

## Инструменты мониторинга данных устройств интернета вещей

Камышев Никита Львович, Исаева Ольга Сергеевна

Институт вычислительного моделирования СО РАН,

Россия, Красноярск, [nik.novov@gmail.com](mailto:nik.novov@gmail.com)

**Аннотация.** В статье представлено решение задачи создания автономной системы мониторинга микроклимата технических помещений, способной к оперативному выявлению аномалий в условиях нестационарных данных и ограниченной сенсорной инфраструктуры. Цель исследования – разработка инструментов мониторинга, которые бы обеспечили непрерывный контроль, детекцию аномалий и их интерпретацию. Основное содержание работы включает разработку функциональной модели системы, реализацию модулей сбора телеметрии и гетерогенного хранения данных, а также внедрение оригинального гибридного алгоритма анализа временных рядов. Ключевым элементом стал предложенный метод, сочетающий спектральный анализ для выделения периодических компонентов сигнала и адаптивные статистические пороги, динамически рассчитываемые на основе скользящих исторических окон. В отличие от существующих аналогов, опирающихся на ресурсоемкое глубокое обучение, предложенный подход использует физически обоснованные признаки для адаптации к сезонной вариативности данных без этапа предварительного обучения на размеченных датасетах. Для унификации подхода к обнаружению аномалий и маршрутизации оповещений настроены специальные правила оповещения, выполнена интеграция с системой визуализации данных и создан чат-бот, выполняющий уведомления и запросы. Для повышения отказоустойчивости внедрён механизм контроля живучести компонентов. Научная новизна работы заключается в предложении гибридного адаптивного метода детекции аномалий, базирующегося на спектрально-статистическом анализе временных рядов. Предложенный подход обеспечивает интерпретируемость результатов, раскрывая физическую природу сбоев через декомпозицию сигнала на гармонические составляющие. Результаты верификации на реальном массиве данных объёмом 5,3 млн записей подтвердили эффективность подхода: полнота детекции составила 0,991 при точности 0,877. Нагрузочное тестирование продемонстрировало устойчивость архитектуры при низкой утилизации ресурсов процессора (<6%) и наличие резерва для масштабирования. Опытная эксплуатация позволила выявить инциденты, связанные с отказом системы кондиционирования. Разработанная система обеспечивает полный цикл от сбора данных до превентивного оповещения, предоставляя понятные инструменты диагностики для предотвращения аварийных ситуаций, поддерживая два уровня взаимодействия: упрощённый интерфейс для конечных пользователей и расширенные аналитические функции для специалистов.

**Ключевые слова:** Интернет вещей, мониторинг микроклимата, обнаружение аномалий, спектральный анализ, адаптивные пороги, отказоустойчивость, Prometheus, AlertManager

**Цитирование:** Камышев Н.Л. Инструменты мониторинга данных устройств интернета вещей / Н.Л. Камышев, О.С. Исаева // Информационные и математические технологии в науке и управлении, 2026. – № 2(42). – С. 206-219. – DOI:10.25729/ESI.2026.42.2.016.

**Введение.** За последние годы технологии Интернета вещей (Internet of Things, IoT) прочно вошли в научные и промышленные приложения, обеспечивая возможность непрерывного сбора, передачи, хранения и анализа данных в реальном времени [1]. IoT-устройства играют ключевую роль в автоматизации и оптимизации технологических процессов и находят широкое применение в таких областях, как умные дома, промышленное производство, здравоохранение, сельское хозяйство и др. [2]. Современные исследования в этой сфере сосредоточены на решении задач, связанных с обеспечением безопасного межмашинного взаимодействия, повышением отказоустойчивости систем, адаптацией к динамически изменяющимся условиям эксплуатации, оптимальным размещением сенсоров, а также разработкой методов и инструментов для анализа больших объёмов данных, генерируемых распределёнными устройствами [3].

В Институте вычислительного моделирования СО РАН ведутся работы по созданию специализированной системы мониторинга микроклимата технологических помещений, в

которых размещается телекоммуникационное оборудование. Такое оборудование характеризуется высоким уровнем тепловыделения и строгими требованиями к параметрам окружающей среды (температура, влажность, PM2.5), поскольку их отклонение от нормативных значений может привести к снижению надёжности или полному отказу аппаратуры. В рамках проекта разработана распределённая IoT-инфраструктура [4] на основе протокола MQTT, объединяющая сенсорные узлы, агенты сбора данных, серверные компоненты и программное обеспечение, совместимое с различными IoT-платформами для решения прикладных задач. Используемая система не включает инструментов автоматического выявления аномалий в данных.

Существующие решения для мониторинга микроклимата могут быть реализованы либо на основе универсальных платформ, либо с использованием специализированных программных систем. К числу широко распространённых IoT-платформ относятся ThingSpeak, AWS IoT Core, Google Cloud IoT Core и IBM Watson IoT [5]. Они предоставляют гибкие инструменты для подключения устройств, управления данными, визуализации и базовой аналитики. Однако при применении в задачах критического мониторинга такие платформы демонстрируют высокую зависимость от облачной инфраструктуры, ограниченные возможности локальной обработки данных, повышенные требования к вычислительным ресурсам, а также необходимость существенной доработки и адаптации к специфике предметной области. Специализированные исследовательские и коммерческие системы [6] опираются на глубокую интеграцию с характеристиками предметной области для повышения качества результатов. Например, в работе [7] представлена система управления знаниями IoT-CANE с многоуровневой архитектурой для обработки больших данных. Прототипы на базе микроконтроллеров демонстрируют высокую точность измерений при низкой стоимости [8]. Для создания программных комплексов интеллектуальных зданий применяются структурные и поведенческие шаблоны, позволяющие обеспечивать устойчивость взаимодействия компонентов и реализовывать различные сценарии управления климатом [9]. Однако представленные архитектурные подходы в большей мере направлены на сбор и визуализацию данных и требуют значительной доработки для интеграции со специализированными алгоритмами анализа временных рядов.

В последние годы значительное число исследований посвящено применению методов машинного обучения для мониторинга технического состояния оборудования. Наблюдается переход от традиционных стратегий к адаптивно-прогнозным системам управления, которые интегрируют модели прогнозирования с механизмами адаптации к неопределённостям [10]. Для повышения эффективности комбинируют эти подходы с методами глубокого обучения, способными в режиме онлайн корректировать параметры модели на основе поступающих данных с сенсоров в условиях высокой стохастичности среды. Активно развиваются подходы на основе автоэнкодеров и рекуррентных нейронных сетей (LSTM, GRU), позволяющие выявлять сложные нелинейные зависимости и аномалии в многомерных временных рядах IoT-сенсоров. Пример развития данного направления можно найти в работе [11]. Авторами проведено сравнение эффективности различных архитектур автоэнкодеров и показано, что использование ячеек LSTM внутри автоэнкодера позволяет модели запоминать долгосрочные зависимости в траектории изменения температуры и детектировать отклонения раньше, по сравнению с классическими (RNN) и полносвязными (DNN) сетями. Однако авторы отмечают, что достижение такого преимущества требует тщательной настройки гиперпараметров (количество слоёв, функция потерь) и процедуры нормализации данных для предотвращения переобучения, а порог срабатывания определяется статистически на основе максимальной ошибки реконструкции. Аналогичные результаты получены и для других областей: так, в

работе [12] гибридные модели успешно применялись для детекции неисправностей ветрогенераторов на основе вибрационных данных после сложной предобработки.

