
ЭКОНОМИКА И МЕНЕДЖМЕНТ СИСТЕМ УПРАВЛЕНИЯ

Научно-практический журнал

Основан в 2011 г.

**2026
№3(61)**

Издательство «Научная книга»



2026

Издательство "Научная книга"

Кафедра управления ВГТУ

Журнал зарегистрирован Управлением Федеральной службы по надзору в сфере связи,
информационных технологий и массовых коммуникаций по Воронежской области

ПИ N ТУ 36-00204 от 26 мая 2011 г.

ISSN 2223-0432

Журнал выходит один раз в квартал

ЭКОНОМИКА И МЕНЕДЖМЕНТ СИСТЕМ УПРАВЛЕНИЯ

Научно-практический журнал

Главный редактор – **Кравец О.Я.**, д-р техн. наук, профессор (Воронеж)

Зам. главного редактора – **Баркалов С.А.**, д-р техн. наук, профессор (Воронеж)

Ответственный секретарь – **Аверина Т.А.** (Воронеж)

РЕДАКЦИОННЫЙ СОВЕТ

Богатырёв В.Д., д-р экон. наук, профессор (Самара)

Бурков В.Н., д-р техн. наук, профессор (Москва)

Вертакова Ю.В., д-р экон. наук, профессор (Курск)

Владимирова И.Л., д-р экон. наук, профессор (Москва)

Гераськин М.И., д-р экон. наук, профессор (Самара)

Жанказиев С.В., д-р техн. наук, профессор (Москва)

Курочка П.Н., д-р техн. наук, профессор (Воронеж)

Остроух А.В., д-р техн. наук, профессор (Москва)

Перова М.Б., д-р экон. наук, профессор (Вологда)

Сибирская Е.В., д-р экон. наук, профессор (Орел)

Толстых Т.О., д-р экон. наук, профессор (Воронеж)

Черникова А.А., д-р экон. наук, профессор (Москва)

Чиркова М.Б., д-р экон. наук, профессор (Воронеж)

Дизайн обложки – **С.А.Кравец**

На основании заключения Президиума Высшей аттестационной комиссии Минобрнауки России журнал "Экономика и менеджмент систем управления" включен в Перечень российских рецензируемых научных журналов, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученых степеней доктора и кандидата наук.

Статьи, поступающие в редакцию, рецензируются. За достоверность сведений, изложенных в статьях, ответственность несут авторы публикаций. Мнение редакции может не совпадать с мнением авторов материалов. При перепечатке ссылка на журнал обязательна.

Правила для авторов доступны на сайте журнала <http://www.sbook.ru/emsu>



Адрес редакции и издателя:
394077 Воронеж, ул. 60-й Армии, д. 25, кв. 120

Тел. (473)2667653
E-mail: emsu@bk.ru
<http://www.sbook.ru/emsu>

Подписной индекс в объединенном каталоге «Пресса России» - 43054

Учредитель и издатель: ООО Издательство "Научная книга"
<http://www.sbook.ru>

Отпечатано с готового оригинал-макета в ООО "Цифровая полиграфия"

Адрес типографии: 394036, г.Воронеж, ул. Куколкина, 6, тел.: (473) 261-03-61

Подписано в печать 01.09.2026.

Свободная цена

Заказ 0000. Тираж 1000. Усл. печ. л. 6,2. Дата выхода в свет 09.10.2026.

№ Экономика и менеджмент систем управления, 2026

Содержание

Экономика и управление

Бекирова О.Н. Устойчивая интегрированная сеть прямой и обратной логистики: определение и постановка задачи.....	4
Кафтулина Ю.А. Цифровизация таможенного декларирования товаров в Российской Федерации: современное состояние, проблемы и перспективы развития.....	19
Сальникова А.В., Каузов А.И. Таможенное администрирование утилизационного сбора в Российской Федерации: проблемы и пути совершенствования	28

Информатика, вычислительная техника и управление

Аль-Имари М.Д.Б. Межуровневая схема координации, ориентированная на микросервисы	35
Баталов Д.И., Рындин А.А. Стратегии сетевого мониторинга для сетей критической инфраструктуры.....	45
Гетманская Д.В. Алгоритм выбора признаков больших наборов данных с высокой размерностью и несбалансированностью, основанный на гранулярном слиянии.....	64
Кравец О.Я. Проектирование алгоритма кластеризации больших наборов данных о конфиденциальности гетерогенной сети, функционирующего на базе облачных вычислений	74
Недикова Т.Н., Сергеев М.Ю. Применение аппарата нечеткой логики в задаче оценки кредитного риска.....	90

Правила для авторов	100
----------------------------------	------------

Экономика и управление

Бекирова О.Н.

