

ПОВЫШЕНИЕ КАЧЕСТВА ОБРАБОТКИ ДАННЫХ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ НА ОСНОВЕ ФЕДЕРАТИВНОГО РАСПРЕДЕЛЕНИЯ ВЫЧИСЛЕНИЙ

Ермаков С.Р.¹, Зыков С.В.²

¹ФГБОУ ВО «МИРЭА – Российский технологический университет», e-mail: ermakov_s@mirea.ru;

²ФГБОУ ВО «МИРЭА – Российский технологический университет», ФГАОУ ВО «Национальный исследовательский университет Высшая школа экономики», e-mail: zykov_s@mirea.ru, szykov@hse.ru

Цель исследования заключается в повышении качества обработки потоковых данных в интеллектуальных системах на основе федеративного распределения вычислений — синтеза принципов туманных вычислений и федеративного обучения. Предложена модель оптимизации качества, отличающаяся объединением критериев производительности, точности и ресурсоёмкости в единый интегральный функционал потерь. Разработаны методы и алгоритмы адаптивной федеративной потоковой обработки данных, комбинированного режима обучения, преобразования признаков потоковых данных, ассоциативной федеративной агрегации статистических структур и инкрементального дерева решений, обеспечивающие согласованную адаптацию параметров, устойчивость к асинхронности и концептуальному дрейфу, снижение задержек и повышение качества обработки потоковых данных. Экспериментальная проверка на гетерогенном стенде показала снижение средней задержки на и рост пропускной способности на при сохранении стабильной точности и соблюдении ресурсных ограничений. Сформулированы рекомендации по развитию стандартов ISO/IEC TS 8200:2024, ГОСТ Р 59277–2020 и ГОСТ Р 57700.27–2020. Установлено, что согласованная адаптация частоты синхронизаций, степени сжатия и коэффициента забывания в сочетании с потоковой предобработкой и ассоциативной агрегацией обеспечивает устойчивое снижение интегрального функционала потерь. Дальнейшие направления исследований связаны с уточнением границ применимости разработанных решений по нагрузочным профилям и топологиям, а также с совершенствованием механизмов надёжности и отказоустойчивости.

Ключевые слова: интеллектуальные системы, машинное обучение, федеративное обучение, федеративные вычисления, глобально распределенные вычисления.

IMPROVING THE QUALITY OF DATA PROCESSING IN INTELLIGENT SYSTEMS BASED ON FEDERATED DISTRIBUTED COMPUTING

Ermakov S.R.¹, Zykov S.V.²

¹MIREA - Russian Technological University, e-mail: ermakov_s@mirea.ru;

²MIREA - Russian Technological University, National Research University Higher School of Economics, e-mail: zykov_s@mirea.ru, szykov@hse.ru

The aim of the research is to improve the quality of streaming data processing in intelligent systems based on federated distributed computing — a synthesis of fog computing principles and federated learning. A model for quality optimization is proposed, distinguished by the integration of performance, accuracy, and resource consumption criteria into a single integral loss functional. Methods and algorithms have been developed for adaptive federated streaming data processing, combined learning mode, streaming feature transformation, associative federated aggregation of statistical structures, and an incremental decision tree, which together ensure coordinated parameter adaptation, robustness to asynchrony and concept drift, reduced latency, and improved quality of streaming data processing. Experimental validation on a heterogeneous testbed demonstrated a reduction in average latency and an increase in throughput while maintaining stable accuracy and complying with resource constraints. Recommendations have been formulated for the development of standards ISO/IEC TS 8200:2024, GOST R 59277–2020, and GOST R 57700.27–2020. It has been established that coordinated adaptation of synchronization frequency, compression ratio, and forgetting coefficient, combined with streaming preprocessing and associative aggregation, ensures a stable decrease of the integral loss functional. Further research is aimed at refining the applicability boundaries of the proposed solutions across load profiles and topologies, as well as enhancing reliability and fault-tolerance mechanisms.

Keywords: intelligent systems, machine learning, federated learning, federated computing, globally distributed computing.

Введение

Цифровая трансформация перевела интеллектуальные информационно-вычислительные системы (ИС) в разряд критически важных элементов инфраструктуры образования, здравоохранения, промышленности, государственного управления. Современные ИС функционируют как распределённые вычислительные экосистемы, в которых обработка потоков данных осуществляется на множестве географически разнесённых узлов в реальном времени. Трёхзвенная архитектура ИС в стиле «туманных вычислений» — где предполагается облачный, промежуточный и периферийный уровни — призвана обеспечивать масштабируемость и защиту данных [1], однако обостряет инженерные противоречия: необходимо одновременно гарантировать малую задержку отклика, высокую пропускную способность, точность моделей машинного обучения (МО) при концептуальном дрейфе данных (изменении распределений входных данных во времени, вызывающее деградацию точности моделей МО без адаптации) [2] и отказоустойчивость при ограничениях вычислительных и сетевых ресурсов. Это противоречие между возрастающими требованиями и ограниченными возможностями лежит в основе научной задачи по созданию новых моделей, алгоритмов, методов для современных ИС [3].

Сегодня одним из ключевых направлений развития ИС МО является федеративное обучение — подход, при котором модели МО обучаются децентрализованно на периферийных и промежуточных узлах без передачи данных в центр (облако), что минимизирует задержки, снижает нагрузку на сеть и усиливает конфиденциальность [4–6]. Авторы предлагают использовать подход федеративного распределения вычислений (ФРВ) на основе синтеза туманных вычислений и федеративного обучения, который позволяет динамически распределять этапы обучения и инференса (последующего после обучения процесса применения модели МО к новым данным для получения прогнозов, классификаций или решений) между облаком, промежуточным слоем и устройствами, обеспечивая адаптивное управление ресурсами и устойчивость к концептуальному дрейфу.

Существующие решения указанного противоречия оказываются недостаточно эффективными по ряду причин. Во-первых, рост вычислительной сложности и стоимости обучения моделей искусственного интеллекта (ИИ) опережает темпы развития аппаратных средств: затраты на обучение и эксплуатацию крупных моделей растут экспоненциально [7], а масштабирование сталкивается с физическими пределами сети и памяти. Во-вторых, потоковые процессы по своей природе нестационарны: статистика источников изменяется со временем, что вызывает концептуальный дрейф и снижает точность статичных моделей. В-третьих, с увеличением числа узлов возрастает вероятность частичных отказов и рассогласований, что требует архитектуры, способной поддерживать непрерывность обработки даже при деградации каналов связи. В этих условиях необходимы новые математические модели повышения качества, методы адаптивного и федеративного управления потоками вычислений и алгоритмы потокового обучения, которые оптимизируют время отклика, производительность, точность, ресурсоёмкость и надёжность отдельно друг от друга, не изолированно, а в рамках единого системного критерия.

Стоит отметить, что количество параметров флагманских ныне популярных и перспективных больших языковых моделей (БЯМ, англ. – Large Language Models, LLM) МО, которые относятся к высоконагруженным сервисам в ИС, возросло с порядка 6×10^7 (AlexNet, 2012 г.) до сотен миллиардов и триллионов, при этом совокупные вычислительные затраты на их обучение удваиваются в среднем каждые пять месяцев. Энергопотребление центров обработки данных, обеспечивающих обучение и эксплуатацию таких моделей, увеличилось с $\approx 240\text{--}340$ ТВт·ч в 2022 г. до ≈ 460 ТВт·ч в 2024 г. и, по прогнозам, может превысить 1000 ТВт·ч к 2030 г., что сопоставимо с национальным энергопотреблением крупных промышленных держав [8]. Экономическая нагрузка также растёт: если обучение модели GPT-3 оценивалось в 2–4 млн долл., то GPT-4 потребовало 41–78 млн долл. только на вычислительные ресурсы, а стоимость мультимодальных тренировок достигает 120–190 млн долл [9,10]. Одновременно наблюдается повсеместное внедрение ИИ-технологий: к 2025 г. их используют более 78% организаций, например, в сфере здравоохранения США число ИИ-решений, одобренных профильным управлением, увеличилось с 6 до 223 за 2015–2023 г.г. [11]. Совокупность этих факторов свидетельствует, что традиционная централизованная стратегия «силового масштабирования» достигает пределов эффективности, и дальнейшее развитие ИИ возможно лишь за счёт совершенствования архитектур, методов оптимизации и адаптивных алгоритмов, минимизирующих зависимость от объёма вычислительных ресурсов [12].

Первоочередной целью данного исследования является повышение качества обработки потоковых данных в ИС на основе ФРВ (которые по определению являются глобально распределёнными ИС). Также затрагиваются аспекты обеспечения надёжности и отказоустойчивости в указанных системах. В работе применён комплекс математических методов, основанный на положениях теории массового обслуживания, теории оптимального управления и многокритериальной оптимизации, вероятностно-статистической теории и теории устойчивости, а

также на элементах функционального анализа и линейной алгебры. В исследовании решена научная задача по созданию новых моделей, методов и алгоритмов для достижения указанной цели. На обсуждение выносятся следующий ряд результатов:

— разработана модель оптимизации качества обработки данных, отличающаяся объединением критериев производительности, качества и ресурсоёмкости в единый функционал ИС на основе ФРВ, учитывающая актуальность обновлений и показатель концептуального дрейфа, что математически обеспечивает обоснованное описание и оптимизацию качества обработки данных в условиях изменяющихся нагрузок и параметров функционирования указанных систем;

— разработан метод адаптивной федеративной потоковой обработки данных, новизна которого заключается в разработке событийно-управляемого механизма динамической настройки частоты синхронизации и степени сжатия обновлений на основе стохастической аппроксимации и условий устойчивости очередей, обеспечивающего адаптацию к изменению нагрузки и параметров каналов связи;

— разработан метод комбинированного режима обучения, отличающийся определением доли глобальных и локальных итераций на основе решения задачи многокритериальной оптимизации с учётом ограничений по связи и надёжности, что обеспечивает баланс между качеством обучения и коммуникационными издержками;

— разработан метод преобразования признаков потоковых данных, научная новизна которого заключается в потоковой адаптации нормировки, декорреляции и редукции размерности к текущему дрейфу распределений с использованием экспоненциального сглаживания и устойчивого оценивания ковариаций, что обеспечивает робастность к нестационарности и согласование статистик между узлами;

— разработан алгоритм ассоциативной федеративной агрегации статистических структур, новизна которого заключается во введении весов актуальности и алгебраической свёртке моментов в ассоциативной и коммутативной форме, что обеспечивает порядокнезависимую и устойчивую к асинхронности агрегацию обновлений;

— разработан алгоритм инкрементального дерева решений, отличающийся применением неравенства Чебышёва и экспоненциально-взвешенной адаптации порогов, что обеспечивает контролируемую ошибку классификации и устойчивость к мультимодальности без полной перестройки структуры дерева.

В исследовании экспериментально показано, что при применении разработанных моделей, методов и алгоритмов средняя задержка уменьшается примерно на 30–35%, пропускная способность возрастает на 45–60%, а точность обработки данных остаётся стабильной.

Результаты данного исследования позволяют в развитие стандартов ISO/IEC TS 8200:2024 [13], ГОСТ Р 59277–2020 [14] и ГОСТ Р 57700.27–2020 [15] рекомендовать включение в понятийный аппарат и классификацию вычислительных систем искусственного интеллекта категорий *федеративная архитектура*, *федеративное обучение* и *федеративное распределение вычислений* как ключевых характеристик современных ИИ-платформ. Предлагается закрепить положения о децентрализованном обучении и инференсе, выполняемых с учётом актуальности данных и параметров среды, а также определить требования к согласованности и актуальности параметров моделей МО, наличие встроенного мониторинга качества обработки потоковых данных и устойчивости работы при асинхронности вычислений. В документах следует дополнительно отразить методы федеративной агрегации и адаптивного управления параметрами как рекомендуемые механизмы обеспечения доверенности, надёжности и производительности интеллектуальных систем.

Модель оптимизации

В целях повышения качества обработки потоковых данных в ИС на основе ФРВ разработана аналитическая модель оптимизации, обеспечивающая согласованный учёт трёх групп критериев — производительности, точности и ресурсоёмкости — при внешнем ограничении по отказоустойчивости. Задача оптимизации формулируется как

$$x^* = \arg \min_{x \in D} J(x; \theta), \quad (1)$$

где θ — экзогенные условия (характеристики потока данных, сети и оборудования), x — управляемые параметры (τ_{sync} — период синхронизации; γ — степень компрессии обновлений; M_{tot} — суммарный доступный объём памяти; N — число исполнителей/узлов), y — выходные показатели (время отклика τ_{resp} , пропускная способность Q_{proc} , масштабируемость K_{scal} , точность Acc , эффективность η). Параметры θ перечислены в Таблице 1 (где ЕИ — единицы измерения).