Несмотря на высокую точность, подходы на базе глубокого обучения имеют ограничения для задач оперативного мониторинга. Несмотря на превосходство моделей глубокого обучения в точности детекции аномалий по сравнению с традиционными методами, их внедрение в промышленную эксплуатацию ограничено эффектом «черного ящика» [13]. Такие модели без методов объяснимого искусственного интеллекта не предоставляют понятных причин срабатывания систем оповещения. Для эффективного принятия решений, кроме выявления аномалии, необходимо понимать, является ли она, например, изменением суточного цикла, выходом из строя элементов системы оповещения или единичным выбросом. Кроме того, обучение и применение нейросетевых моделей требуют значительных вычислительных мощностей и энергопотребления. Модели требуют больших объёмов размеченных или тщательно отфильтрованных данных для обучения, чтобы избежать переобучения на артефактах или случайных отклонениях в данных.

Целью работы является создание автономной IoT-системы, способной обеспечивать непрерывный контроль параметров микроклимата в технических помещениях и выявлять аномалии для снижения рисков выхода из строя чувствительного оборудования. В отличие от решений, опирающихся на сложные архитектуры глубокого обучения, используется гибридный метод, сочетающий спектральный анализ и адаптивные статистические пороги. Данный подход близок к современным методам интерпретируемой диагностики, основанным на извлечении физически обоснованных статистических признаков [14]. Такая стратегия позволяет преодолеть ограничения «черного ящика», свойственные глубоким нейросетям, и обеспечивает детектирование периодических нарушений с высокой интерпретируемостью логики принятия решений.

**1. Выбор инструментов для реализации системы мониторинга.** При выборе системы управления базами данных для хранения информации с IoT-устройств были проанализированы как специализированные решения, ориентированные на хранение и обработку временных рядов (TSDB), так и универсальные системы классов NoSQL и SQL. В Таблице 1 представлены результаты сравнения по совокупности критериев, необходимых для задач непрерывного сбора, хранения и анализа временных рядов.

На основе анализа характеристик, представленных производителями и аналитическим ресурсом DB-Engines, а также по результатам нагрузочного тестирования, были выделены ключевые параметры производительности и ресурсной эффективности. В качестве сценария тестирования использовалась последовательная пакетная запись данных в соответствии со структурой полей датчика CL210-E без предварительного создания индексов.

**Таблица 1.** Сравнительный анализ систем управления базами данных

Параметры	MongoDB	InfluxDB	PostgreSQL	ClickHouse
Тип базы данных	Документо-ориентированная NoSQL	Временных рядов	Реляционная SQL	Колоночная, ориентированная на аналитику
Хранение временных рядов	Требуется настройка	Изначально хранит так	Создание временных таблиц	Имеется
Совместимость с SQL	PromQL	Limited SQL	SQL	Full SQL и собственные конструкции

Индексация	Динамическая индексация полей	Индексация по тегам и временным меткам	Глобальная индексация	Специальные индексы для колоночной структуры
Оптимальное использование	Хранилище произвольных данных	Сбор метрик и мониторинг	Анализ исторических данных	Большие данные
Горизонтальное масштабирование	Ограничено (есть шардирование)	Среднее	Шардирование	Имеется
Производительность	Высокая	Высокая для временных данных	Высокая	Исключительно быстрая для OLAP
Объем памяти на 1000 данных	14.35 КБ	19.5 КБ	17.32 КБ	9.8 КБ

Сравнительный анализ показал преимущества систем MongoDB и PostgreSQL в контексте поставленных задач. Итоговая оценка проводилась по результатам измерения скорости записи и объема дискового пространства (рис. 1). При достижении объема в 10 млн записей явное преимущество по экономичности хранения показала MongoDB 143.47 МБ, в отличие от PostgreSQL, объем которой составил 173.18 МБ (на 20.7% больше).

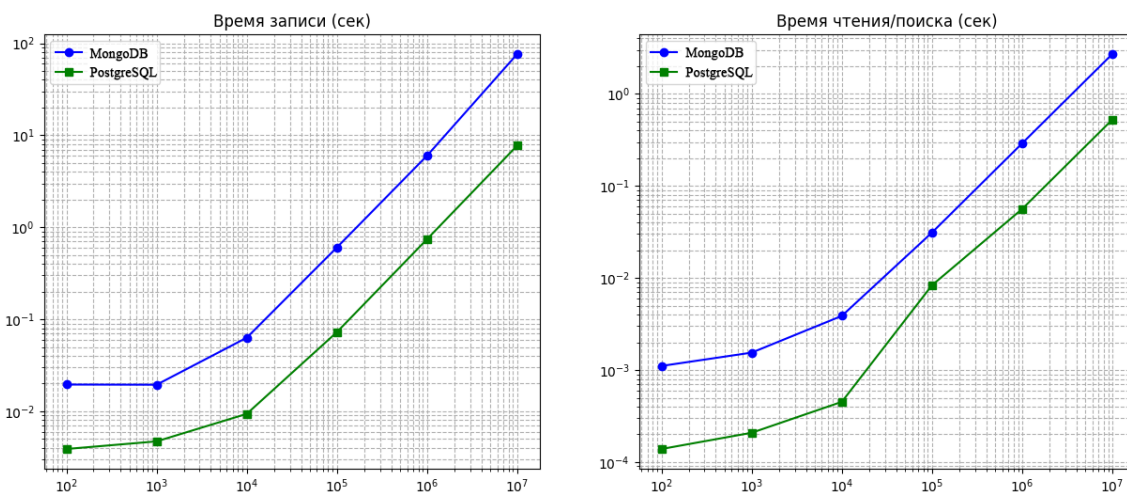


Рис. 1. Результаты тестов БД

Определяющими критериями стали гибкость документно-ориентированной модели данных, позволяющая эффективно хранить разнородные сведения, поступающие от различных IoT-устройств, а также встроенная поддержка временных рядов [15]. Их использование обеспечивает автоматическую оптимизацию хранения и индексацию по временным меткам, что существенно упрощает обработку динамических данных. Кроме того, система имеет поддержку языка Python и широкую интеграцию с популярными инструментами визуализации, такими, как Grafana и Tableau. По эффективности хранения MongoDB уступает специализированным решениям, таким, как InfluxDB и ClickHouse, но её способность поддерживать гибридные сценарии, предполагающие одновременное хранение как структурированных метрик, так и метаданных устройств, делает её практичным выбором для решения поставленной задачи.

