимо использование сокращений, кроме самых общепринятых (РАН, РФ, САПР и т. п.).

Текст рукописи должен быть оригинальным, а цитирование и самоцитирование корректно оформлено.

Объем статьи (текст, таблицы, иллюстрации и библиография) не должен превышать эквивалента в 20 страниц, напечатанных на бумаге формата А4 на одной стороне через 1,5 интервала Word шрифтом Times New Roman размером 13, поля не менее двух сантиметров.

Обязательными элементами оформления статьи являются: индекс УДК, заглавие, инициалы и фамилия автора (авторов), ученая степень, звание (при отсутствии — должность), полное название организации, аннотация и ключевые слова на русском и английском языках, ORCID и электронный адрес одного из авторов. При написании аннотации не используйте аббревиатур и не делайте ссылок на источники в списке литературы. Предоставляйте подрисуночные подписи и названия таблиц на русском и английском языках.

Статьи авторов, не имеющих ученой степени, рекомендуется публиковать в соавторстве с научным руководителем, наличие подписи научного руководителя на рукописи обязательно; в случае самостоятельной публикации обязательно предоставляйте заверенную по месту работы рекомендацию научного руководителя с указанием его фамилии, имени, отчества, места работы, должности, ученого звания, ученой степени.

Простые **формулы** набирайте в Word, сложные с помощью редактора Mathtype или Equation. Для набора одной формулы не используйте два редактора; при наборе формул в формульном редакторе знаки препинания, ограничивающие формулу, набирайте вместе с формулой; для установки размера шрифта в Mathtype никогда не пользуйтесь вкладкой Other, Smaller, Larger, используйте заводские установки редактора, не подгоняйте размер символов в формулах под размер шрифта в тексте статьи, не растягивайте и не сжимайте мышью формулы, вставленные в текст; пробелы в формуле ставьте только после запятых при перечислении с помощью Ctrl+Shift+Space (пробел); не отделяйте пробелами знаки: + = – ×, а также пространство внутри скобок; для выделения греческих символов в Mathtype полужирным начертанием используйте Style → Other → bold.

Для набора формул в Word никогда не используйте вкладки: «Уравнение», «Конструктор», «Формула» (на верхней панели: «Вставка» — «Уравнение»), так как этот ресурс предназначен только для внутреннего использования в Word и не поддерживается программами, предназначенными для изготовления оригинал-макета журнала.

При наборе символов в тексте помните, что символы, обозначаемые латинскими буквами, набираются светлым курсивом, русскими и греческими — светлым прямым, векторы и матрицы — прямым полужирным шрифтом.

Подробнее см. http://i-us.ru/index.php/ius/author-guide

Иллюстрации:

— рисунки, графики, диаграммы, блок-схемы предоставляйте в виде отдельных исходных файлов, поддающихся редактированию, используя векторные программы: Visio (*.vsd, *.vsdx); Adobe Illustrator (*.ai); Coreldraw (*.cdr, версия не выше 15); Excel (*.xls); Word (*.docx); AutoCad, Matlab (экспорт в PDF, EPS, SVG, WMF, EMF); Компас (экспорт в PDF); веб-портал DRAW.IO (экспорт в PDF); Inkscape (экспорт в PDF);

— фото и растровые — в формате *.tif, *.png с максимальным разрешением (не менее 300 pixels/inch).

Наличие подрисуночных подписей и названий таблиц на русском и английском языках обязательно (желательно не повторяющих дословно комментарии к рисункам в тексте статьи).

В редакцию предоставляются:

— сведения об авторе (фамилия, имя, отчество, место работы, должность, ученое звание, учебное заведение и год его окончания, ученая степень и год защиты диссертации, область научных интересов, количество научных публикаций, домашний и служебный адреса и телефоны, e-mail), фото авторов: анфас, в темной одежде на белом фоне, должны быть видны плечи и грудь, высокая степень четкости изображения без теней и отблесков на лице, фото можно представить в электронном виде в формате *.tif, *.png, *.jpg с максимальным разрешением — не менее 300 pixels/inch при минимальном размере фото 40×55 мм;

— экспертное заключение;

— экспортное заключение.

Список литературы составляется по порядку ссылок в тексте и оформляется следующим образом:

— для книг и сборников — фамилия и инициалы авторов, полное название книги (сборника), город, издательство, год, общее количество страниц, doi;

— для журнальных статей — фамилия и инициалы авторов, полное название статьи, название журнала, год издания, номер журнала, номера страниц, doi;

— ссылки на иностранную литературу следует давать на языке оригинала без сокращений;

— при использовании web-материалов указывайте адрес сайта и дату обращения.

Список литературы оформляйте двумя отдельными блоками по образцам lit.dot на сайте журнала (http://i-us.ru/paperrules): Литература и References.

Более подробно правила подготовки текста с образцами изложены на нашем сайте в разделе «Руководство для авторов» – http://i-us.ru/index.php/ius/author-guide.

Контакты

Куда: 190000, г. Санкт-Петербург, ул. Большая Морская, д. 67, лит. А, ГУАП, РИЦ
Кому: Редакция журнала «Информационно-управляющие системы»
Тел.: (812) 494-70-02
Эл. почта: ius.spb@gmail.com
Сайт: www.i-us.ru