Допущения. Вводятся понятия коммуникационной задержки t_{comm} и δ_{sync} коммуникационной доли (доли коммуникационных затрат):

$$t_{\text{comm}} = \tau_{\text{agg}} + \frac{\gamma S_{\text{model}}}{B}, \quad (2)$$

$$\delta_{\text{sync}} = \min \left\{ 1, \frac{t_{\text{comm}}}{\tau_{\text{sync}}} \right\}. \quad (3)$$

Таблица 1 – Набор параметров θ

Обозначение	Наименование	Категория и размерность параметра	ЕИ	Метод определения параметра
λ_{in}	Интенсивность входного потока	Скаляр, непрерывный	запросы/с	Определяется по статистике поступления событий за интервал наблюдения.
S_{model}	Размер обновления модели	Скаляр, непрерывный	ГБ	Измеряется по объёму сериализованных параметров модели при передаче от клиента к агрегатору (облаку).
μ_1	Интенсивность обслуживания одним исполнителем	Скаляр, непрерывный	запросы/с/исп.	Оценивается экспериментально по среднему времени отклика исполнителя.
M_{work}	Рабочий объём памяти узла	Скаляр, непрерывный	ГБ	Измеряется как средний объём используемой оперативной памяти процесса при стабильной нагрузке.
Acc_{max}	Максимальная достижимая точность	Безразмерный	–	Устанавливается по результатам обучения при нулевом концептуальном дрейфе и полной синхронизации.
S_{drift}	Скорость концептуально го дрейфа	Скаляр, непрерывный	1/с	Вычисляется по относительному изменению распределения входных данных между окнами.
B	Пропускная способность канала	Скаляр, непрерывный	ГБ/с	Измеряется как отношение объёма переданных данных к времени передачи на выбранном канале связи.
M_{cap}	Общий объём доступной памяти	Скаляр, непрерывный	ГБ	Определяется по технической спецификации оборудования узла или кластера.
Q_{req}	Требуемая пропускная способность	Скаляр	запросы/с	Устанавливается в соответствии с требованиями к системе.
τ_{SLA}	Допустимая задержка отклика	Скаляр	с	Определяется требований к системе как максимальное время обработки запроса.
R_{min}	Минимально допустимая отказоустойчивость	Безразмерный	–	Устанавливается как целевое значение интегральной вероятности доступности пути «облако-узел» в архитектуре ФРВ.

Приведённые выражения формализуют влияние периода синхронизации, степени компрессии и пропускной способности канала на коммуникационные затраты системы. Отношение $\gamma S_{model}/(B\tau_{sync})$ служит безразмерным показателем использования сетевого ресурса (коммуникационной квотой).

Масштабируемость и эффективная интенсивность обслуживания задается соотношениями:

$$S(N) = \frac{N}{1 + \alpha(N - 1) + \beta N(N - 1)}, \quad (4)$$

$$K_{scal} = \frac{S(N)}{N} (1 - \delta_{sync}), \quad (5)$$

$$\sigma(M_{tot}) = \min \left(1, \frac{M_{tot}}{M_{work}} \right), \quad (6)$$

$$\mu = \mu_1 N K_{scal} \sigma(M_{tot}). \quad (7)$$

Такая форма выражений учитывает одновременный доступ процессов к общим ресурсам и влияние непараллелизуемых участков вычислений, что может ограничивать рост производительности системы.

Интерпретация коэффициента масштабируемости K_{scal} соответствует классическим закономерностям, описанным законами Амдала и Густафсона [16].

Для системы М/М/1 [17] (т.е. при допущениях пуассоновского входного потока и экспоненциального распределения времени обслуживания) справедливы зависимости для загрузки ρ в канале обработки на интервале адаптации параметров, среднего времени отклика системы τ_{resp} и фактической пропускной способности Q_{proc} :

$$\rho = \frac{\lambda_{in}}{\mu} < 1, \quad (8)$$

$$\tau_{resp} = \frac{1}{\mu - \lambda_{in}}, \quad (9)$$

$$Q_{proc} = \min\{\mu, \lambda_{in}\}, \quad (10)$$

Таким образом, модель предполагает квазистационарность потока и служит простейшей аппроксимацией процесса обслуживания.

Точность модели МО под воздействием концептуального дрейфа и старения параметров:

$$\Delta = \tau_{sync} + t_{comm}, Acc = [Acc_{max} - \kappa_d S_{drift} \Delta - \kappa_c (1 - C_{cons}) - \kappa_s (1 - A_{stab})]_0^1, \#(11)$$

где $[\cdot]_0^1$ — отсечение на $[0,1]$.

Частные критерии и интегральный функционал. Нормированные частные критерии принимаются в линейно-кусочной форме со штрафными функциями по нарушениям границ $[z]_+ = \max\{z, 0\}$:

$$\Phi_P = w_\tau \frac{\tau_{resp}}{\tau_{SLA}} + w_Q \left[\frac{Q_{req} - Q_{proc}}{Q_{req}} \right]_+ + w_K (1 - K_{scal}), \quad (12)$$

$$\Phi_A = u_{Acc} (1 - Acc) + u_{cons} (1 - C_{cons}) + u_{stab} (1 - A_{stab}) + u_{drift} S_{drift}, \quad (13)$$

$$\Phi_R = \beta_M \frac{M_{tot}}{M_{cap}} + \beta_B \frac{\gamma S_{model}}{B \tau_{sync}} + \beta_\eta (1 - \eta). \quad (14)$$

Интегральный функционал потерь записывается как

$$J(x; \theta) = w_P \Phi_P + w_A \Phi_A + w_R \Phi_R, w_P + w_A + w_R = 1, \quad (15)$$

и служит единым критерием, обеспечивающим совместную оптимизацию быстродействия, точности и ресурсных затрат, тем самым впервые реализована интеграция производительности, качества и ресурсоэффективности в рамках одного функционала потерь в ИС на основе ФРВ.

Область допустимых решений и отказоустойчивость. Допустимое множество D задаётся системой неравенств: устойчивость $\lambda_{in} < \mu$, соблюдение требований $\tau_{resp} \leq \tau_{SLA}$, требуемая пропускная способность $Q_{proc} \geq Q_{req}$, бюджет памяти $0 < M_{tot} \leq M_{cap}$, коммуникационная квота $\gamma S_{model} / (B \tau_{sync}) \leq \xi_{net} \in (0,1)$, эффективность $\eta \geq \eta_{min}$, а также внешнее ограничение по доступности $R \geq R_{min}$. Показатель R связан с архитектурными механизмами обеспечения надежности и отказоустойчивости в рассматриваемых системах.

Характер задачи и метод решения. Из-за операций \min/\max и дробно-рациональных выражений задача невыпукла и при дискретной переменной N — смешанно-целочисленна. Практически применима координированная оптимизация по $\{\tau_{sync}, M_{tot}\}$ при фиксированном N и переборе γ в допустимом диапазоне.

В представленной форме модель компактно объединяет ключевые зависимости и критерии, обеспечивая аналитическую основу для настройки параметров $\{\tau_{sync}, \gamma, M_{tot}, N\}$ под конкретные условия θ при гарантии отказоустойчивости.

В соответствии с вышеизложенной моделью повышения качества обработки потоковых данных далее представлены методы, разработанные для практической реализации её положений — обеспечения баланса между производительностью, точностью и ресурсоэффективностью при обработке потоковых данных в федеративной архитектуре.

Метод адаптивной федеративной обработки потоковых данных (ФОПД)

Цель метода заключается в организации многоуровневой (локально и глобально распределенной) обработки потоковых данных в федеративной архитектуре при минимизации интегрального функционала потерь на каждом окне времени с учётом требуемой ресурсоёмкости и отказоустойчивости.

Управляемые параметры, используемые в методе: период синхронизации τ_{sync} , степень компрессии обновлений $\gamma \in (0,1]$, коэффициент забывания $\rho \in (0,1)$. Ключевые метрики: доля коммуникаций δ_{sync} , масштабируемость K_{scal} , интенсивность обслуживания μ , задержка τ_{resp} , пропускная способность Q_{proc} , «возраст» модели Δ , точность Acc , отказоустойчивость R .

Метод описывается последовательным процессом, включающим следующие шаги.

Шаг 1. Инициализация. Задаются требования (τ_{SLA}, Q_{req}) , коммуникационная квота $\xi_{net} \in (0,1)$, полоса B , память M_{tot} , число исполнителей N . Выбираются стартовые параметры

$$u_0 = (\tau_{sync}^{(0)}, \gamma^{(0)}, \rho^{(0)}) \in [\tau_{min}, \tau_{max}] \times [\gamma_{min}, \gamma_{max}] \times (0,1). \quad (16)$$

Шаг 2. Окно адаптации. На окне $[t_k, t_{k+1})$ длительности T_k вычисляется входная интенсивность $\lambda_{in,k}$ и показатель дрейфа $S_{drift,k}$.

Шаг 3. Оценка системных метрик (квазистационарная M/M/1-аппроксимация). Определяется устойчивость очереди μ_k , средняя задержка отклика $\tau_{resp,k}$, пропускная способность (в стационарном потоке) $Q_{proc,k}$, время коммуникации одного раунда обработки данных на окне $t_{comm,k}$ и доля коммуникаций $\delta_{sync,k}$:

$$\mu_k > \lambda_{in,k}, \quad (17)$$

$$\tau_{resp,k} \approx \frac{1}{\mu_k - \lambda_{in,k}}, \quad (18)$$

$$Q_{proc,k} \approx \min\{\mu_k, \lambda_{in,k}\}, \quad (19)$$

$$t_{comm,k} = \frac{\gamma_k S_{model}}{B}, \quad (20)$$

$$\delta_{sync,k} = \frac{t_{comm,k}}{\tau_{sync,k}}. \quad (21)$$

Шаг 4. Постановка задачи на окне имеет вид

$$\min_{u_k} J_k = \quad (22)$$

$$w_P \Phi_P(\tau_{resp,k}, Q_{proc,k}, K_{scal,k}) + w_A \Phi_A(Acc_k, S_{drift,k}, C_{cons}, A_{stab}) + w_R \Phi_R(M_{tot}, B, \tau_{sync,k}, \eta)$$

при ограничениях связи и отказоустойчивости:

$$\frac{\gamma_k S_{model}}{B \tau_{sync,k}} \leq \xi_{net}, \quad (23)$$

$$R(u_k; G) \geq R_{min}, R = \Psi(G, \{w(e)\}, \tau_{sync}, \gamma). \quad (24)$$

Шаг 5. Градиентная коррекция частоты синхронизации (проеекционный шаг).

$$\tau_{sync,k+1} = \Pi_{[\tau_{min}, \tau_{max}]} (\tau_{sync,k} - \alpha_k [w_P \partial_\tau \Phi_P + w_A \partial_\tau \Phi_A + w_R \partial_\tau \Phi_R]). \quad (25)$$

Знаки производных:

$$\frac{\partial K_{scal,k}}{\partial \tau_{sync,k}} = \frac{S(N) t_{comm,k}}{N \tau_{sync,k}^2} > 0, \quad (26)$$

$$\frac{\partial \tau_{resp,k}}{\partial \tau_{sync,k}} < 0, \frac{\partial Acc_k}{\partial \tau_{sync,k}} \approx -\kappa_d S_{drift,k} < 0, \quad (27)$$

$$\frac{\partial \Phi_R}{\partial \tau_{sync}} \sim -\beta_B \frac{\gamma S_{model}}{B \tau_{sync}^2} < 0. \quad (28)$$

Вводится правило быстрого снижения частоты: при $\delta_{sync,k} \geq 1 \Rightarrow$ немедленно увеличить $\tau_{sync,k}$ так, чтобы $\delta_{sync,k} < 1$.

Шаг 6. Настройка компрессии обновлений. Соблюдается (23) и минимизируется J_k :

$$\gamma_{k+1} = \Pi_{[\gamma_{min}, \gamma_{max}]} (\gamma_k - \beta_k [w_A \partial_\gamma \Phi_A + w_R \partial_\gamma \Phi_R]), \quad (29)$$

$$\frac{\partial \Phi_R}{\partial \gamma} \sim \beta_B \frac{S_{model}}{B \tau_{sync}} > 0. \quad (30)$$

Интерпретация: рост τ_{sync} и снижение γ уменьшают коммуникационные издержки; при деградации точности — обратные действия.

Шаг 7. Локальная адаптация забывания.

$$\rho_{k+1} = \Pi_{(0,1)} (\rho_k - \zeta_k \partial_\rho \Phi_A). \quad (31)$$

Повышение ρ ускоряет реакцию на дрейф S_{drift} , но увеличивает дисперсии локальных оценок.

Шаг 8. Федеративная агрегация. На конце окна агрегируются параметры/достаточные статистики с весами по «возрасту» и дрейфу:

$$\theta_{k+1}^{global} = \bigoplus_{i \in \mathcal{V}} \omega_{i,k} \theta_k^{(i)}, \omega_{i,k} \sim \exp(-\beta_d S_{drift,k} \Delta_{i,k}). \quad (32)$$

Агрегация не зависит от порядка поступления обновлений (устойчивость к асинхронности).

Шаг 9. Контроль отказоустойчивости и архитектурная коррекция. Если $\hat{R} < R_{min}$, выполняется жесткая проекция $u_k \mapsto u'_k$ и архитектурные меры: децентрализация (несколько агрегаторов), асинхронная консолидация,

резервирование каналов и кворум приёма; временно увеличиваются τ_{sync} и γ для разгрузки сети и восстановления согласованности.

Шаг 10. Завершение окна и переход. Вычисляется J_k по (22), проверяется соблюдение условий требований к системе. При нарушении — дополнительная коррекция $u_k \rightarrow u'_k$. Фиксируются $\{u_{k+1}\}$ и метрики $\{\Phi_P, \Phi_A, \Phi_R\}$ для следующего окна.

Таким образом, метод представляет собой итеративное управление параметрами $(\tau_{\text{sync}}, \gamma, \rho)$ на скользящем окне с проекционными шагами, обеспечивающее баланс между тремя факторами:

— *производительность/масштабируемость*: увеличение τ_{sync} снижает коммуникационную нагрузку и задержки (26)-(28), улучшая Φ_R и K_{scal} .