Для реализации модуля анализа и визуализации собранных телеметрических данных был проведен сравнительный анализ популярных решений: Prometheus, Grafana Cloud, Graphite и Graylog. Оценка инструментов осуществлялась по комплексу критериев, включающих архитектурную автономность, нативную поддержку временных рядов, механизмы сбора

данных (Pull/Push), функциональность подсистем оповещения и степень совместимости с выбранной СУБД MongoDB. По результатам анализа в качестве основного инструмента мониторинга был выбран Prometheus. Его преимущества – это архитектурная автономность и специализация на работе с временными рядами. Prometheus представляет собой самодостаточную систему, включающую компоненты для сбора метрик, хранения данных (встроенная TSDB), обработки запросов и генерации оповещений (через Alertmanager). Возможность размещения в изолированных сетевых средах делает его одним из наиболее подходящих решений для промышленных и корпоративных приложений с повышенными требованиями к информационной безопасности.

## 2. Проектирование инструментов мониторинга данных устройств интернета вещей

**2.1. Функциональная модель.** Для обеспечения системного подхода к проектированию была разработана функциональная модель в нотации IDEF0. В соответствии с выбранной нотацией процессы описаны в виде иерархических диаграмм, где каждая функция включает следующие компоненты: входы, выходы, контролирующие факторы и механизмы (рис. 2).



Рис. 2. Контекстная диаграмма модели в нотации IDEF0

Целевой функцией верхнего уровня модели является формирование оповещений о критических отклонениях параметров микроклимата. Модель обеспечивает модульность архитектуры, способствует комплексной реализации необходимых функций и упрощает анализ взаимодействия между компонентами системы. На основе модели выполнена формализация подсистем и верификация их поведения.

Формальное описание модели:  $F = \{A_1, A_2, A_3, A_4\}$  – множество ключевых функций системы, где каждая функция  $A_i$  описывается кортежем:  $A_i = \langle I_i, O_i, C_i, M_i \rangle$ , где  $I_i$ : входы,  $O_i$  – выходы, представляющие результаты обработки, объекты,  $C_i$  – контролирующие факторы, ограничения, правила, нормативные требования, конфигурации,  $M_i$  – механизмы, представляющие собой программные и аппаратные средства реализации. Далее приведено описание функции  $A_i$ .

**2.2. Получение данных ( $A_1$ ).** Функция предназначена для сбора данных в реальном времени от сенсорных устройств через протокол MQTT, а также от внешних источников через API (публичные метеоданные). Реализация выполнена в виде клиентского приложения на

языке Python, поддерживающего параллельную обработку потоков. Данные поступают в реальном времени через подписки на темы  $T = \{t_1, \dots, t_7\}$ , ассоциированные с собираемыми показателями (температура, влажность и др.), и преобразуются в JSON-формат для дальнейшей обработки. Используются настройки MQTT (конфигурация брокера Mosquitto) и ключи API для аутентификации.

В кортеж  $A_1$  входят:  $I_1 = \{D_{MQTT}, D_{API}\}$  – потоки данных от датчиков и внешних API,  $O_1 = D'$  – объединённые данные в формате JSON,  $C_1 = \{MQTT_{conf}, API_{keys}\}$  – настройки протокола и ключи доступа,  $M_1 = \{Python_{API}\}$  – клиент сбора данных на Python.

Функция обеспечивает параллельное отображение  $pl: S \rightarrow X$ , где  $S = \{s_1, s_2, \dots, s_n\}$  – множество датчиков ( $n$ ), генерирующее временной ряд  $X = \{x_{ij} \mid i = 1 \dots n, j = 1 \dots m\}$ ,  $m$  – множество параметров,  $x_{ij} = \langle temp_t, hum, dew_p, s_t, pm2.5, \dots \rangle$  – кортеж значений (температуры, влажности, точки росы, состояния датчика, показателей мелкодисперсной пыли и другие). Доступ к темам задаётся подпиской  $sub: T \rightarrow X$ , где  $T = \{t_1, \dots, t_m\}$ , с параллельной обработкой для минимизации задержек.

**2.3. Сохранение метрик ( $A_2$ ).** Функция отвечает за долговременное хранение и временное буферирование данных. Реализация выполнена через интеграцию с системами Prometheus (для оперативного мониторинга) и MongoDB (для долгосрочного хранения).

В Python-приложении создаются классы для каждого датчика, содержащие временные буферы объёмом до 20 элементов. Эти буферы используются для локального анализа и формирования агрегированных метрик. Prometheus собирает метрики приложения. Интервал сбора  $\Delta_s = 10s$  в соответствии с конфигурационным файлом `prometheus.yml`. Одновременно данные записываются в коллекции MongoDB.

Кортеж  $A_2$  состоит из  $I_2 = X'$  – нормализованные входные данные,  $O_2 = Met \cup \{W_i\}$  – набор метрик и временные буферы,  $C_2 = \{prometheus.yml, K_i\}$  – конфигурация сбора и параметры классов (размер буфера, частота обновления),  $M_2 = \{MongoDB, Prometheus\}$  – используемые программные механизмы.

Функция импорта данных определяется как:  $imp: X' \rightarrow Met$ , где  $Met = \{m_k(t)\}$ . Каждый класс  $K_i$  задаётся кортежем:  $K_i = \langle buffer_i, limit = 20 \rangle$ , где  $buffer_i = \{x_{ij}(t - limit), \dots, x_{ij}(t)\}$  – скользящее окно измерений, обновляемое операцией  $update(K_i, x_{ij})$ . Конфигурация:  $C_2 = \{global, scrape_configs, ports, K_i(limit)\}$ , где  $global = \{\Delta_s = 10s, \Delta_e = 10s\}$ , применяется с помощью операции  $apply: C_2 \rightarrow Prometheus$ .

**2.4. Анализ и проверка ( $A_3$ ).** Функция реализует механизмы обнаружения аномалий в потоке данных, сочетая классические статистические подходы и алгоритмический анализ временных рядов. Выбор методов обусловлен спецификой IoT-данных [15-16]. Анализ включает использование: квантильного коридора, пороговое сравнение на основе статистических границ и спектральный анализ.