УСТОЙЧИВАЯ ИНТЕГРИРОВАННАЯ СЕТЬ ПРЯМОЙ И ОБРАТНОЙ ЛОГИСТИКИ: ОПРЕДЕЛЕНИЕ И ПОСТАНОВКА ЗАДАЧИ

Воронежский государственный технический университет

1. Введение

В последние годы правительственные постановления и давление неправительственных организаций убедили корпорации учитывать вопросы устойчивого развития при принятии решений [1]. Хотя в литературе существует множество определений устойчивого развития, самое простое из них звучит так: «Использование ресурсов для удовлетворения текущих потребностей человека без пренебрежения способностью будущих поколений удовлетворять свои потребности» [2]. Ранние исследования устойчивого развития были направлены на решение экологических проблем. Но со временем в таких статьях, как [3], был принят подход «тройной нижней линии»; исследователи рассматривают экологические, экономические и социальные проблемы как основные факторы устойчивого развития. На рис. 1 показаны эти три нижние линии и их взаимосвязь в контексте устойчивого развития.

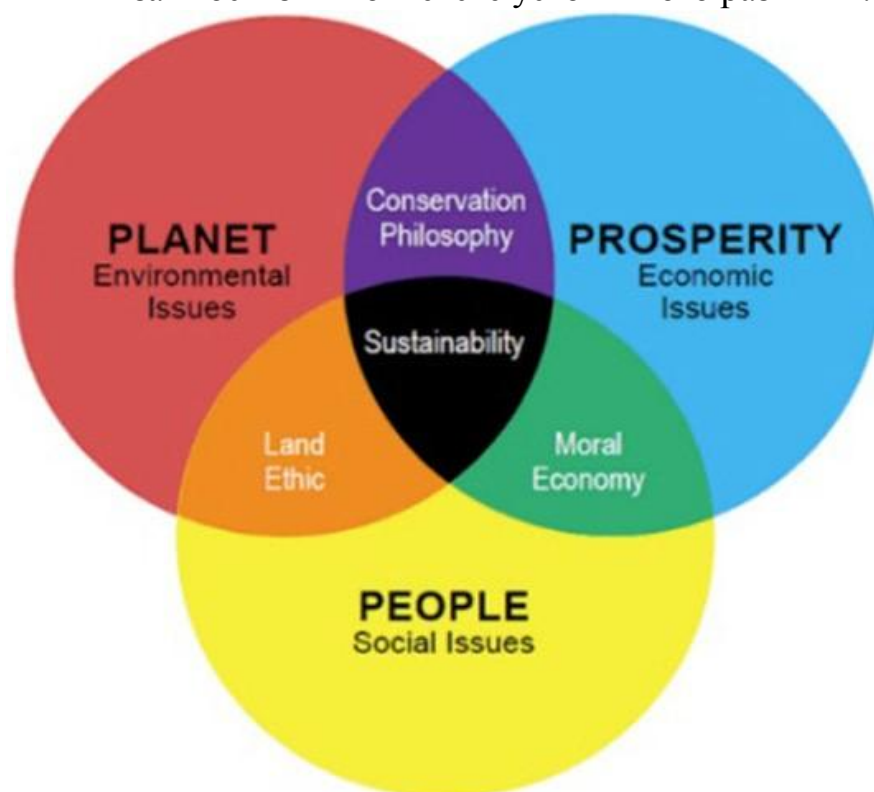


Рис. 1. Компоненты устойчивого развития и их взаимосвязи [3]

Управление операциями (УО) определяется как рациональное управление всеми операциями вдоль цепочки поставок, от закупки сырья до доставки готовой продукции клиентам. Одним из важнейших вопросов при проектировании устойчивой системы УО является концепция 3R (т.е. сокращение, повторное использование и переработка). Все большее внимание уделяется необходимости сокращения отходов, переработки и повторного использова-

ния продукции и деталей. Это известно как устойчивость в операциях [4]. Сокращение касается всех действий, направленных на снижение общей стоимости или количества отходов готовой продукции. Эти действия могут включать в себя исключение избыточной части продукта или даже изменение процесса производства изделия. Повторное использование может заключаться в сборе дефектной продукции у клиентов и эффективном устранении неисправных компонентов или деталей в процессе производства. Кроме того, корпорации могут перерабатывать продукцию и отправлять ее на рынок с меньшими затратами. Наконец, переработка состоит в сборе отходов, снижении их рисков с помощью предприятий по переработке и преобразовании их в пригодное для повторного использования сырье.