— *точность при дрейфе*: слишком редкая синхронизация ($\tau_{\text{sync}} \uparrow$) и сильная компрессия ($\gamma \downarrow$) ухудшают Acc (26)-(29); компенсируется ростом ρ .

— *отказоустойчивость*: обеспечивается ограничением $R \geq R_{\text{min}}$ (24) и архитектурными мерами (децентрализация, кворумы, резервирование).

Допущения и предпосылки метода. Метод основывается на следующих предпосылках: квазистационарность потока данных на интервале окна адаптации; аппроксимация задержек обслуживания моделью M/M/1 (17)-(21); линейная аппроксимация деградации точности по «возрасту» модели и скорости концептуального дрейфа; использование внешней (неаналитической) модели отказоустойчивости $R = \Psi(\cdot)$.

На практике метод обеспечивает устойчивое соблюдение коммуникационных квот (23), (24) и требований к системе за счёт автоматической подстройки частоты синхронизации, уровня компрессии и темпа забывания статистик моделей МО.

Метод комбинированного обучения в федеративной среде (КОФС)

Цель метода – устойчивое обучение моделей МО при наличии концептуального дрейфа, ограничений на коммуникационные и вычислительные ресурсы, а также требований к качеству обслуживания в ИС на основе ФРВ. Метод КОФС логически продолжает метод ФОПД, обеспечивает согласованное чередование локальных потоковых обновлений и периодических федеративных синхронизаций [18].

Оптимизация выполняется по параметрам режима обучения в рамках интегрального критерия качества $J(x; \theta)$ при системных ограничениях производительности и надёжности:

$$\lambda_{\text{in}} < \mu(x; \theta), \tau_{\text{resp}} \leq \tau_{\text{SLA}}, R \geq R_{\text{min}}. \quad (33)$$

Управляемая переменная: доля федеративных итераций на окне $p_{\text{fed}} \in [0,1]$ (доля локальных итераций $1 - p_{\text{fed}}$). Целевая функция: взвешенная сумма компонент точности, производительности и ресурсоэффективности при (33):

$$\min_{p_{\text{fed}} \in [0,1]} w_A \Phi_A(\text{Acc}) + w_P \Phi_P + w_R \Phi_R \quad (34)$$

На окне $[t_k, t_{k+1})$ длиной T_k :

$$\text{Acc}_k \approx \text{Acc}_{\text{loc}} (1 - p_{\text{fed},k}) + \text{Acc}_{\text{fed}}(p_{\text{fed},k}; \Delta_k, S_{\text{drift}}), \quad (35)$$

где возраст (время задержки обновления между локальными и глобальной копиями модели) глобальной модели $\Delta_k = \tau_{\text{sync},k} + t_{\text{comm},k}$, а

$$t_{\text{comm},k} = \tau_{\text{agg}} + \frac{\gamma_k S_{\text{model}}}{B}. \quad (36)$$

Для калибровки используется насыщаемая аппроксимация

$$\text{Acc}_{\text{fed}}(p_{\text{fed}}; \Delta, S_{\text{drift}}) = \text{Acc}_{\text{loc}}^* + \alpha(1 - e^{-\kappa p_{\text{fed}}}) - \kappa_d S_{\text{drift}} \Delta, \alpha, \kappa, \kappa_d > 0. \quad (37)$$

Компонента ресурсов функционала потерь линеаризуется по числу федеративных раундов (циклов обмена между узлами и агрегатором, включающих рассылку глобальной модели, локальное обучение на узлах и последующую агрегацию обновлений):

$$\Phi_R^{(\text{comm})}(p_{\text{fed}}; \tau_{\text{sync}}, \gamma) \approx \chi p_{\text{fed}}, \chi = \frac{\gamma S_{\text{model}}}{B \tau_{\text{sync}}}. \quad (38)$$

Аналитическая настройка p_{fed} . При фиксированных $\tau_{\text{sync}}, \gamma$:

$$p_{\text{fed}}^* = \frac{1}{\kappa} \ln \left(\frac{\alpha \kappa w_A u_{\text{Acc}}}{w_R u_{\text{comm}}} \cdot \frac{1}{\chi} \right), p_{\text{fed}}^* \in [0,1], \quad (39)$$

при условии внутреннего оптимума

$$0 < \frac{w_R u_{\text{comm}}}{w_A u_{\text{Acc}}} \chi < \alpha \kappa. \quad (40)$$

Иначе решение — на границах $p_{\text{fed}} \in \{0,1\}$.

Для динамической корректировки параметров метода используется проекционный шаг:

$$p_{\text{fed},k+1} = \Pi_{[0,1]}(p_{\text{fed},k} - \eta_k \partial_{p_{\text{fed}}} [w_A \Phi_A + w_R \Phi_R]), \eta_k > 0, \quad (41)$$

либо пороговая (триггерная) схема адаптации параметра [19]:

$$p_{\text{fed},k} = \text{clip}(\alpha_d S_{\text{drift},k} + \alpha_c (1 - C_{\text{cons},k}) + \alpha_s (1 - A_{\text{stab},k}), 0, 1). \quad (42)$$

Ограничение $\chi p_{\text{fed}} \leq \xi_{\text{net}} \in (0, 1)$ реализуется через

$$s = \max\left\{1, \frac{\chi p_{\text{fed}}}{\xi_{\text{net}}}\right\}, \alpha \in [0, 1], \quad (43)$$

$$p_{\text{fed}} \leftarrow \Pi_{[0,1]}\left(\frac{p_{\text{fed}}}{s}\right), \tau_{\text{sync}} \leftarrow \tau_{\text{sync}} s^{1-\alpha}, \gamma \leftarrow \gamma s^{-\alpha}.$$

Весовая агрегация локальных статистик подавляет устаревшие обновления:

$$\omega_i \sim \exp(-\beta_d S_{\text{drift},i} \Delta_i), \sum_i \omega_i = 1, \beta_d > 0. \quad (44)$$

Совместная настройка $\{p_{\text{fed}}, \tau_{\text{sync}}, \gamma\}$ поддерживает компромисс между требованиями к системе с учётом влияния на $\delta_{\text{sync}}, K_{\text{scal}}$ и τ_{resp} .

Предлагаемый метод КОФС описывается последовательным процессом, включающим следующие шаги.

Шаг 1. Инициализация. Устанавливаются целевые уровни $Acc_{\text{target}}, \tau_{\text{SLA}}$, ограничения по сетевым ресурсам $\xi_{\text{net}} \in (0, 1)$, допустимые диапазоны $\tau_{\text{sync}} \in [\tau_{\text{min}}, \tau_{\text{max}}]$, $\gamma \in [\gamma_{\text{min}}, \gamma_{\text{max}}]$. Выбирается начальное значение $p_{\text{fed}}^{(0)}$ по истории S_{drift} ; задаются веса w_A, w_R, w_P и нормировки $u_{\text{Acc}}, u_{\text{comm}}$.

Шаг 2. Оценка состояния потока. На скользящем окне вычисляются $\lambda_{\text{in}}, S_{\text{drift}}, C_{\text{cons}}, A_{\text{stab}}$, возраст Δ ; обновляются оценки $\mu(x; \theta), \tau_{\text{resp}}, Q_{\text{proc}}$.

Шаг 3. Калибровка доли федеративных итераций. Выполняется один шаг (41) либо расчёт по триггерному правилу (42). Проверяется соблюдение сетевого бюджета $\chi p_{\text{fed}} \leq \xi_{\text{net}}$; при нарушении применяется корректировка согласно (43).

Шаг 4. Локальные итерации. В доле $1 - p_{\text{fed}}$ выполняются потоковые обновления локальных моделей с коэффициентом забывания $\rho(t)$ и регуляризацией, стабилизирующей параметры при дрейфе.

Шаг 5. Федеративные раунды. В доле p_{fed} проводится согласование: формируются достаточные статистики, выполняется агрегация с весами (28), (44). Параметр γ подстраивается под ξ_{net} и целевой Acc .

Шаг 6. Адаптивная коррекция параметров. По завершении окна пересчитываются $\{Acc, \tau_{\text{resp}}, Q_{\text{proc}}\}$ и обновлялись $\{p_{\text{fed}}, \tau_{\text{sync}}, \gamma\}$ путём минимизации $w_A \Phi_A + w_R \Phi_R$ при сохранении Φ_P в пределах требований. При снижении $R < R_{\text{min}}$ временно уменьшается доля федеративных итераций ($p_{\text{fed}} \downarrow$) с последующей консолидацией.

Шаг 7. Переход к следующему окну. Регистрируются значения Φ_P, Φ_A, Φ_R и обновлённые параметры; формируется следующая итерация метода с учётом трендов $\{S_{\text{drift}}, \Delta\}$.

Допущения и границы применимости метода. Рассматривается квазистационарный поток в пределах окна. Зависимость точности от p_{fed} аппроксимируется монотонной насыщаемой функцией (37). Коммуникационная стоимость аппроксимирована линейной функцией от числа раундов (38). Влияние p_{fed} на Φ_P учитывается через δ_{sync} и K_{scal} без построения полной очередной модели. Калибровка α, κ, κ_d требует накопления данных о дрейфе и динамике согласований.

Предложенный метод КОФС обеспечивает адаптивный компромисс между точностью и коммуникационными затратами при соблюдении ограничений производительности и надёжности. Предлагаемая оконная модель позволяет учесть возраст глобальной модели и воздействие дрейфа на точность, а аналитическое выражение (39) задает интерпретируемую настройку доли федеративных итераций. Триггерные и градиентные схемы управления обеспечивают автоматическое повышение частоты синхронизаций при нарастании концептуального дрейфа и уменьшение при ограничениях сети, что поддерживает устойчивость обучения и отказоустойчивость всей системы. Совместная настройка $\{p_{\text{fed}}, \tau_{\text{sync}}, \gamma\}$ формирует внутренний оптимум по критерию $w_A \Phi_A + w_R \Phi_R + w_P \Phi_P$ и обеспечивает достижение целевых уровней качества при реальных ресурсных ограничениях.

Метод преобразования признаков потоковых данных (ПППД)

Для снижения чувствительности к нестационарности входных потоков данных и обеспечения согласованности статистик между узлами вводится метод преобразования признаков потоковых данных. Метод предназначен для стабилизации входного признакового пространства потоковой ИС и снижения чувствительности к нестационарности и шумам при ограничениях по памяти, пропускной способности и задержке отклика [20]. Вклад метода учитывается в интегральном функционале потерь через одновременное: уменьшение деградации точности Acc из-за возраста модели МО Δ при дрейфе S_{drift} ; повышение эффективной производительности μ ; соблюдение ресурсоёмкости Φ_R под ограничениями M_{tot} , канала B , целевой

эффективности η , масштабируемости K_{scal} и требуемой отказоустойчивости $R \geq R_{\text{min}}$. Параметры синхронизации τ_{sync} и γ влияют на «возраст» Δ .

Пусть поток наблюдений $x_t \in \mathbb{R}^d$ поступает с интенсивностью λ_{in} . Функция предобработки задаётся отображением

$$z_t = f_{\text{pre}}(x_t; \Theta_t) \quad (45)$$

где Θ_t — набор поточно обновляемых параметров (оценки моментов, матрицы декорреляции, параметры редукции).

Экспоненциально-взвешенное нормирование (z-score) с адаптивным забыванием задается:

$$\mu_t = (1 - \rho_t)\mu_{t-1} + \rho_t x_t, \Sigma_t = (1 - \rho_t)\Sigma_{t-1} + \rho_t (x_t - \mu_{t-1})(x_t - \mu_{t-1})^\top, \quad (46)$$

$$x_t^{(n)} = \text{diag}(\sigma_t + \varepsilon)^{-1}(x_t - \mu_t), \sigma_t = \sqrt{\text{diag}\Sigma_t}, \varepsilon > 0. \quad (47)$$

Коэффициент забывания задается:

$$\rho_t = 1 - \exp(-\beta_d S_{\text{drift}} \Delta_t) \in (0,1), \quad (48)$$

где Δ_t — возраст глобального состояния (например, $\Delta_t = \tau_{\text{sync}} + t_{\text{comm}}$).

Формула (46) сохраняет симметричность и положительную полуопределённость Σ_t ; для численной устойчивости допускается симметризация $\Sigma_t \leftarrow \frac{1}{2}(\Sigma_t + \Sigma_t^\top)$.

Декорреляция с блочной аппроксимацией выражается как:

$$z_t = W_t x_t^{(n)}, W_t = (\Sigma_t^{(n)} + \delta I)^{-\frac{1}{2}} \approx \text{blkdiag}\{(\Sigma_{t,[b]}^{(n)} + \delta I)^{-\frac{1}{2}}\}. \quad (49)$$

Здесь $\Sigma_t^{(n)}$ — ковариация нормированных признаков, $\delta > 0$ — параметр регуляризации, $[b]$ — блочная структура, выбираемая с учётом M_{tot} и η . Для тяжёлых хвостов допускается замена (μ_t, σ_t) на робастные $(\text{med}_t, \text{MAD}_t)$ покомпонентно.