Метрики, сформированные в  $A_2$ , периодически (с интервалом  $\Delta_s = 10s$ ) поступают в Prometheus, где с помощью правил проверяются условия срабатывания оповещений. При выполнении одного из условий формируется событие, которое передаётся на следующий этап для маршрутизации. Кортеж функции  $A_3$  описывается следующими элементами:  $I_3 = Met \cup \{W_i\}$  – входные метрики и временные буферы,  $O_3 = A$  – множество обнаруженных аномалий,  $C_3 = \{alert_rules.yml\}$  – правила оповещения,  $M_3 = \{Alertmanager, Quantile, Threshold, FFT\}$  – механизмы анализа и реагирования. Для каждого момента времени  $t_i$  формируется скользящее окно:  $W_i = \{x_{kj} \mid t_k \in [t_i - \Delta t, t_k], k = i - max_i, \dots, i\}$ , содержащее  $max_i$  значений

$x_{kj}$ . В методе  $Threshold \in M_3$  используются фиксированные границы  $x_{min}$  и  $x_{max}$ , на основе длительного периода наблюдений и нормативных требований к микроклимату технических помещений. Значение  $x_{ij}$  – аномальное,  $x_{ij} < x_{min}$  или  $x_{ij} > x_{max}$ .

Для выявления отклонений в потоке данных применяется квантильный метод  $Quantile \in M_3$ . Для каждого скользящего окна вычисляются эмпирические квантили  $q_{0.05}(W_i)$  и  $q_{0.95}(W_i)$ . Практическая апробация показала, что статические заданные границы не способны адаптироваться к суточным и сезонным изменениям параметров среды.

Выбор уровней квантилей 0.05 и 0.95 обусловлен результатами анализа плотности распределения исторических данных (рис. 3).

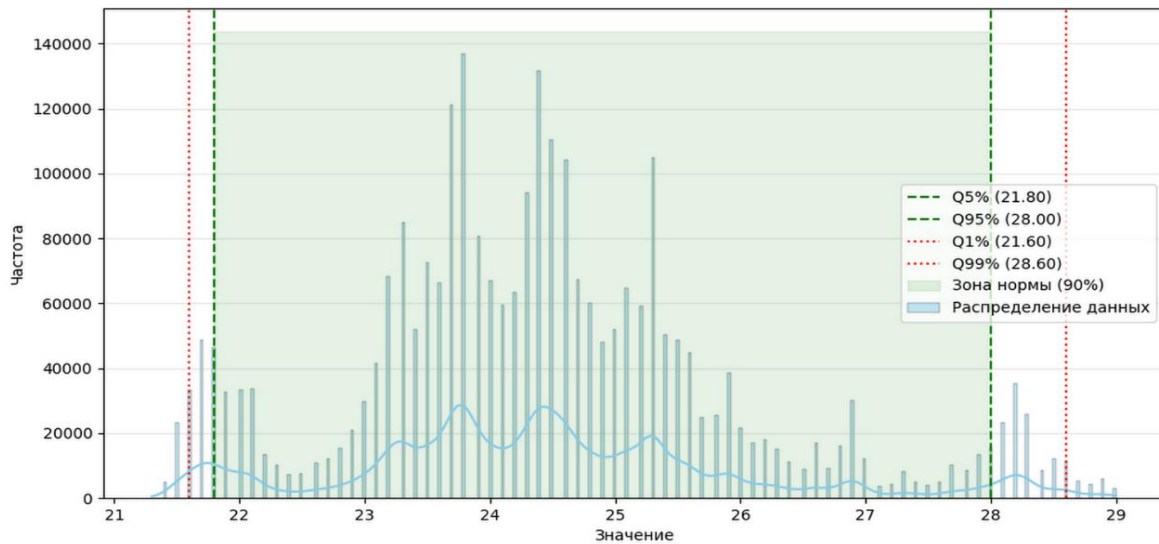


Рис. 3. Статические пороги  $q_{0.01}/q_{0.99}$  и  $q_{0.05}/q_{0.95}$

Выявлено, что при использовании порогов ( $q_{0.01}/q_{0.99}$ ) система игнорирует существенные, но кратковременные нарушения суточного цикла. При расширении интервала до  $q_{0.10}/q_{0.90}$  увеличивается количество ложных срабатываний. Точка  $x_{ij}$  классифицируется как аномалия, если  $x_{ij} < q_{0.05}(W_i)$  или  $x_{ij} > q_{0.95}(W_i)$ .

Спектральный метод  $FFT \in M_3$  заключается в применении к скользящему окну  $W_i$  быстрого преобразования Фурье  $F(W_i) = FFT(\{x_{kj}\}_{k: t_k \in [t_i - max_i, t_i]})$ . Обоснование параметров метода выполнено на основе ретроспективного анализа 3,1 млн наблюдений (2024–2025 гг.), выявившего устойчивую двухчастотную структуру сигнала: низкочастотный компонент (суточный цикл, ~24 ч) и высокочастотный (цикл климат-контроля, 5.6–7.1 мин). Статистический анализ подтвердил значительную сезонную вариативность энергетического вклада суточной гармоник (разброс >14%: от 18.6% зимой до 32.6% летом) и нестабильность периода работы оборудования, что исключает использование фиксированных порогов.

Вариативность энергетического профиля и нестабильность периодов работы оборудования обосновывают необходимость внедрения адаптивного подхода к формированию границ нормы спектра. Пороговые значения  $\eta_f$  для каждой частоты  $f_k$  определяются динамически, как 95-й процентиль распределения амплитуд:  $\eta_f(t_i) = q_{0.95}(\{|F(W_j)(f_k)|\}_{j=i-N}^{i-1})$  накопленного эталонного спектра, сформированного из исторического буфера длиной  $N$  предыдущих скользящих окон. Размеры скользящих окон определены с учётом выявленной периодичности: малое окно ( $N_1 = 120$  отсчетов, ~20 мин) для фиксации резких изменений и расширенное окно ( $N_2 = 540$  отсчетов, ~90 мин) для статистически устойчивой оценки гармоник без эффекта спектрального размывания.

Точка  $x_{ij}$  считается аномальной, если амплитуда на частоте  $f_k$  превышает заданный порог:  $|F(W_i)(f_k)| > A_k$ , где  $A_k$  – пороговое значение для частоты  $f_k$ . Физически это соответствует резкому количественному доминированию конкретных гармоник над эталонным профилем, сформированным в предыдущие периоды наблюдения, что интерпретируется как появление непериодического скачка или нарушение циклического режима работы оборудования, отличного от обычных процессов.

**2.5. Интерфейс в Telegram (A<sub>4</sub>).** Функция реализует канал взаимодействия с пользователем через мессенджер Telegram. Разработан бот на основе Telegram Bot API, интегрированный с Alertmanager. Бот позволяет выполнять отправку push-уведомлений при обнаружении аномалий и обработку пользовательских запросов (например, команда /info для получения графика за заданный период). Alertmanager настроен для маршрутизации событий с использованием токена бота и идентификатора чата. Инструменты визуализации выполнены в Python и передаются в чат в виде изображений.

Кортеж функции  $A_4$  представляет:  $I_4 = A \cup Ev$  – аномалии и события,  $O_4 = Resp$  – уведомления и графики,  $C_4 = \{bot\_token, alert\_rules.yml\}$  – конфигурационные параметры,  $M_4 = \{AlertManager, TelegramAPI\}$  – программные механизмы для выполнения правил и взаимодействия с пользователем. Обработка запросов формализуется функцией:  $proc: Ev \cup Req \rightarrow Resp$ , где  $Req = \{A_u = \langle type_u, period_u \rangle$  – множество пользовательских запросов. Графики формируются, как:  $plot: Met_{A_u(periode)} \rightarrow Image$ , а уведомления, как:  $push: Req \rightarrow Msg$ , в соответствии с правилами с правилами  $C_4(Req)$ .

**3. Механизм контроля живучести системы.** Для обеспечения надёжности работы реализован механизм контроля живучести компонентов в реальном времени – HeartBeat. Он выполняет генерацию запросов с периодической проверкой активности IoT-датчиков и других узлов инфраструктуры на основе заданных в Prometheus метрик.

Рассмотрим  $H = \{h_1, h_2, \dots, h_n\}$  – множество компонентов системы (включая датчики). Состояние каждого компонента определяется как:  $state(h_k) \in \{active, in\_active\}$ . Механизм контроля реализован через alert\_rules.yml. Пример правила приведён на Рис. 4.

```

- alert: Heartbeat_000DE0163B57
  expr: >
  (1) absent_over_time(mqtt_000DE0163B57_temperature_c[5m])
  (2) and
  (time() - process_start_time_seconds{job="prometheus"} > 600)
  (3) for: 2m
  (4) labels:
    severity: critical
    sensor_id: 000DE0163B57
  annotations:
    summary: "Нет данных от датчика 000DE0163B57"
    description: |
      **Статус:** Не отвечает более 5 минут,
      **Датчик:** 000DE0163B57

```

**Рис. 4.** Пример правила HeartBeat для IoT-датчика

Правило включает следующие элементы (см. рис. 4): (1) – проверка отсутствия метрики температуры в течение 5 минут, (2) – исключение ложных срабатываний в первые 10 минут после запуска Prometheus, (3) – задержка активации оповещения на 2 минуты для фильтрации кратковременных перебоев, (4) – метки для классификации и группировки правил.

Интервал проверки в 5 минут выбран на основе расчета максимальной скорости изменения данных в ситуации, где пиковое отклонение не превышает критическое значение безопасности. Таким образом, он обеспечивает гарантированное обнаружение выхода датчика из строя до момента наступления неконтролируемого перегрева.

Генерация запросов формализуется, как функция:  $gen: H \times \Delta t \rightarrow Q$ , где  $Q = \{q_m = \langle sender_m, receiver_m, timestamp_m, type_m \rangle\}$  – множество служебных запросов,  $sender_m$  – узел Prometheus,  $receiver_m = h_{000DE0163B57}$  – целевой датчик,  $timestamp_m$  – время отправки,  $type_m = status$  – запрос статуса, а периодичность  $\Delta t$  синхронизирована с интервалом сбора метрик в Prometheus. Операция проверки определяется, как:

$$check(h_{000DE0163B57}) = \begin{cases} 1, & \text{если } mqtt_{000DE0163B57_{t_c}} \neq \emptyset \\ 0, & \text{если } mqtt_{000DE0163B57_{t_c}} = \emptyset \end{cases}$$

где условие означает, что метрика  $mqtt_{000DE0163B57_{t_c}}$  присутствует в окне  $[t - 5m, t]$ . При отсутствии метрики устанавливается  $state(h_{000DE0163B57}) = inactive$ , и генерируется аномалия  $a_h$ , которая инициирует оповещение пользователя.

**4. Внедрение инструментов мониторинга данных IoT-устройств.** Созданная система оповещения на основе данных IoT-устройств обеспечивает непрерывный мониторинг и анализ параметров микроклимата в реальном времени, а также своевременное выявление аномалий и информирование пользователей. Первая версия системы была развёрнута на облачной платформе, функционирует в штатном режиме и предназначена для последующей замены обновлённой реализацией. Тестирование обновлённой системы проводилось в рамках исследовательской IoT-среды [17], включающей четыре сенсорных устройства CL210-E, интегрированных через протокол MQTT.

Для оценки эксплуатационных характеристик был проведен мониторинг ресурсов, показавший, что средняя нагрузка на CPU составила 5,94%, а объем занимаемой оперативной памяти (RAM) стабилизировался на отметке 1014,74 МБ. Данные показатели подтверждают стабильность работы системы при интенсивной обработке потоков телеметрии и выполнении аналитических процедур в реальном времени. Динамика потребления ресурсов представлена на рисунке 5.

CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O
0.21%	157.5MiB / 15.52GiB	0.99%	42.9MB / 48.8MB	82.1MB / 0B
1.84%	135.1MiB / 15.52GiB	0.85%	101MB / 239MB	135MB / 0B
0.56%	104.3MiB / 15.52GiB	0.66%	20.6MB / 553MB	561MB / 160MB
0.02%	181.9MiB / 15.52GiB	1.14%	8.64kB / 21.6kB	130MB / 8.51MB
0.66%	164.5MiB / 15.52GiB	1.04%	52MB / 46.4MB	7.8GB / 2.34GB
4.66%	193MiB / 15.52GiB	1.21%	52.3MB / 26.8MB	17.4MB / 4.58MB
0.36%	22.54MiB / 15.52GiB	0.14%	14.4MB / 161MB	142MB / 582kB
1.23%	61.8MiB / 15.52GiB	0.39%	841MB / 54.4MB	1.47GB / 1.29GB

Рис. 5. Использование CPU, RAM

Расчётные затраты дискового пространства на поддержание базы данных при достижении объёма в 10 млн записей составляет 143.46 МБ. При интенсивности поступления данных порядка 34 560 пакетов в сутки, текущая конфигурация хранилища обеспечивает автономную работу системы в течении года без необходимости архивации данных. При этом низкая загрузка CPU и стабильный объём RAM свидетельствуют о наличии вычислительного резерва для дальнейшего масштабирования системы.

Функционирование модуля  $A_1$  обеспечивает сбор данных и формирование временного ряда:  $X = \{x_{ij} \mid i = 1 \dots 4, j = 1 \dots 7\}$ , где  $i$  – индекс датчика,  $j$  – индекс измеряемого параметра. Функции модуля  $A_2$  сохраняют метрики в базе данных MongoDB и в локальных буферах объёмом 120 и 540 значений для каждого параметра. За период тестирования было накоплено 5,3 млн записей общим объёмом 191,9 МБ.