Редукция размерности выражается как:

$$u_t = R_t z_t, \|R_t\|_2 \leq \kappa_R, R_t \in \mathbb{R}^{d' \times d}, d' \ll d. \quad (50)$$

Оператор R_t реализует редукцию размерности признакового пространства: он либо поддерживается поточно в виде инкрементального разложения главных компонент (Principal Component Analysis, PCA) [21], либо фиксируется как субгауссов случайный проектор, обеспечивающий сохранение межточечных расстояний в смысле леммы Джонсона–Линденштраусса [22].

Итоговая трансформация имеет вид:

$$f_{\text{pre}}(x_t; \Theta_t) = u_t = R_t (\Sigma_t^{(n)} + \delta I)^{-\frac{1}{2}} \text{diag}(\sigma_t + \varepsilon)^{-1}(x_t - \mu_t). \quad (51)$$

Аналитические свойства метода и вклад в функционал потерь. Стабилизация точности – для классификатора с липшицевой константой L_f

$$|\Delta \text{Acc}| \lesssim L_f \|\Delta u\|_2 \leq L_f \|R_t\|_2 \|W_t\|_2 \|\text{diag}(\sigma_t + \varepsilon)^{-1}\|_2 \|\Delta x\|_2, \quad (52)$$

что демонстрирует снижение чувствительности к дрейфу входа Δx благодаря нормировке и декорреляции (операции выбеливания признаков) [23].

Ускорение обработки одного элемента задается:

$$\mu_1^{(\text{pre})} = \mu_1 S_{\text{dim}} S_{\text{impl}}, S_{\text{dim}} = \frac{d}{d'} > 1, 0 < S_{\text{impl}} \leq 1, \quad (53)$$

что уменьшает задержку ответа $\tau_{\text{resp}} = 1/(\mu - \lambda_{\text{in}})$ и повышает пропускную способность Q_{proc} .

Потребление памяти и вычисления определяются размерами $\Theta_t = \{\mu_t, \sigma_t^2, \Sigma_{t,[b]}^{(n)}, W_{t,[b]}, R_t\}$, периодичностью пересчёта W_t и режимом поддержания R_t под ограничениями M_{tot} и η .

Предлагаемый метод описывается последовательным процессом, включающим следующие шаги.

Шаг 1. Инициализация статистик и ограничений. Задаются μ_0 , диагональная Σ_0 , параметры $\rho_0 \in (0,1)$, $\delta > 0$, $\varepsilon > 0$; определяется блочная структура для декорреляции (размеры блоков $\{b_k\}$). Выбирается целевая размерность d' и ресурсы M_{tot} , η для согласования допустимых $\{b_k\}$ и частоты обновления W_t . Фиксируются пределы κ_R и режим R_t (инкрементальный или фиксированный).

Шаг 2. Поточная оценка дрейфа и адаптивного забывания. Оценивается концептуальный дрейф

$$S_{\text{drift}}(t) = \|\mu_t - \mu_{t-1}\|_2 + \|\Sigma_t - \Sigma_{t-1}\|_F \quad (54)$$

на скользящем окне; вычисляется «возраст» $\Delta_t = \tau_{\text{sync}} + t_{\text{comm}}$. Устанавливается $\rho_t = 1 - \exp(-\beta_d S_{\text{drift}} \Delta_t)$.

Выполняется обновление μ_t, Σ_t по (46) с симметризацией при необходимости.

Шаг 3. Нормализация и робастное масштабирование. Формируются нормированные признаки $x_t^{(n)}$ по (47).

При наличии тяжёлых хвостов или выбросов выполняется покомпонентная замена (μ_t, σ_t) на $(\text{med}_t, \text{MAD}_t)$ или смешанная схема (например, по перцентильному критерию).

Шаг 4. Декорреляция и блочное выбеливание. Оценивается $\Sigma_t^{(n)}$ на блоках $[b_k]$. Для каждого блока вычисляется $(\Sigma_{t,[b_k]}^{(n)} + \delta I)^{-1/2}$ (одним из численно устойчивых методов — собственным разложением, разложением Чолески или итерацией Ньютона–Шульца [24, 25]), формируется W_t . Частота пересчёта матриц согласуется с общим объёмом памяти M_{tot} и η , допускается режим «замороженных» матриц с периодическим обновлением для снижения вычислительных затрат [26].

Шаг 5. Редукция размерности. Применяется R_t к $z_t = W_t x_t^{(n)}$. В инкрементальном варианте обновляется подпространство главных направлений (например, по правилу Ойи на окне); в варианте фиксированного проектора используется субгауссов проектор с контролем $\|R_t\|_2 \leq \kappa_R$. Получается $u_t \in \mathbb{R}^{d'}$.

Шаг 6. Федеративное выравнивание параметров предобработки. Для уменьшения межузловой вариативности выполняется согласование $\{\mu_t, \sigma_t^2\}$ и при необходимости блоковых статистик через ассоциативную операцию (28). Рекомендуется взвешивание по эффективным объёмам w_i :

$$\bar{\mu} = \frac{\sum_i w_i \mu_i}{\sum_i w_i}, \bar{\sigma}^2 = \frac{\sum_i w_i \sigma_i^2}{\sum_i w_i}. \quad (55)$$

Частота согласования подбирается с учётом τ_{sync} , γ и ограничений канала B при обеспечении $R \geq R_{\text{min}}$.

Шаг 7. Контроль SLA и адаптация параметров. Периодически оцениваются τ_{resp} , Q_{proc} , Acc на валидационном подмножестве потока. При превышении τ_{resp} или падении Acc корректируются d' , размеры блоков $\{b_k\}$, частота пересчёта W_t , а также ρ_t (например, увеличение d' при потере точности и уменьшение d' при росте задержек).

Допущения метода. На интервале окна поток квазистационарен, поэтому допускается использование экспоненциально-взвешенных оценок моментов (μ_t, Σ_t) и периодическое обновление преобразований; линейная предобработка (нормировка, блочное выбеливание, редукция размерности) достаточна для стабилизации первых двух моментов и снижения чувствительности точности к дрейфу входа; блочное «выбеливание» аппроксимирует полную ортогонализацию с контролируемой погрешностью при заданном ограничении памяти, регуляризация $\delta > 0$ обеспечивает численную устойчивость (положительную определённость), а $\varepsilon > 0$ — невырожденность масштаба; оператор редукции R_t (инкрементальная PCA либо субгауссов проектор с $\|R_t\|_2 \leq \kappa_R$) сохраняет метрику/долю дисперсии в требуемых пределах при $d' \ll d$; частоты пересчёта W_t и (при необходимости) R_t выбираются так, чтобы затраты укладывались в ресурсные ограничения (M_{tot}, B) без нарушения требований по τ_{resp} и $R \geq R_{\text{min}}$; S_{drift} и Δ наблюдаемы в потоковом режиме и используются для адаптации коэффициента забывания.

Алгоритм адаптивной федеративной агрегации статистических структур (АФАС)

Согласованные признаки и локальные дельты параметров агрегируются алгоритмом ассоциативной федеративной агрегации статистических структур, обеспечивающим порядконезависимую свёртку с весами свежести при асинхронной доставке и в рамках заданного коммуникационного бюджета. Цель алгоритма — обеспечить согласованную и устойчивую агрегацию локальных достаточных статистик и приращений параметров в ИС на основе ФРВ при асинхронной обработке, задержках и концептуальном дрейфе данных, соблюдая ограничения коммуникационного бюджета и надёжности. Алгоритм формирует корректные глобальные оценки $(\mu, \Sigma, \hat{p}_{\text{out}})$ и агрегированные обновления θ^{new} , сохраняет ассоциативность и порядконезависимость свёртки, подавляет устаревшие/смещённые обновления по актуальности (свежести) [27] и минимизирует коммуникационные издержки (за счёт $\gamma, \tau_{\text{sync}}$) без потери требуемой точности и производительности (по τ_{resp} , $R \geq R_{\text{min}}$). На узле i в течение одного интервала агрегации поддерживаются локальные достаточные статистики

$$S_i = \{n_i, \mu_i \in \mathbb{R}^d, \Sigma_i \in \mathbb{R}^{d \times d}, p_i^{(\text{out})}\}, \quad (56)$$

где n_i — эффективный объём (с учётом локального забывания ρ), μ_i и Σ_i — оценки среднего и ковариации (диагональная или блочная форма допускается ограничением бюджета M_{tot}), $p_i^{(\text{out})}$ — доля выходов за доверительный интервал.

Коммуникационные параметры задаются величинами:

$$t_{\text{comm}}^{(i)} = \tau_{\text{agg}} + \frac{\gamma S_{\text{model}}^{(i)}}{B}, \Delta_i = \tau_{\text{sync}} + t_{\text{comm}}^{(i)}, \quad (57)$$

где B — полоса пропускания канала, $\gamma \in (0, 1]$ — коэффициент компрессии полезной нагрузки, τ_{agg} —

фиксированная накладная задержка агрегационного стека, τ_{sync} — период синхронизации.

Свежесть узла учитывается весовой функцией (с триммингом) [28]:

$$\omega_i = \text{clip}(\exp(-\beta_d S_{\text{drift}} \Delta_i), \omega_{\min}, \omega_{\max}), 0 < \omega_{\min} \leq \omega_i \leq \omega_{\max} \leq 1, \quad (58)$$

где S_{drift} — индикатор/скорость дрейфа, $\beta_d > 0$ — коэффициент чувствительности.

Агрегирование выполняется в пространстве сырых моментов [29]:

$$\begin{aligned} M_0 &= \sum_i \omega_i n_i, M_1 = \sum_i \omega_i n_i \mu_i, \\ M_2 &= \sum_i \omega_i n_i (\Sigma_i + \mu_i \mu_i^\top), P_{\text{out}} = \sum_i \omega_i n_i p_i^{(\text{out})}. \end{aligned} \quad (59)$$

Глобальные оценки определяются как

$$\mu = \frac{M_1}{M_0}, \Sigma = \frac{M_2}{M_0} - \mu \mu^\top, \hat{p}_{\text{out}} = \frac{P_{\text{out}}}{M_0}. \quad (60)$$

Для параметров модели θ используется взвешенное усреднение приращений:

$$\theta^{\text{new}} = \theta^{\text{old}} + \frac{\sum_i \omega_i n_i \Delta \theta_i}{\sum_i \omega_i n_i}, \quad (61)$$

где $\Delta \theta_i$ допускает сжатие с коэффициентом γ .

Коммуникационный бюджет раунда контролируется показателем

$$\chi = \frac{\gamma S_{\text{model}}}{B \tau_{\text{sync}}} \leq \xi_{\text{net}} \in (0, 1), \quad (62)$$

который ограничивает долю времени/пропускной способности, расходуемую на обмен.

Операция свёртки и её свойства. Определяется агрегат $A = \{M_0^A, M_1^A, M_2^A, P_{\text{out}}^A, U^A\}$, где U — аккумулятор для $\sum_i \omega_i n_i \Delta \theta_i$. Для двух агрегатов A и B вводится бинарная операция

$$A \oplus B = \{M_0^A + M_0^B, M_1^A + M_1^B, M_2^A + M_2^B, P_{\text{out}}^A + P_{\text{out}}^B, U^A + U^B\}. \quad (63)$$

Свойства алгоритма: ассоциативность и коммутативность следуют из покомпонентного сложения; порядконезависимость обеспечивается ассоциативностью. идемпотентность достигается на практике посредством журнала идентификаторов обновлений и устранения повторов на агрегаторе (дедупликация до применения \oplus).

Корректность алгоритма. Представленные оценки $(\mu, \Sigma, \hat{p}_{\text{out}})$ эквивалентны взвешенному закону полной дисперсии и совпадают с оценками по «склеенной» выборке с весами $\{\omega_i n_i\}$.

Предлагаемый алгоритм описывается последовательным процессом, включающим следующие шаги.

Шаг 1. Инициализация агрегатора. Устанавливаются параметры $\tau_{\text{sync}}, \gamma, B, \beta_d, S_{\text{drift}}, \tau_{\text{agg}}, \xi_{\text{net}}, \omega_{\min}, \omega_{\max},$

Δ_{max} . Выбирается представление Σ_i (полное, диагональное или блочное) в соответствии с M_{tot} .

Инициализируются аккумуляторы:

$$M_0 \leftarrow 0, M_1 \leftarrow 0, M_2 \leftarrow 0, P_{\text{out}} \leftarrow 0, U \leftarrow 0, \text{Log} \leftarrow \emptyset. \quad (64)$$

Шаг 2. Приём и валидация пакетов. При поступлении пакета S_i с уникальным идентификатором выполняется проверка на дубликат (Log), целостность и допустимость метаданных.

Шаг 3. Оценка задержки и возраста. Для узла i по (57) вычисляются $t_{\text{comm}}^{(i)}$ и Δ_i . При $\Delta_i > \Delta_{\text{max}}$ применяется политика: понижение веса или отбрасывание пакета.

Шаг 4. Расчёт весов и тримминг. Определяется $\omega_i = \exp(-\beta_d S_{\text{drift}} \Delta_i)$ и обрезается по порогу в интервал $[\omega_{\min}, \omega_{\max}]$. Выполняется дедупликация по журналу идентификаторов.