Для верификации разработанного алгоритма было проведено сравнение автоматически выявленных аномалий с эталонной разметкой, выполненной экспертами на ретроспективных данных за 2024–2025 годы, которая составила 1370 эталонных событий. Анализ метрик качества: Precision (точность) = 0.877, Recall (полнота) = 0.991, F1-Score = 0.930 показал

способность метода детектировать практически все значимые сбои. Наличие 12 ложноотрицательных срабатываний объясняется различием в условиях анализа: эксперт при ретроспективной разметке обладает полным контекстом развития события во времени, тогда как алгоритм работает в режиме онлайн и фиксирует аномалию только по факту проявления явных признаков отклонения, что может приводить к небольшой задержке детекции на начальных стадиях развития инцидента. Снижение точности обусловлено наличием 182 ложноположительных срабатываний. На рисунке 6 приведён фрагмент временного ряда температуры и области, показывающие результаты классификации событий (истинно положительные срабатывания, совпадающие с разметкой, и ложно положительные, когда программа выдаёт сигнал об аномалии при отсутствии сбоя по мнению эксперта).

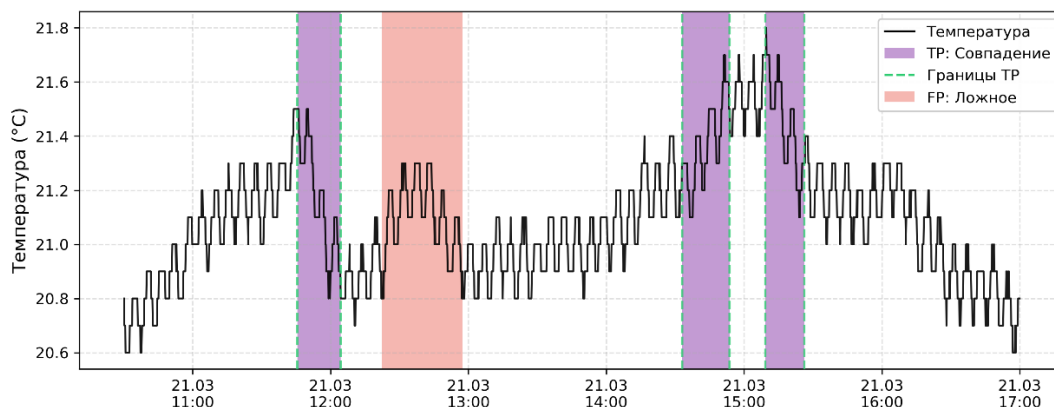


Рис. 6. Сравнение диагностики и экспертной разметки

Детальный анализ показал: ложноположительные результаты соответствуют нарушениям цикличности работы оборудования, которые не приводят к немедленному выходу параметров за аварийные пределы, но сигнализируют о нестабильности режима и требуют превентивного контроля, что в полной мере подходит для систем мониторинга.

Дополнительно применимость метода была подтверждена в ходе опытной эксплуатации системы мониторинга. За период внедрения было выявлено и верифицировано 11 реальных инцидентов, корректно классифицированных алгоритмом. Среди диагностированных неисправностей: отказы и разбалансировка работы кондиционирующих систем, обмерзание теплообменников внутренних блоков, а также дефекты датчиков. При обнаружении аномалий выполняется генерация оповещений на основе правил в Prometheus. Если условие сохраняется в течение 5 минут, событие передаётся в Alertmanager, который отправляет уведомление в настроенный Telegram-бот. Система реализует повторные оповещения по одному и тому же событию с периодичностью раз в 20 минут, при условии, что аномалия сохраняется дольше 5 минут. Для детального анализа данных выполняется подключение к инструментам визуализации Grafana.

**Заключение.** В работе реализован и верифицирован полный цикл создания автономной IoT-системы мониторинга микроклимата, охватывающий этапы от сбора телеметрии до формирования физически интерпретируемых оповещений о критических отклонениях. Разработанная архитектура обеспечивает модульное взаимодействие разнородных компонентов (сенсоры CL210-E, брокер Mosquitto, СУБД MongoDB, система мониторинга Prometheus) и поддерживает многоуровневый интерфейс взаимодействия: от автоматизированных push-уведомлений в Telegram до углубленного анализа через Grafana. Нагрузочное тестирование при обработке потока данных интенсивностью 34 560 пакетов в сутки и накоплении массива свыше 5,3 млн записей подтвердило стабильность работы системы и наличие значительного вычислительного резерва для дальнейшего масштабирования.

Ключевым результатом работы стало решение проблемы детекции аномалий в условиях выраженной нестационарности данных микроклимата. В отличие от статических пороговых методов и сложных моделей «черного ящика», в системе реализован гибридный подход, сочетающий спектральный анализ и адаптивные статистические коридоры. Предложенный механизм адаптации, основанный на динамическом пересчёте пороговых значений на основе амплитуд в скользящем историческом буфере, позволил автоматически подстраивать чувствительность системы под текущий режим эксплуатации, эффективно отсекая ложные срабатывания на штатные сезонные колебания. Опытная эксплуатация системы подтвердила её практическую ценность: за период тестирования были выявлены и верифицированы инциденты, связанные с отказом системы кондиционирования, их обмерзанием и дефектами датчиков. Созданная система представляет собой готовое к внедрению решение, способное функционировать в автономном режиме, адаптироваться к изменяющимся условиям среды и предоставлять персоналу прозрачные инструменты диагностики для предотвращения аварийных ситуаций.

**Благодарности.** Работа поддержана Красноярским математическим центром, финансируемым Минобрнауки РФ в рамках мероприятий по созданию и развитию региональных НОМЦ (Соглашение 075-02-2026-735).

#### Список источников

1. Довгаль В.А. Интернет Вещей: концепция, приложения и задачи / В.А. Довгаль, Д.В. Довгаль // Вестник Адыгейского государственного университета. Сер. Естественно-математические и технические науки, 2017. – № 4(211). – С. 140–147.
2. Kumar S., Tiwari P., Zymbler M. Internet of Things is a revolutionary approach for future technology enhancement: a review. *Journal of Big Data*, 2019, no. 6, p. 111, DOI: 10.1186/s40537-019-0268-2.
3. Исаева О.С. Формирование адаптивных рассылок брокера данных интернета вещей / О.С. Исаева, Н.В. Кулясов, С.В. Исаев // Информационно-управляющие системы, 2022. – № 5 (120). – С. 23-31. – DOI: 10.31799/1684-8853-2022-5-23-31.
4. Исаева О.С. Создание инструментов сбора данных для анализа аспектов безопасности Интернета вещей / О.С. Исаева, Н.В. Кулясов, С.В. Исаев // Информационные и математические технологии в науке и управлении. – 2022. – № 3(27). – С. 113-125. – DOI:10.38028/ESI.2022.27.3.011
5. Saputri F.R., Linelson R., Salehuddin M. et al. Design and development of an irrigation monitoring and control system based on blynk internet of things and thingspeak. *PLoS ONE*, 2025, vol. 20, no. 4, pp. 1-19, DOI:10.1371/journal.pone.0321250.
6. Spandonidis C., Tsantilas S., Sedikos E. et al. A compact, modular and low-cost Internet of Things (IoT) platform for air quality monitoring in urban areas. *Journal of Physics: Conference Series*, 2020, vol. 1710, no. 1, p. 012004, DOI:10.1088/1742-6596/1710/1/012004.
7. Yin hao L., Alqahtani A., Solaiman E. et al. IoT-CANE: A unified knowledge management system for data-centric Internet of Things application systems. *Journal of Parallel and Distributed Computing*, 2019, no. 131, pp. 161-172, DOI: 10.1016/j.jpdc.2019.04.016.
8. Табиев С. IoT технологии в системах мониторинга микроклимата помещений // С. Табиев, А. Орынбай // Вестник КазАТК, 2024. – № 136(1). – С. 299-307, DOI: 10.52167/1609-1817-2025-136-1-299-307.
9. Кычкин А.В. Разработка программной системы для управления IoT-устройствами с использованием структурных и поведенческих паттернов / А.В. Кычкин, О.В. Горшков // Прикладная информатика, 2020. – Т. 15. – № 4. – С. 35–44.
10. Gholamzadehmir M., Del Pero C., Buffa S. et al. Adaptive-predictive control strategy for HVAC systems in smart buildings – a review. *Sustainable Cities and Society*, 2020, vol. 63, p. 102480, DOI: 10.1016/j.scs.2020.102480.
11. Qais M.H., Kewat S., Loo K.H. et al. LSTM-based stacked autoencoders for early anomaly detection in induction heating systems. *Mathematics*, 2023, vol. 11, no. 15, p. 3319, DOI:10.3390/math11153319.
12. Lee Y., Park C., Kim N. et al. LSTM-autoencoder based anomaly detection using vibration data of wind turbines. *Sensors*, 2024, vol. 24, no. 9, p. 2833, DOI:10.3390/s24092833.
13. Sharma J., Mittal M.L., Soni G. Explainable artificial intelligence (XAI) enabled anomaly detection and fault classification of an industrial asset. *Research Square*, 2023, DOI:10.21203/rs.3.rs-2780708/v1.

14. Vasman S., Sukhanov I., Thalfeldt M. et al. Fault detection and diagnostics for HVAC systems using heat meter data, statistical features, and data mining. *Energy and Buildings*, 2026, art. no. 117343, DOI: 10.1016/j.enbuild.2026.117343.
15. Kurniawan H.M., Maulidan M., Zulmaulidin M.F. et al. Leveraging MongoDB in real-time emotion recognition from video for scalable and efficient data handling. *Engineering Proceedings*, 2025, vol. 107, no. 1, p. 84, DOI:10.3390/engproc2025107084.
16. Шкодырев В.П. Обзор методов обнаружения аномалий в потоках данных / В.П. Шкодырев, К.И. Ягафаров, В.А. Баштовенко // Proc. of the Second Conference on Software Engineering and Information Management. – Санкт-Петербург, 2017. – Т. 1864. – С. 7.
17. Исаева О.С. Инфраструктура сбора данных и имитации угроз безопасности сети интернета вещей / О.С. Исаева, Н.В. Кулясов, С.В. Исаев // Сибирский аэрокосмический журнал, 2025. – Т. 26. – № 1. – С. 8-20. – DOI: 10.31772/2712-8970-2025-26-1-8-20.

**Камышев Никита Львович.** Инженер отдела Красноярского математического центра Института вычислительного моделирования СО РАН. Область научных интересов: автоматизация процессов мониторинга, структуры систем передачи данных, аппаратура передачи данных, Cloud-Native и микросервисы, разработка десктопных приложений. ORCID: 0009-0008-4608-156X. [nik.novov@gmail.com](mailto:nik.novov@gmail.com).

**Исаева Ольга Сергеевна.** Доктор технических наук, ведущий научный сотрудник отдела Красноярского математического центра Института вычислительного моделирования СО РАН. Область научных интересов: методы искусственного интеллекта, анализ данных, цифровые двойники, информационная безопасность, интернет вещей. AuthorID: 165828, SPIN: 8412-5807, ORCID: 0000-0002-5061-6765, Researcher ID: A-8905-2018. [isaeva@icm.krasn.ru](mailto:isaeva@icm.krasn.ru). г. Красноярск, ул. Академгородок, д. 50, стр. 44, каб. 209.

UDC 004.822

DOI:10.25729/ESI.2026.42.2.016

## IoT device data monitoring tools

**Nikita L. Kamyshev, Olga S. Isaeva**

Institute of Computational Modelling SB RAS,

Russia, Krasnoyarsk, [nik.novov@gmail.com](mailto:nik.novov@gmail.com)

**Abstract.** This paper presents a solution for creating an autonomous microclimate monitoring system for technical facilities, capable of promptly detecting anomalies under conditions of non-stationary data and limited sensor infrastructure. The research aim is to develop monitoring tools that ensure continuous control, anomaly detection, and their interpretation. The core content of the work includes the development of a functional system model, the implementation of telemetry collection and heterogeneous data storage modules, and the deployment of an original hybrid algorithm for time series analysis. The key element is the proposed method, which combines spectral analysis to extract periodic signal components with adaptive statistical thresholds dynamically calculated based on sliding historical windows. Unlike existing counterparts relying on resource-intensive deep learning, the proposed approach utilizes physically grounded features to adapt to seasonal data variability without a preliminary training stage on labeled datasets. To unify the anomaly detection approach and alert routing, specific alert rules were configured, integration with a data visualization system was performed, and a chatbot was created to handle notifications and queries. To enhance fault tolerance, a component liveness control mechanism was implemented. The scientific novelty lies in proposing a hybrid adaptive anomaly detection method based on spectral-statistical time series analysis. The proposed approach ensures result interpretability by revealing the physical nature of failures through signal decomposition into harmonic components. Verification results on a real dataset of 5.3 million records confirmed the approach's effectiveness: detection recall reached 0.991 with a precision of 0.877. Load testing demonstrated the architecture's stability under low CPU utilization (<6%) and indicated reserves for scalability. The project successfully identified incidents related to air conditioning system malfunctions. The developed system provides a full cycle from data collection to preventive alerting, offering understandable diagnostic tools for preventing emergency situations while supporting two levels of interaction: a simplified interface for end-users and advanced analytical functions for specialists.