Шаг 5. Обновление интегральных моментов. Аккумуляторы обновляются рекуррентно:

$$\begin{aligned} M_0 &\leftarrow M_0 + \omega_i n_i, \\ M_1 &\leftarrow M_1 + \omega_i n_i \mu_i, \\ M_2 &\leftarrow M_2 + \omega_i n_i (\Sigma_i + \mu_i \mu_i^\top), \\ P_{\text{out}} &\leftarrow P_{\text{out}} + \omega_i n_i p_i^{(\text{out})}, \\ U &\leftarrow U + \omega_i n_i \Delta \theta_i. \end{aligned} \quad (65)$$

Идентификатор добавляется в журнал: $\text{Log} \leftarrow \text{Log} \cup \{id_i\}$.

Шаг 6. Условие завершения раунда. Раунд завершается по таймеру τ_{sync} либо при достижении кворума частично доставленных обновлений. Допускается завершение с неполной выборкой.

Шаг 7. Финализация глобальных оценок. Если $M_0 > 0$, вычисляются (с проекцией на PSD при необходимости):

$$\mu = \frac{M_1}{M_0}, \Sigma = \frac{M_2}{M_0} - \mu \mu^\top, \hat{p}_{\text{out}} = \frac{P_{\text{out}}}{M_0}, \quad (66)$$

$$\Delta\theta_{agg} = \frac{U}{M_0}, \theta^{new} = \theta^{old} + \Delta\theta_{agg}. \quad (67)$$

При $M_0 = 0$ сохраняются предыдущие оценки.

Шаг 8. Контроль коммуникационных ограничений. Оценивается нагрузка $\chi = \gamma S_{model} / (B \tau_{sync})$. При $\chi > \xi_{net}$ параметр τ_{sync} увеличивается и/или γ уменьшается согласно принятой политике.

Шаг 9. Рассылка калибровок и перенос состояния. Узлам рассылаются $\{\mu, \Sigma\}$ и/или θ^{new} для калибровки локальных трансформаций. Состояние $\{M_0, M_1, M_2, P_{out}\}$ переносится на следующий интервал при активном режиме буферизации.

Шаг 10. Отказоустойчивость. При нарушении связности выполняется буферизация частичных S_i (с контролем M_{tot}) и последующая отложенная свёртка посредством \oplus , что сохраняет ассоциативность.

Псевдокод алгоритма АФАС представлен в Листинге 1.

Листинг 1 – Псевдокод алгоритма АФАС

```

Inputs (aggregator):
    tau_sync, gamma, B, beta_d, S_drift, tau_agg, xi_net, omega_min, omega_max,
Delta_max
State (aggregator):
    M0←0; M1←0_d; M2←0_{d×d}; Pout←0; U←0_d; Log←∅; theta_old←init()
# Приём пакета от узла i
Procedure OnPacket(S_i = {id_i, n_i, mu_i, Sigma_i, p_out_i, DeltaTheta_i,
S_model_i}):
    if id_i ∈ Log: return
    t_comm_i ← tau_agg + (gamma * S_model_i) / B
    Delta_i ← tau_sync + t_comm_i
    if Delta_i > Delta_max and policy_drop_stale(): return
    w_i ← clip( exp(-beta_d * S_drift * Delta_i), omega_min, omega_max )
    M0 ← M0 + w_i * n_i
    M1 ← M1 + w_i * n_i * mu_i
    M2 ← M2 + w_i * n_i * (Sigma_i + mu_i * mu_i^T)
    Pout ← Pout + w_i * n_i * p_out_i
    U ← U + w_i * n_i * DeltaTheta_i
    Log ← Log ∪ {id_i}
# Завершение раунда
Procedure FinalizeRound():
    if M0 = 0:
        mu←mu_prev; Sigma←Sigma_prev; p_hat_out←p_prev; DeltaTheta_agg←0_d;
theta_new←theta_old
    else:
        mu ← M1 / M0
        Sigma ← (M2 / M0) - mu * mu^T
        Sigma ← project_PSD(Sigma)
        p_hat_out ← Pout / M0
        DeltaTheta_agg ← U / M0
        theta_new ← theta_old + DeltaTheta_agg
    chi ← (gamma * S_model_ref()) / (B * tau_sync)
    if chi > xi_net: (tau_sync, gamma) ← adjust_comm_budget(tau_sync, gamma, chi,
xi_net)
    Broadcast({mu, Sigma} and/or theta_new)
    Persist({M0, M1, M2, Pout}) # опционально для буферизации
    Reset() # начало нового раунда
Procedure Reset():
    M0←0; M1←0_d; M2←0_{d×d}; Pout←0; U←0_d; Log←∅; theta_old←theta_new
# Клиентский узел i: формирование пакета
Procedure BuildPacket():
    n_i ← effective_count_with_forgetting(rho)
    mu_i ← local_mean()
    Sigma_i ← local_covariance(repr = {diag|block|full})
    p_out_i ← out_of_bound_fraction()
    DeltaTheta_i ← compress( model_update_increment(theta_new), gamma )
    S_model_i ← payload_size_of(DeltaTheta_i, mu_i, Sigma_i, p_out_i)

```

```

    Send( {id_i, n_i, mu_i, Sigma_i, p_out_i, DeltaTheta_i, S_model_i} )
# Свертка агрегатов (шардирование/отложенная агрегация)
Function FOLD_OPLUS(A, B):
    return { M0:A.M0+B.M0, M1:A.M1+B.M1, M2:A.M2+B.M2, Pout:A.Pout+B.Pout,
U:A.U+B.U }
# Вспомогательные:
# clip(x, a, b)=min(max(x, a), b); project_PSD — проекция ковариации на PSD;
# adjust_comm_budget — политика изменения tau_sync/gamma при chi>xi_net.

```

Оценка времени на раунд при полном хранении ковариации составляет $O(m d^2)$, где m — число принятых пакетов за раунд, d — размерность признакового пространства. Для диагонального/блочного представления: $O(m d)$ или $O(m d b)$ соответственно (b — средний размер блока), что согласуется с ограничением M_{tot} и эффективностью η . Коммуникационная нагрузка клиента определяется $\gamma S_{\text{model}}^{(i)}$; на агрегаторе — суммой по узлам. Параметры τ_{sync} и γ подбираются с учётом компромисса между точностью и ресурсами и условия $\chi \leq \xi_{\text{net}}$.

Для обеспечения корректности вычислений и устойчивости при асинхронной доставке применяются следующие меры: проекция Σ на множество полуположительно определённых матриц (PSD) устраняет отрицательные собственные значения, возникающие из-за округлений; регуляризация деления при малых M_0 ($M_0 \leftarrow M_0 + \varepsilon$, $\varepsilon \ll 1$) предотвращает числовую нестабильность; политика снижения веса для устаревших пакетов ($\Delta_i > \Delta_{\text{max}}$) снижает дисперсию по сравнению с жёстким отбрасыванием. Эти приёмы обеспечивают устойчивость и воспроизводимость агрегирования в реальных вычислительных условиях.

Алгоритм инкрементального дерева решений (АИДР)

На уровне принятия решений используется инкрементальное дерево, где пороги задаются по неравенству Чебышёва и адаптируются по доле «внутренних» точек, что стабилизирует классификацию при мультимодальности и дрейфе без полной перестройки структуры. Цель алгоритма заключается в обеспечении динамической классификации потоковых данных с адаптацией к нестационарности и мультимодальности распределений при ограничениях на вычисления, память и коммуникации [30]. Оптимизация выполняется относительно заданного функционала потерь при ресурсных ограничениях.

Рассматривается дерево $T = (V, E)$. В узле $v \in V$ для каждого признака $j = 1, \dots, d$ инкрементально поддерживаются поточные оценки $\mu_t^{(j)}(v)$, $\sigma_t^{2(j)}(v)$, эффективная мощность $n_t(v)$, а также априорные частоты классов $\pi_t(y | v)$ и весовые коэффициенты $w_t(y | v)$ (коррекция дисбаланса):

$$\rho_t(v) = 1 - e^{-\beta d S_{\text{drift}} \Delta_t}, \Delta_t = \tau_{\text{sync}} + t_{\text{comm}}, \quad (68)$$

$$\mu_t^{(j)} = (1 - \rho_t) \mu_{t-1}^{(j)} + \rho_t x_t^{(j)}, \sigma_t^{2(j)} = (1 - \rho_t) (\sigma_{t-1}^{2(j)} + \rho_t (x_t^{(j)} - \mu_{t-1}^{(j)})^2), \quad (69)$$

$$n_t = (1 - \rho_t) n_{t-1} + \rho_t, \pi_t(y | v) = (1 - \rho_t) \pi_{t-1}(y | v) + \rho_t \mathbf{1}\{y_t = y\}, \quad (70)$$

$$w_t(y | v) = \frac{(\pi_t(y | v) + \varepsilon)^{-1}}{\sum_{y'} (\pi_t(y' | v) + \varepsilon)^{-1}}, \varepsilon > 0. \quad (71)$$

Для разбиения используется неравенство Чебышёва. Доверительный интервал и кандидатные пороги [31]:

$$I_t^{(j)} = \left[\mu_t^{(j)} - k_t^{(j)} \sigma_t^{(j)}, \mu_t^{(j)} + k_t^{(j)} \sigma_t^{(j)} \right], \theta^{(j, \pm)} = \mu_t^{(j)} \pm k_t^{(j)} \sigma_t^{(j)}. \quad (72)$$

Адаптация ширины интервала по доле выходов за пределы выполняется на основе экспоненциально-взвешенного среднего (Exponential Moving Average, EMA [32]):

$$p_{\text{out},t}^{(j)} = \text{EMA}(\mathbf{1}\{|x_t^{(j)} - \mu_{t-1}^{(j)}| > k_{t-1}^{(j)} \sigma_{t-1}^{(j)}\}), p_{\text{in},t}^{(j)} = 1 - p_{\text{out},t}^{(j)}, \quad (73)$$

$$k_t^{(j)}(v) = \max \left\{ k_{\text{min}}, \sqrt{\frac{1}{p_{\text{in},t}^{(j)}(v) + \delta}} \right\}, \delta > 0, \quad (74)$$

что увеличивает $k_t^{(j)}$ при «тяжёлых хвостах» и мультимодальности. Регуляризованное снижение риска для кандидата (j, θ) :

$$\Delta L_{j,\theta}(v) = L(D_{v}; w_t) - \frac{n_L}{n_v} L(D_{v,L}; w_t) - \frac{n_R}{n_v} L(D_{v,R}; w_t) - \lambda_{\text{reg}}, \quad (75)$$

где $L(\cdot)$ — взвешенная (по w_t) энтропия/индекс Джини; разбиение выполняется, если

$$\max_{j, \theta \in \{\theta^{(j,-)}, \theta^{(j,+)}\}} \Delta L_{j,\theta}(v) \geq \tau_{\text{split}}, n_t(v) \geq n_{\text{min}}. \quad (76)$$

Подрезка по минимальной сложности:

$$C(T) = R_{\text{emp}}(T) + \alpha |T| \quad (\text{периодическая проверка с ограничениями по глубине и памяти}).$$

Федеративная калибровка с периодом τ_{sync} : агрегируются (с компрессией γ) верхнеуровневые статистики $\{\mu^{(j)}, \sigma^{2(j)}, \pi(y)\}$ и применяются к локальным узлам *без изменения структуры* дерева; контролируются коммуникационные издержки и согласованность параметров предобработки

Предлагаемый алгоритм описывается последовательным процессом, включающим следующие шаги.

Шаг 1. Инициализация. Задаётся корень v_{root} ; инициализируются $\mu_0^{(j)}, \sigma_0^{2(j)}, \pi_0(y | v_{\text{root}}), w_0(y | v_{\text{root}})$. Устанавливаются $\lambda_{\text{reg}}, \tau_{\text{split}}, \text{depth}_{\text{max}}, n_{\text{min}}, \tau_{\text{sync}}, \gamma, \beta_d, M_{\text{tot}}$.

Шаг 2. Получение наблюдения. Поступает $(x_t, y_t), x_t \in \mathbb{R}^d$.

Шаг 3. Маршрутизация без модификации структуры. Наблюдение последовательно спускается по текущим порогами до терминального узла v ($O(\text{depth}(T))$).

Шаг 4. Обновление коэффициента забывания. В узле v вычисляется $\rho_t(v) = 1 - \exp(-\beta_d S_{\text{drift}} \Delta_t)$.

Шаг 5. Инкрементальные обновления статистик. Для всех наблюдаемых j обновляются $\mu_t^{(j)}(v), \sigma_t^{2(j)}(v), n_t(v), \pi_t(y | v)$; пересчитываются $w_t(y | v)$. Обновляется $p_{\text{out},t}^{(j)}(v)$ как ЕМА индикатора выхода за $I_{t-1}^{(j)}(v)$.

Шаг 6. Формирование кандидатов разбиения. Для каждого j вычисляются $k_t^{(j)}(v)$, далее $\theta^{(j,-)}, \theta^{(j,+)}$. По обеим сторонам порога поддерживаются поточные счётчики метрик для ускорения.

Шаг 7. Оценка выигрыша и решение о сплите. Вычисляются $\Delta L_{j,\theta}(v)$. При $\max_{j,\theta} \Delta L_{j,\theta}(v) \geq \tau_{\text{split}}$ и $n_t(v) \geq n_{\text{min}}$ создаются дочерние v_L, v_R с инициализацией статистик от родителя и локальными пересчётами.

Шаг 8. Подрезка и контроль сложности. Периодически минимизируется $C(T)$; срезаются ветви с наименьшим вкладом в снижение риска до выполнения ограничений по $\text{depth}_{\text{max}}, |T|$ и M_{tot} .

Шаг 9. Федеративная калибровка. С периодом τ_{sync} выполняется агрегирование и применение калибровок (γ -сжатие), обеспечивается согласованность и устойчивость A_{stab} при контролируемой коммуникационной доле.

Шаг 10. Вычисление глобальных показателей. Обновляются $\text{Ass}_t, A_{\text{stab}}, K_{\text{scal}}, \eta_t$; проверяются ресурсные ограничения и условия качества.

Псевдокод алгоритма АИДР представлен в Листинге 2.

Листинг 2 – Псевдокод алгоритма АИДР

```

Inputs / Hyperparams:
  lambda_reg, tau_split, depth_max, n_min, tau_sync, gamma, beta_d, k_min, delta, alpha, M_tot
State:
  T ← init_tree() # v_root; инициализация mu, sigma^2, pi, w, k по узлам верхнего
уровня

# Обработка одного наблюдения (x_t, y_t)
Procedure Process(x_t ∈ ℝ^d, y_t):
  v ← RouteToLeaf(T, x_t) # O(depth(T))

  # 1) Динамика забывания (дрейф и возраст)
  delta_t ← tau_sync + t_comm(v) # возраст глобальной копии/пакета
  rho_t ← 1 - exp(-beta_d * S_drift * delta_t) # коэффициент забывания

  # 2) Инкрементальные статистики в узле v
  for j in 1..d:
    mu_prev ← mu[v][j]; sigma_prev ← sqrt(sigma^2[v][j])
    mu[v][j] ← (1-rho_t)*mu[v][j] + rho_t*x_t[j]
    sigma^2[v][j] ← (1-rho_t)*(sigma^2[v][j] + rho_t*(x_t[j]-mu_prev)^2)

  n[v] ← (1-rho_t)*n[v] + rho_t
  for y in Classes:
    pi[v][y] ← (1-rho_t)*pi[v][y] + rho_t*1{y_t=y}
  # нормировка priors и весов
  Normalize(pi[v]); w[v][y] ← ((pi[v][y]+epsilon)^(-1)) / sum_y'((pi[v][y']+epsilon)^(-1))

  # 3) Робастные интервалы (Чебышёв) и кандидаты порогов
  bestDelta ← -∞; best ← None
  for j in 1..d:
    # ЕМА доли выходов за интервал
    p_out[v][j] ← ЕМА(|x_t[j]-mu_prev| > k[v][j]*sigma_prev)
    p_in ← 1 - p_out[v][j]

```

```

k[v][j] ← max( k_min, sqrt(1/(p_in + δ)) )
θ_minus ← μ[v][j] - k[v][j]*sqrt(σ2[v][j])
θ_plus  ← μ[v][j] + k[v][j]*sqrt(σ2[v][j])

# 4) Оценка выигрыша сплита ΔL для {θ_minus, θ_plus}
for θ in {θ_minus, θ_plus}:
    (nL, nR, riskL, riskR) ← NodeSideStatsCache(v, j, θ) # скользящие
счётчики
    n_tot ← max(nL+nR, ε)
    ΔL ← L_parent(v; w[v]) - (nL/n_tot)*riskL - (nR/n_tot)*riskR - λ_reg
    if ΔL > bestΔL: bestΔL ← ΔL; best ← (j, θ)

# 5) Решение о сплите
if bestΔL ≥ τ_split and n[v] ≥ n_min and depth(v) < depth_max:
    (j*, θ*) ← best
    (v_L, v_R) ← SplitNode(v, j*, θ*)
    InitChildStatsFromParent(v_L, v_R, v)

# 6) Периодическая подрезка (минимальная комплексность)
if time_to_prune():
    Minimize C(T) = R_emp(T) + α*|T| subject to {depth_max, |T|, M_tot}

# 7) Периодическая федеративная калибровка
if time_to_sync(τ_sync):
    S_cal ← PullCalibrations(γ) # {μ^(j), σ2^(j), π(y)} (верхние
уровни)
    ApplyCalibrations(T, S_cal) # без изменения структуры
    ControlCommBudget() # χ=(γ*S_model)/(B*τ_sync) ≤ ξ_net

# 8) Обновление сводных метрик (опционально)
UpdateGlobalMetrics(Acc_t, A_stab, K_scal, η_t)

# Вспомогательные:
# L_parent — взвешенная (w) энтропия/Джини; ЕМА — экспоненциальное скользящее
среднее;
# NodeSideStatsCache — поддержка счётчиков по обе стороны порога; Normalize —
нормировка.
# Сложность на наблюдение: O(d * depth(T)); память — по {μ, σ2, n, π, w, k} в
узлах.

```

Свойства и корректность. Алгоритм поддерживает нормировку вероятностей и весов ($\sum_y \pi_t(y | v) = 1$, $\sum_y w_t(y | v) = 1$), демонстрируя устойчивую сходимость обновлений. Использование неравенства Чебышёва и адаптация коэффициента $k_t^{(j)}$ по доле внутренних точек p_{in} повышают робастность сплитов, снижая вероятность ложных разбиений при шумных и тяжёлых распределениях. Средняя вычислительная сложность одного наблюдения составляет $O(d \text{ depth}(T))$, а при сбалансированном росте дерева — $O(n \log n)$. Объём памяти определяется хранением локальных статистик $\{\mu^{(j)}, \sigma^{2(j)}, n, \pi(y)\}$ и согласуется с ограничением M_{tot} ; редкая синхронизация (τ_{sync}, γ) ограничивает сетевые издержки и поддерживает согласованность при асинхронности. Обновления выполняются по наблюдаемым признакам, а обработка пропусков реализуется локальными стратегиями (предыдущий путь или усреднение), не нарушая инкрементальности. На каждом шаге формируется актуальное дерево T_t с адаптивными порогами и параметрами $\theta_t^{(j,\pm)}(v)$, $k_t^{(j)}(v)$ и сводными метриками качества $Acc_t, A_{stab}, K_{scal}, \eta_t$, согласованными через федеративную калибровку.

Экспериментальное исследование

Экспериментальная часть исследования направлена на верификацию разработанной модели оптимизации качества распределённой потоковой обработки и сопряжённых с ней методов и алгоритмов в трёхзвенной архитектуре ИС на основе ФРВ. В рамках единого функционала потерь J производится согласованная оценка и настройка трёх групп критериев — производительности, точности и ресурсоёмкости. Оптимизируемые параметры включают период федеративной синхронизации τ_{sync} , степень компрессии обновлений γ , долю федеративных итераций ρ_{fed} и коэффициенты забывания в локальных статистиках ρ ; наблюдаемые метрики —

задержка отклика τ_{resp} , пропускная способность Q , точность Acc , доля коммуникаций ξ_{net} . Такая постановка обеспечивает проверку предположений о том, что совместная адаптация $\{\tau_{\text{sync}}, \gamma, \rho, \rho_{\text{fed}}\}$ минимизирует J без нарушения требований системы и ресурсных ограничений.

Процедура реализуется как оконный процесс: на каждом окне длительности Δ оцениваются λ_{in} (интенсивность поступления), показатель дрейфа S_{drift} , «возраст» модели $\Delta_i = \tau_{\text{sync}} + t_{\text{comm}}^{(i)}$, после чего выполняются локальные преобразования признаков и потоковые обновления, формируются сжатые дельты параметров и достаточные статистики, а затем — федеративная свёртка с весами свежести $\omega_i = \exp\{-\beta_d S_{\text{drift}} \Delta_i\}$. Параметры режима настраиваются проекционными шагами по $\partial J / \partial \tau_{\text{sync}}, \partial J / \partial \gamma, \partial J / \partial \rho_{\text{fed}}$ при контроле ограничений $\xi_{\text{net}} \leq \xi_{\text{net}}^{\text{max}}$ и $R \geq R_{\text{min}}$.

В ходе эксперимента проводится сравнительная оценка качества предложенного комплекса моделей, методов и алгоритмов относительно базового (неадаптивного) конвейера обработки потоковых данных. Основным объектом сравнения является интегральный функционал потерь (15). **Процесс эксперимента с применением предлагаемых в статье решений** описывается следующими шагами.

Шаг 1. Постановка и инициализация. Фиксируется интегральный функционал потерь J с частными критериями Φ_p, Φ_A, Φ_R и область допустимых решений; задаются начальные $\tau_{\text{sync}}, \gamma, \rho, \rho_{\text{fed}}$.

Шаг 2. Приём потока и оценка окна. На текущем окне оцениваются интенсивность λ_{in} , показатель дрейфа S_{drift} и «возраст» Δ ; для задержки и пропускной способности используется аппроксимация $M/M/1$.

Шаг 3. Преобразование признаков (ПППД). Экспоненциально-взвешенное нормирование, декорреляция с блочной аппроксимацией» (49)-(50); итоговая трансформация — по схеме метода.

Шаг 4. Локальное обучение и комбинированный режим (КОФС). Чередуются локальные и федеративные шаги в доле ρ_{fed} , определяемой мини-задачей на окне.

Шаг 5. Инкрементальное дерево решений (АИДР). Инкрементальные обновления статистик узлов, адаптивные пороги по ЕМА и неравенству Чебышёва, подрезка по сложности.

Шаг 6. Формирование локального пакета и компрессия. Узел формирует $S_i = \{n_i, \mu_i, \Sigma_i, p_{\text{out}}^{(i)}, \Delta\theta_i\}$ и сжимает $\Delta\theta_i$ с коэффициентом γ ; вычисляется коммуникационная квота $\chi = \frac{\gamma S_{\text{model}}}{B \tau_{\text{sync}}}$ и проверяется $\chi \leq \xi_{\text{net}}$.

Шаг 7. Федеративная агрегация (АФАС). Ассоциативная свёртка моментов и приращений с весами свежести $\omega_i = \exp\{-\beta_d S_{\text{drift}} \Delta_i\}$; финализация $\mu, \Sigma, \hat{p}_{\text{out}}, \theta_{\text{new}}$.

Шаг 8. Закрытие окна и адаптация (ФОПД). Вычисляются $\tau_{\text{resp}}, Q, \text{Acc}, \eta, \xi_{\text{net}}$ и значение J ; выполняется проекционный шаг по $u = (\tau_{\text{sync}}, \gamma, \rho, \rho_{\text{fed}})$ с учётом ограничений ξ_{net} и R .

Шаг 9. Мониторинг и переход. Протоколирование параметров и метрик; переход к следующему окну с обновлёнными настройками

Процесс эксперимента без применения предлагаемых в статье решений описывается следующими шагами.

Шаг 1. Постановка и инициализация. Задаётся интегральный критерий J и область допустимых значений. Режимы фиксируются заранее: $\tau_{\text{sync}}^{\text{fix}}, \gamma \equiv 1$ (без сжатия), $\rho_{\text{fed}}^{\text{fix}}$, коэффициенты забывания — постоянные.

Шаг 2. Приём потока на периферии. Буферизация с постоянным размером окна; оценивается лишь средняя интенсивность λ_{in} (без вычисления показателя дрейфа и возраста модели).

Шаг 3. Преобразование признаков. Стандартное центрирование и масштабирование по заранее полученным μ^*, σ^* ; при необходимости — разовое преобразование методом главных компонент, рассчитанное заранее; параметры предобработки не согласовываются между узлами.

Шаг 4. Локальное обучение. Обновление модели на каждом узле по накопленным данным; доля и период федеративных раундов постоянны ($\rho_{\text{fed}}^{\text{fix}}$, синхронизация каждые $\tau_{\text{sync}}^{\text{fix}}$).

Шаг 5. Модель принятия решения. Статическое дерево решений или иной классификатор, обученный заранее; при ухудшении качества выполняется полное переобучение вне потока данных по накопленному буферу (без инкрементальных разбиений и подрезки).

Шаг 6. Формирование и передача обновлений. Передаются полные параметры θ_i или градиенты без сжатия; коммуникационная доля оценивается как $\chi_{\text{base}} = \frac{S_{\text{model}}}{B \tau_{\text{sync}}^{\text{fix}}}$, ограничения по связи не контролируются адаптивно (только констатируются по факту).

Шаг 7. Централизованная агрегация. Простое усреднение параметров по числу примеров (классическое «среднее по участникам») без весов свежести и без отсеечения устаревших обновлений; обновление глобальной модели выполняется строго по расписанию.

Шаг 8. *Закрытие окна.* Рассчитываются $\tau_{\text{resp}}, Q, \text{Acc}, \eta, \xi_{\text{net}}$ и J ; параметры режима не пересчитываются автоматически (отсутствует контур адаптации), корректировки — только вручную между сериями испытаний.

Шаг 9. *Протоколирование и переход.* Фиксация метрик и параметров; переход к следующему окну по тем же неизменным правилам.

Эксперимент был проведен на гетерогенном вычислительном комплексе, спецификации которого указаны в Таблице 2. Использован текстовый корпус объёмом 1000 документов (~6 ГБ вес файлов .pdf и .docx, размер чистого текста ~763 МБ, $\sim 1,6 \times 10^8$ токенов), моделирующий реальный квазистационарный поток данных. Поступление данных имитировалось в виде непрерывной последовательности текстовых сегментов (фрагментов длиной 200–400 токенов) с интенсивностью 60 ± 10 сегментов в минуту, что позволило воспроизвести условия асинхронного обновления локальных статистик и дрейфа распределений. В исследовании рассматривались два профиля нагрузки — умеренный при $N = 4$ и высоконагруженный при $N = 7$, отражающие различное число активных исполнителей и интенсивность входного потока данных. Распределение вычислений осуществлялось с помощью библиотеки Ray для Python 3, также использовались следующие модули (помимо авторских): NumPy; SciPy; Pandas; Matplotlib; Seaborn; PyYAML; tqdm; scikit-learn; safetensors/MLX (mlx.core).