**Keywords:** Internet of Things (IoT), microclimate monitoring, anomaly detection, spectral analysis, adaptive thresholds, fault tolerance, Prometheus, AlertManager

**Acknowledgements:** 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. 075-02-2026-735).

## References

1. Dovgal V.A., Dovgal D.V. Internet veshchey: kontseptsiya, prilozheniya i zadachi [Internet of Things: concept, applications and tasks]. Vestnik Adygeyskogo gosudarstvennogo universiteta. Ser. Estestvenno-matematicheskiye i tekhnicheskiye nauki [Bulletin of Adyghe state university. Series: Natural, Mathematical and technical sciences], 2017, no. 4(211), pp. 140–147.
2. Kumar S., Tiwari P., Zymbler M. Internet of Things is a revolutionary approach for future technology enhancement: a review. Journal of Big Data, 2019, no. 6, p. 111, DOI: 10.1186/s40537-019-0268-2.
3. Isaeva O.S., Kulyasov N.V., Isaev S.V. Formirovaniye adaptivnykh rassylok brokera dannykh interneta veshchey [Formation of adaptive mailings of the Internet of Things data broker]. Informatsionno-upravlyayushchiye sistemy [Information and control systems], 2022, no. 5 (120), pp. 23-31, DOI: 10.31799/1684-8853-2022-5-23-31.
4. Isaeva O.S., Kulyasov N.V., Isaev S.V. Sozdaniye instrumentov sbora dannykh dlya analiza aspektov bezopasnosti Interneta veshchey [Creation of data collection tools for analyzing Internet of Things security aspects]. Informatsionnyye i matematicheskiye tekhnologii v nauke i upravlenii [Information and mathematical technologies in science and management], 2022, no. 3(27), pp. 113-125, DOI:10.38028/ESI.2022.27.3.011.
5. Saputri F.R., Linelson R., Salehuddin M. et al. Design and development of an irrigation monitoring and control system based on blynk internet of things and thingspeak. PLoS ONE, 2025, vol. 20, no. 4, pp. 1-19, DOI:10.1371/journal.pone.0321250.
6. Spandonidis C., Tsantilas S., Sedikos E. et al. A compact, modular and low-cost Internet of Things (IoT) platform for air quality monitoring in urban areas. Journal of Physics: Conference Series, 2020, vol. 1710, no. 1, p. 012004, DOI:10.1088/1742-6596/1710/1/012004.
7. Yin hao L., Alqahtani A., Solaiman E. et al. IoT-CANE: A unified knowledge management system for data-centric Internet of Things application systems. Journal of Parallel and Distributed Computing, 2019, no. 131, pp. 161-172, DOI: 10.1016/j.jpdc.2019.04.016.
8. Tabiyev S., Orynbay A. IoT tekhnologii v sistemakh monitoringa mikroklimata pomeshcheniy [IoT technologies in indoor microclimate monitoring systems]. Vestnik KazATK [Bulletin of KazATC], 2024, no. 136(1), pp. 299-307, DOI: 10.52167/1609-1817-2025-136-1-299-307.
9. Kychkin A.V., Gorshkov O.V. Razrabotka programmnoy sistemy dlya upravleniya IoT-ustroystvami s ispol'zovaniyem strukturnykh i povedencheskikh patternov [Development of a software system for managing IoT devices using structural and behavioral patterns]. Prikladnaya informatika [Applied Informatics], 2020, vol. 15, no. 4, pp. 35–44, DOI: 10.37791/2687-0649-2020-15-4-35-44.
10. Gholamzadehmir M., Del Pero C., Buffa S. et al. Adaptive-predictive control strategy for HVAC systems in smart buildings – a review. Sustainable Cities and Society, 2020, vol. 63, p. 102480, DOI: 10.1016/j.scs.2020.102480.
11. Qais M.H., Kewat S., Loo K.H. et al. LSTM-based stacked autoencoders for early anomaly detection in induction heating systems. Mathematics, 2023, vol. 11, no. 15, p. 3319, DOI:10.3390/math11153319.
12. Lee Y., Park C., Kim N. et al. LSTM-autoencoder based anomaly detection using vibration data of wind turbines. Sensors, 2024, vol. 24, no. 9, p. 2833, DOI:10.3390/s24092833.
13. Sharma J., Mittal M.L., Soni G. Explainable artificial intelligence (XAI) enabled anomaly detection and fault classification of an industrial asset. Research Square, 2023, DOI:10.21203/rs.3.rs-2780708/v1.
14. Vasman S., Sukhanov I., Thalfeldt M. et al. Fault detection and diagnostics for HVAC systems using heat meter data, statistical features, and data mining. Energy and Buildings, 2026, art. no. 117343, DOI: 10.1016/j.enbuild.2026.117343.
15. Kurniawan H.M., Maulidan M., Zulmaulidin M.F. et al. Leveraging MongoDB in real-time emotion recognition from video for scalable and efficient data handling. Engineering Proceedings, 2025, vol. 107, no. 1, p. 84, DOI:10.3390/engproc2025107084.
16. Shkodiyev V.P., Yagafarov K.I., Bashtovenko V.A. Obzor metodov obnaruzheniya anomalii v potokakh dannykh [Review of anomaly detection methods in data streams]. Proceedings of the Second Conference on Software Engineering and Information Management, Saint Petersburg, 2017, vol. 1864, p. 7.
17. Isaeva O.S., Kulyasov N.V., Isaev S.V. Infrastruktura sbora dannykh i imitatsii ugroz bezopasnosti seti interneta veshchey [Data collection and threat simulation infrastructure for Internet of Things network security]. Sibirskiy aerokosmicheskiy zhurnal [Siberian aerospace journal], 2025, vol. 26, no. 1, pp. 8-20, DOI: 10.31772/2712-8970-2025-26-1-8-20.

**Kamyshev Nikita Lvovich**, Engineer, Department of the Krasnoyarsk Mathematical Center, Institute of Computational Modelling SB RAS, Research interests: Automation of monitoring processes, data transmission system architectures, data transmission hardware, Cloud-Native and microservices, desktop application development. ORCID: 0009-0008-4608-156X. [nik.novov@gmail.com](mailto:nik.novov@gmail.com).

**Isaeva Olga Sergeevna**, Dr. Technical Sciences, Leading Researcher, Department of the Krasnoyarsk Mathematical Center, Institute of Computational Modelling SB RAS, Research interests: Artificial intelligence methods, data analysis, digital twins, information security, Internet of Things, AuthorID: 165828, SPIN: 8412-5807, ORCID: 0000-0002-5061-6765, Researcher ID: A-8905-2018. [isaeva@icm.krasn.ru](mailto:isaeva@icm.krasn.ru). Russia, Krasnoyarsk, Akademgorodok St., 50/44. Статья поступила в редакцию 15.10.2025; одобрена после рецензирования 24.03.2026; принята к публикации 25.05.2026.

*The article was submitted 10/15/2025; approved after reviewing 03/24/2026; accepted for publication 05/25/2026.*