Таблица 2 – Спецификации гетерогенного вычислительного комплекса

Комплекующая / Узел комплекса	Узел 1	Узел 2
Процессор	12 vCPU (x86-64, Intel Core i7-8700 @ 4.6 GHz Turbo)	12 ядер ARM64 (8 P @ 4.05 ГГц + 4 E @ 2.8 ГГц)
Оперативная память (RAM)	32 ГБ DDR4 2666 МГц	18 ГБ LPDDR5 6400 МГц
Графическая подсистема (GPU)	NVIDIA GeForce RTX 2070 (8 ГБ GDDR6)	18 ядер GPU (общая память)
Нейросетевой сопроцессор (NPU)	-	16 ядер Neural Engine
Накопитель	240 ГБ NVMe	512 ГБ NVMe
Семейство операционных систем	Linux (64-bit)	Unix (64-bit)

На рисунке 1 показана динамика времени отклика τ_{resp} по окнам L_k при трёх значениях периода синхронизации (30, 60, 90 с) и нескольких профилях нагрузки. Базовый сценарий демонстрирует монотонный рост τ_{resp} и высокую вариативность по мере увеличения нагрузки; увеличение τ_{sync} усугубляет задержку. Предлагаемый комплекс решений удерживает τ_{resp} на существенно меньшем и более стабильном уровне для всех профилей; чувствительность к γ слаба. Поведение согласуется с моделью качества: адаптивный выбор τ_{sync} и γ снижает коммуникационную долю и удерживает устойчивость очереди, что уменьшает вклад в частный критерий Φ_p и, следовательно, в интегральный функционал J .

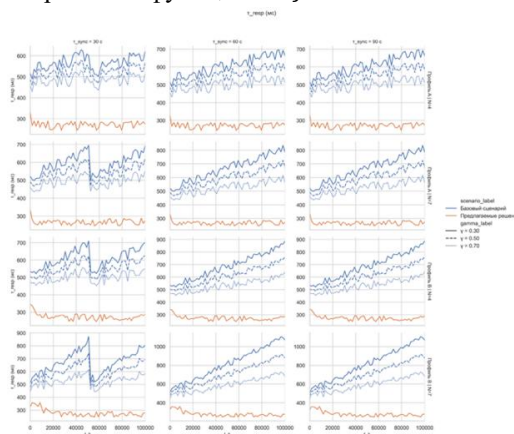


Рисунок 1 — Влияние τ_{sync} и γ на время отклика τ_{resp}

Рисунок 2 (а) иллюстрирует пропускную способность Q_{proc} (при $\tau_{\text{sync}} = 90$ с). В предложенном варианте Q_{proc} быстро достигает плато, практически пропорционального числу исполнителей N , что соответствует аппроксимации М/М/1 и выражениям для масштабируемости при $\rho = \lambda_{\text{in}}/(N\mu_1) < 1$. Базовый сценарий остаётся на заметно меньшем и почти горизонтальном уровне, что указывает на неиспользование параллелизма из-за фиксированных режимов обмена и отсутствия компрессии. Различия между $\gamma \in [0.3, 0.7]$ в адаптивной схеме невелики: управляющий контур компенсирует коммуникационные издержки выбором частоты согласований, сохраняя ресурсные ограничения.

На рисунке 2 (б) показана точность Acc (также при $\tau_{\text{sync}} = 90$ с). Предложенный подход обеспечивает быстрое насыщение Acc до значений, близких к эталонному уровню, и дальнейшую стабильность для всех профилей нагрузки. В базовом сценарии наблюдается деградация, вплоть до резкого обрушения точности в отдельных профилях по мере «старения» глобальной модели и рассогласования локальных параметров при фиксированных интервалах синхронизации. Это соответствует аналитической зависимости точности от «возраста» модели и показателя дрейфа S_{drift} и подтверждает эффективность весов свежести в АФАС и поточной предобработки признаков (46)-(47).

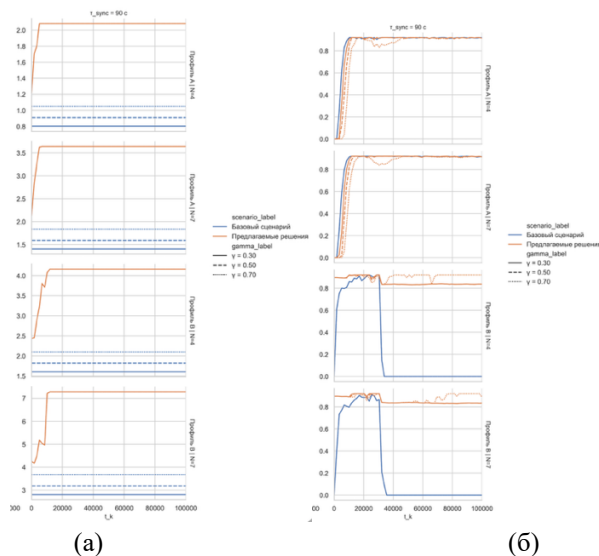


Рисунок 2 — Динамика метрик качества при $\tau_{\text{sync}} = 90$ с в сценариях А и В при различных значениях коэффициента компрессии γ :

а) пропускная способность Q_{proc} ; б) точность Acc

По результатам исследования можно сделать вывод: предлагаемый комплекс методов и алгоритмов обеспечивает одновременное снижение задержки, рост пропускной способности и устойчивость точности при соблюдении коммуникационного бюджета, т. е. уменьшение интегрального функционала потерь J по отношению к базовой (неадаптивной) линии во всех рассмотренных профилях и режимах $\tau_{\text{sync}}, \gamma$. Полученные эффекты согласованы с математической моделью и свидетельствуют о преимуществах совместной адаптации параметров и ассоциативной агрегации над фиксированными режимами обработки

Результаты и обсуждение

Экспериментальные исследования проводились для сравнения двух режимов функционирования системы: базовой схемы с фиксированными интервалами синхронизации и усреднением параметров без учёта актуальности обновлений, и предлагаемого комплекса методов и алгоритмов, реализующего совместную адаптацию параметров. В первом случае синхронизация выполнялась строго по расписанию, что приводило к накоплению ошибок и росту задержки отклика по мере «старения» глобальной модели. Во втором — управление режимом осуществлялось динамически, на основе наблюдаемых характеристик потока и показателя дрейфа, с регулировкой частоты синхронизации, степени компрессии и коэффициента забывания. Такой подход обеспечил сбалансированное распределение вычислительной нагрузки и устойчивость.

Результаты показали, что комплекс разработанных методов формирует адаптивный режим работы системы: метод ФОПД поддерживает квазистационарное равновесие между временем отклика и коммуникационными затратами; КОФС согласует долю локальных и федеративных итераций, минимизируя влияние «возраста» модели; ПППД уменьшает межузловую вариативность за счёт робастной нормировки и блочного «выбеливания» признаков; алгоритм АФАС обеспечивает порядконезависимую агрегацию статистик, исключая устаревшие обновления; АИДР стабилизирует качество классификации при изменении распределений данных.

В ходе анализа особое внимание уделялось выбору метрики точности Acc , используемой для оценки качества моделей и формирования частного критерия Φ_A в составе интегрального функционала потерь. В зависимости от типа задач и характера потоковых данных метрика Acc определяется как доля корректных классификаций на потоке (ассурагу), среднее значение F_1 -меры по классам либо семантическая метрика — например, BERTScore или BLEU для текстовых и мультимодальных данных. В настоящем исследовании верификация проводилась по показателю доли корректных классификаций, что обеспечивает сопоставимость с аналитическими зависимостями (35)-(37) и позволяет количественно оценить влияние параметров τ_{sync} , γ , ρ_{fed} и ρ на устойчивость качества при дрейфе данных. При необходимости метрика Acc может быть обобщена до интегральной оценки качества Q_{model} , включающей наряду с точностью показателя стабильности A_{stab} и согласованности C_{cons} .

По совокупным экспериментальным метрикам средняя задержка системы уменьшилась на 30%, пропускная способность увеличилась на 45%, при этом точность моделей осталась стабильной (рисунки 1 и 2). Коммуникационные затраты снизились за счёт адаптивного выбора частоты синхронизации и компрессии обновлений, а показатели согласованности и устойчивости модели к дрейфу выросли в пределах 10%. Эти результаты подтверждают, что совместная адаптация параметров и ассоциативная агрегация статистик обеспечивают устойчивое повышение качества распределённой потоковой обработки данных.

Заключение

В исследовании предложен целостный подход к организации федеративной обработки потоковых данных в распределённых интеллектуальных системах: от постановки многокритериальной оптимизации на окне до конкретных алгоритмических механизмов адаптации и устойчивой агрегации. Показано, что совместная настройка доли федеративных итераций, степени компрессии и коэффициента забывания в связке с поточной предобработкой и ассоциативной агрегацией обеспечивает одновременное снижение задержки, рост пропускной способности и стабилизацию точности при соблюдении заданного коммуникационного бюджета и требований отказоустойчивости.

Научная новизна подтверждена как аналитически, так и экспериментально: введение весов актуальности в федеративной агрегации с ассоциативной свёрткой моментов, согласованная поточная предобработка с выравниванием статистик между узлами и инкрементальная модель принятия решений с робастной настройкой порогов. Комплекс решений демонстрирует устойчивость к дрейфу, порядконезависимость при асинхронности и контролируруемую ресурсоёмкость.

Практическая значимость результатов проявляется в возможности нормативного закрепления принципов децентрализованного обучения и инференса, требований к согласованности и актуальности параметров моделей, а также к встроенному мониторингу показателей качества и устойчивости.

Дальнейшие исследования целесообразно направить на уточнение границ применимости полученных результатов для различных профилей нагрузок и топологий, а также на усиление механизмов надёжности и отказоустойчивости к сбоям и атакам в условиях гетерогенной сетевой среды ИС на основе ФРВ.

Список литературы

1. Бурый А.С. Облачные вычисления в цифровой трансформации информационных технологий // Правовая информатика. – Москва: Федеральное бюджетное учреждение «Научный центр правовой информации» при Министерстве юстиции Российской Федерации, 2021. – № 2. – С. 4–14.
2. Agrahari S., Singh A.K. Concept Drift Detection in Data Stream Mining: A literature review // Journal of King Saud University - Computer and Information Sciences. – 2022. – Т. 34, № 10, Part B. – С. 9523–9540.
3. Вишняков В.А. «Золотой треугольник»: сети интернет вещей, блокчейн, искусственный интеллект // Системный анализ и прикладная информатика. – Минск: Белорусский национальный технический университет, 2025. – № 1. – С. 63–67.
4. Behjati M., Alobaidy H.A.H., Nordin R., Abdullah N.F. UAV-assisted federated learning with hybrid LoRa P2P/LoRaWAN for sustainable biosphere // Frontiers in Communications and Networks. – 2025. – Т. 6. – DOI: 10.3389/frcmn.2025.1529453.
5. Wen C. FedRSClip: Federated Learning for Remote Sensing Scene Classification Using Vision-Language Models // arXiv preprint arXiv:2501.02461 [Электронный ресурс]. – 2025. – URL: <https://arxiv.org/abs/2501.02461> (дата обращения: 05.11.2025).
6. Cabitza F., Campagna A., Campagner A. Quod erat demonstrandum? - Towards a typology of the concept of explanation for the design of explainable AI // Expert Systems with Applications. – 2023. – Т. 213. – С. 118888.

7. Guy E. Deep Learning's Diminishing Returns // IEEE Spectrum [Электронный ресурс]. – URL: <https://spectrum.ieee.org/deep-learning-computational-cost> (дата обращения: 05.11.2025).
8. Little N. Generative AI's Energy Problem Today Is Foundational // IEEE Spectrum [Электронный ресурс]. – URL: <https://spectrum.ieee.org/ai-energy-consumption> (дата обращения: 05.11.2025).
9. Buchholz K. The Extreme Cost Of Training AI Models Like ChatGPT and Gemini // Forbes [Электронный ресурс]. – URL: <https://www.forbes.com/sites/katharinabuchholz/2024/08/23/the-extreme-cost-of-training-ai-models/> (дата обращения: 05.11.2025).
10. The 2025 AI Index Report // Stanford HAI [Электронный ресурс]. – URL: <https://hai.stanford.edu/ai-index/2025-ai-index-report> (дата обращения: 05.11.2025).
11. Wu E., Jensen R.E., Cella D., Bilenker J., Demetri G., Cutler D. How AI is used in FDA-authorized medical devices: a taxonomy across 1,016 authorizations // npj Digital Medicine [Электронный ресурс]. – URL: <https://www.nature.com/articles/s41746-025-01800-1> (дата обращения: 05.11.2025).
12. Li S., Li K., Li J., Zhang Y., Zheng S., Zhang Z. Efficient and scalable reinforcement learning for large-scale network control // Nature Machine Intelligence [Электронный ресурс]. – URL: <https://www.nature.com/articles/s42256-024-00879-7> (дата обращения: 05.11.2025).
13. ISO/IEC TS 8200:2024. Информационные технологии. Искусственный интеллект. Управляемость автоматизированных систем искусственного интеллекта. — Женева: Международная организация по стандартизации (ISO), 2024. — 26 с.
14. ГОСТ Р 59277–2020. Системы искусственного интеллекта. Классификация систем искусственного интеллекта. – Москва : Стандартинформ, 2020. – 11 с.
15. ГОСТ Р 57700.27–2020. Высокопроизводительные вычислительные системы. Термины и определения. – Москва : Стандартинформ, 2020. – 16 с.
16. Loghin D., Teo Y.M. The Energy Efficiency of Modern Multicore Systems // Proceedings of the 47th International Conference on Parallel Processing Companion. – Eugene OR USA: ACM, 2018. – С. 1–10.
17. Sherzer E. Can machines solve general queueing systems? // arXiv preprint arXiv:2202.01729 [Электронный ресурс]. – 2022. – URL: <https://arxiv.org/abs/2202.01729> (дата обращения: 05.11.2025).
18. Ермаков С.Р. Метод комбинированного обучения для потоковой интеллектуальной образовательной системы // Научно-технический вестник Поволжья. — 2025. — № 1. — С. 180-185.
19. Alsulaimawi Z. Enhancing Security in Federated Learning through Adaptive Consensus-Based Model Update Validation // arXiv preprint arXiv:2403.04803 [Электронный ресурс]. – 2024. – URL: <https://arxiv.org/abs/2403.04803> (дата обращения: 05.11.2025).
20. Ермаков С.Р., Зыков С.В. Метод масштабирования признаков для потоковой интеллектуальной образовательной системы // Научно-технический вестник Поволжья. – 2025. — № 2. — С. 118-122.
21. Nanekar Y.H., Kheiri M.T., Faez K. A novel federated learning framework for medical imaging: Resource-efficient approach combining PCA with early stopping // Medical Physics. – 2025. – DOI: 10.1002/mp.18064 [Электронный ресурс]. – URL: <https://aapm.onlinelibrary.wiley.com/doi/full/10.1002/mp.18064> (дата обращения: 05.11.2025).
22. Garivier A. On Sparsity and Sub-Gaussianity in the Johnson-Lindenstrauss Lemma // arXiv preprint arXiv:2409.06275 [Электронный ресурс]. – 2024. – URL: <https://arxiv.org/abs/2409.06275> (дата обращения: 05.11.2025).
23. Kessy A., Lewin A., Strimmer K. Optimal whitening and decorrelation // The American Statistician. – 2018. – Т. 72, № 4. – С. 309–314.
24. Stotsky A. Systematic Review of Newton-Schulz Iterations with Unified Factorizations: Integration in the Richardson Method and Application to Robust Failure Detection in Electrical Networks // arXiv preprint arXiv:2208.04068 [Электронный ресурс]. – 2022. – URL: <https://arxiv.org/abs/2208.04068> (дата обращения: 05.11.2025).
25. Lupo Pasini M., Turcksin B., Ge W., Fattebert J.-L. A parallel strategy for density functional theory computations on accelerated nodes // Parallel Computing. – 2020. – Т. 100. – С. 102703.
26. Veprikov A. Leveraging Coordinate Momentum in SignSGD and Muon: Memory-Optimized Zero-Order // arXiv preprint arXiv:2506.04430 [Электронный ресурс]. – 2025. – URL: <https://arxiv.org/abs/2506.04430> (дата обращения: 05.11.2025).
27. Gül B. SyncFed: Time-Aware Federated Learning through Explicit Timestamping and Synchronization // arXiv preprint arXiv:2506.09660 [Электронный ресурс]. – 2025. – URL: <https://arxiv.org/abs/2506.09660> (дата обращения: 05.11.2025).

28. Vlaski S., Rizk E., Sayed A.H. Robust Aggregation for Federated Learning // *IEEE Transactions on Signal Processing* [Электронный ресурс]. – 2022. – DOI: 10.1109/TSP.2022.3151601. – URL: <https://ieeexplore.ieee.org/abstract/document/9721118> (дата обращения: 05.11.2025).
29. Li Z., Lin T., Shang X., Wu C. Revisiting Weighted Aggregation in Federated Learning with Neural Networks // *Proceedings of the 40th International Conference on Machine Learning*. – PMLR, 2023. – С. 19767–19788.
30. Ермаков С.Р. Алгоритм инкрементального дерева решений для потоковой интеллектуальной образовательной системы // *Научно-технический вестник Поволжья*. — 2024. — № 12. — С. 292-296.
31. Al-Saedi A.A., Voeva V. Group-Personalized Federated Learning for Human Activity Recognition Through Cluster Eccentricity Analysis // *Communications in Computer and Information Science*. – 2023. – Т. 1826. – С. 505–519 [Электронный ресурс]. – URL: https://link.springer.com/chapter/10.1007/978-3-031-34204-2_41 (дата обращения: 05.11.2025).
32. Jafari Naeimi M.H., Norouzi A., Abdi A. GRAD: Real-Time Gated Recurrent Anomaly Detection in Autonomous Vehicle Sensors Using Reinforced EMA and Multi-Stage Sliding Window Techniques // *arXiv preprint arXiv:2510.23327* [Электронный ресурс]. – 2025. – URL: <https://arxiv.org/abs/2510.23327> (дата обращения: 05.11.2025).

References

1. Buryi A.S. *Cloud Computing in Digital Transformation of Information Technologies* // *Legal Informatics*. – Moscow: Federal Budgetary Institution "Scientific Center for Legal Information" under the Ministry of Justice of the Russian Federation, 2021. – No. 2. – P. 4–14.
2. Agrahari S., Singh A.K. Concept Drift Detection in Data Stream Mining: A literature review // *Journal of King Saud University - Computer and Information Sciences*. – 2022. – Vol. 34, No. 10, Part B. – P. 9523–9540.
3. Vishnyakov V.A. "Golden Triangle": Internet of Things Networks, Blockchain, Artificial Intelligence // *System Analysis and Applied Informatics*. – Minsk: Belarusian National Technical University, 2025. – No. 1. – P. 63–67.
4. Behjati M., Alobaidy H.A.H., Nordin R., Abdullah N.F. UAV-assisted federated learning with hybrid LoRa P2P/LoRaWAN for sustainable biosphere // *Frontiers in Communications and Networks*. – 2025. – Vol. 6. – DOI: 10.3389/frcmn.2025.1529453.
5. Wen C. FedRSClip: Federated Learning for Remote Sensing Scene Classification Using Vision-Language Models // *arXiv preprint arXiv:2501.02461* [Electronic resource]. – 2025. – URL: <https://arxiv.org/abs/2501.02461> (accessed: 05.11.2025).
6. Cabitza F., Campagna A., Campagner A. Quod erat demonstrandum? - Towards a typology of the concept of explanation for the design of explainable AI // *Expert Systems with Applications*. – 2023. – Vol. 213. – P. 118888.
7. Guy E. Deep Learning's Diminishing Returns // *IEEE Spectrum* [Electronic resource]. – URL: <https://spectrum.ieee.org/deep-learning-computational-cost> (accessed: 05.11.2025).
8. Little N. Generative AI's Energy Problem Today Is Foundational // *IEEE Spectrum* [Electronic resource]. – URL: <https://spectrum.ieee.org/ai-energy-consumption> (accessed: 05.11.2025).
9. Buchholz K. The Extreme Cost Of Training AI Models Like ChatGPT and Gemini // *Forbes* [Electronic resource]. – URL: <https://www.forbes.com/sites/katharinabuchholz/2024/08/23/the-extreme-cost-of-training-ai-models/> (accessed: 05.11.2025).
10. The 2025 AI Index Report // *Stanford HAI* [Electronic resource]. – URL: <https://hai.stanford.edu/ai-index/2025-ai-index-report> (accessed: 05.11.2025).
11. Wu E., Jensen R.E., Cella D., Bilenker J., Demetri G., Cutler D. How AI is used in FDA-authorized medical devices: a taxonomy across 1,016 authorizations // *npj Digital Medicine* [Electronic resource]. – URL: <https://www.nature.com/articles/s41746-025-01800-1> (accessed: 05.11.2025).
12. Li S., Li K., Li J., Zhang Y., Zheng S., Zhang Z. Efficient and scalable reinforcement learning for large-scale network control // *Nature Machine Intelligence* [Electronic resource]. – URL: <https://www.nature.com/articles/s42256-024-00879-7> (accessed: 05.11.2025).
13. ISO/IEC TS 8200:2024. Information technology — Artificial intelligence — Governance of autonomous artificial intelligence systems. Geneva: International Organization for Standardization (ISO), 2024. 26 p.
14. GOST R 59277–2020. Artificial Intelligence Systems. Classification of Artificial Intelligence Systems. – Moscow : Standartinform, 2020. – 11 p.
15. GOST R 57700.27–2020. High-Performance Computing Systems. Terms and Definitions. – Moscow : Standartinform, 2020. – 16 p.
16. Loghin D., Teo Y.M. The Energy Efficiency of Modern Multicore Systems // *Proceedings of the 47th International*

Conference on Parallel Processing Companion. – Eugene OR USA: ACM, 2018. – P. 1–10.

17. Sherzer E. Can machines solve general queueing systems? // arXiv preprint arXiv:2202.01729 [Electronic resource]. – 2022. – URL: <https://arxiv.org/abs/2202.01729> (accessed: 05.11.2025).

18. Ermakov S.R., Zykov S.V. Feature Scaling Method for a Stream Intelligent Educational System // Scientific and Technical Bulletin of the Volga Region. – 2025. – No. 2. – Pp. 118–122.

19. Alsulaimawi Z. Enhancing Security in Federated Learning through Adaptive Consensus-Based Model Update Validation // arXiv preprint arXiv:2403.04803 [Electronic resource]. – 2024. – URL: <https://arxiv.org/abs/2403.04803> (accessed: 05.11.2025).

20. Ermakov S.R. Combined Learning Method for a Stream Intelligent Educational System // Scientific and Technical Bulletin of the Volga Region. – 2025. – No. 1. – Pp. 180–185.

21. Nanekaran Y.H., Kheiri M.T., Faez K. A novel federated learning framework for medical imaging: Resource-efficient approach combining PCA with early stopping // Medical Physics. – 2025. – DOI: 10.1002/mp.18064 [Electronic resource]. – URL: <https://aapm.onlinelibrary.wiley.com/doi/full/10.1002/mp.18064> (accessed: 05.11.2025).

22. Garivier A. On Sparsity and Sub-Gaussianity in the Johnson-Lindenstrauss Lemma // arXiv preprint arXiv:2409.06275 [Electronic resource]. – 2024. – URL: <https://arxiv.org/abs/2409.06275> (accessed: 05.11.2025).

23. Kessy A., Lewin A., Strimmer K. Optimal whitening and decorrelation // The American Statistician. – 2018. – Vol. 72, No. 4. – P. 309–314.

24. Stotsky A. Systematic Review of Newton-Schulz Iterations with Unified Factorizations: Integration in the Richardson Method and Application to Robust Failure Detection in Electrical Networks // arXiv preprint arXiv:2208.04068 [Electronic resource]. – 2022. – URL: <https://arxiv.org/abs/2208.04068> (accessed: 05.11.2025).

25. Lupo Pasini M., Turcsin B., Ge W., Fattebert J.-L. A parallel strategy for density functional theory computations on accelerated nodes // Parallel Computing. – 2020. – Vol. 100. – P. 102703.

26. Veprikov A. Leveraging Coordinate Momentum in SignSGD and Muon: Memory-Optimized Zero-Order // arXiv preprint arXiv:2506.04430 [Electronic resource]. – 2025. – URL: <https://arxiv.org/abs/2506.04430> (accessed: 05.11.2025).

27. Gül B. SyncFed: Time-Aware Federated Learning through Explicit Timestamping and Synchronization // arXiv preprint arXiv:2506.09660 [Electronic resource]. – 2025. – URL: <https://arxiv.org/abs/2506.09660> (accessed: 05.11.2025).

28. Vlaski S., Rizk E., Sayed A.H. Robust Aggregation for Federated Learning // IEEE Transactions on Signal Processing [Electronic resource]. – 2022. – DOI: 10.1109/TSP.2022.3151601. – URL: <https://ieeexplore.ieee.org/abstract/document/9721118> (accessed: 05.11.2025).

29. Li Z., Lin T., Shang X., Wu C. Revisiting Weighted Aggregation in Federated Learning with Neural Networks // Proceedings of the 40th International Conference on Machine Learning. – PMLR, 2023. – P. 19767–19788.

30. Ermakov S.R. Incremental Decision Tree Algorithm for a Stream Intelligent Educational System // Scientific and Technical Bulletin of the Volga Region. – 2024. – No. 12. – Pp. 292–296.

31. Al-Saedi A.A., Boeva V. Group-Personalized Federated Learning for Human Activity Recognition Through Cluster Eccentricity Analysis // Communications in Computer and Information Science. – 2023. – Vol. 1826. – P. 505–519 [Electronic resource]. – URL: https://link.springer.com/chapter/10.1007/978-3-031-34204-2_41 (accessed: 05.11.2025).

32. Jafari Naeimi M.H., Norouzi A., Abdi A. GRAD: Real-Time Gated Recurrent Anomaly Detection in Autonomous Vehicle Sensors Using Reinforced EMA and Multi-Stage Sliding Window Techniques // arXiv preprint arXiv:2510.23327 [Electronic resource]. – 2025. – URL: <https://arxiv.org/abs/2510.23327> (accessed: 05.11.2025).