Обратная логистика - одна из важнейших составляющих операционного менеджмента, которая в последние годы привлекает внимание исследователей. В литературе существует множество различных определений обратной логистики. Например, в [5] обратная логистика определяется как процесс, в котором производитель систематически принимает ранее отгруженные продукты или детали из точки потребления для возможной переработки, восстановления и утилизации. За последние десятилетия многие компании инвестировали в деятельность по переработке и повторному использованию [6]. В [7] основные элементы возросшего внимания и инвестиций в обратную логистику разделены на две основные группы: экологические и деловые факторы. Экологические факторы включают вредное воздействие выброшенных товаров и материалов на окружающую среду, правила и законы о переработке и восстановлении продукции компании, срок службы которой подошел к концу, а также недостаточный доступ к ресурсам, необходимым для производства товаров. Эти экологические факторы побуждают исследователей обращаться к экологически чистым сетям для проектирования системы обратной логистики.

К факторам, влияющим на бизнес, относятся все аспекты обратной логистики, приносящие компании выгоду. К таким факторам относятся преимущества, получаемые за счет повторного использования материалов или деталей в производственных процессах, а также снижение уровня экологических преступлений, совершаемых государственными органами. Однако эти факторы не ограничиваются перечисленными выше преимуществами.

Конфигурация логистической сети, включая узлы и дуги между этими узлами, оказывает несколько воздействий на различные аспекты деятельности компании, такие как общая стоимость системы и уровень удовлетворенности клиентов [8]. Проектирование сети включает в себя различные решения, которые должны приниматься на разных уровнях управления, включая стратегический, оперативный и тактический. Решения о местоположении, количестве, объеме или мощности объектов, используемых технологиях и т. д. относятся к стратегическим решениям, которые оказывают долгосрочное влияние на эффективность компании. Некоторые другие решения, такие как планирование производства, маршруты обслуживания, составление графиков, выбор поставщиков и т. д., являются примерами оперативных и тактиче-

ских решений управления, которые должны приниматься часто и оказывают краткосрочное влияние на компанию. Исследователи [9], пришли к выводу, что раздельное рассмотрение этих решений без учета их влияния друг на друга приведет к созданию неоптимальной сети с точки зрения затрат, оперативности сети и некоторых других целей системы. Кроме того, прямую и обратную логистику следует рассматривать комплексно, поскольку обратная логистика оказывает большое влияние на эффективность прямой логистики и наоборот. Именно поэтому в сети могут быть доступны некоторые общие объекты для прямой и обратной логистики [10].

В литературе имеется множество исследований, посвященных проблеме проектирования цепочек поставок и логистических сетей. Эти исследования были рассмотрены в [11]. Следует отметить, что лишь немногие работы рассматривали инвестиции в экологические аспекты, и, насколько известно, нет работ, рассматривающих существующие технологии в объектах логистической сети. Статья [12] стала первым исследованием, в котором рассматривались инвестиции в экологически чистые аспекты на этапах проектирования цепочки поставок. Инвестиции в экологические факторы обеспечивают устойчивую конкурентоспособность корпораций, а небольшие затраты на экологически чистое развитие корпорации повышают ее позиции среди клиентов [13].

На основании изложенного в статье представлен проект многоцелевой интегрированной прямой и обратной логистики, в которой сеть включает производственные/перерабатывающие площадки, распределительные центры, зоны обслуживания клиентов, пункты сбора, предприятия по переработке и утилизации отходов. Помимо решений об открытии предприятий, лица, принимающие решения, должны учитывать уровень мощности каждого предприятия и уровень инвестиций в экологические аспекты деятельности предприятий. Несмотря на то, что во многих работах в этой области рассматривался только один уровень мощности предприятий, в данной статье рассматриваются несколько уровней мощности. Кроме того, впервые рассматривается использование экологически чистого автопарка на каждом предприятии. Экологичность сети исследуется путем оценки объема выбросов CO₂ во всей сети. Также минимизируются вредные последствия деятельности нежелательных предприятий для населенных пунктов.

2. Обзор литературы

В [14] был проведен всесторонний обзор моделей планирования сети обратной логистики. В своей работе авторы сосредоточились на таких факторах, как экологическое законодательство, экономические аспекты и корпоративная социальная ответственность. В [8] было указано, что большинство предложенных математических моделей для проектирования логистической сети (прямой, обратной или интегрированной) представляют собой модели смешанного целочисленного линейного программирования (MILP), которые часто используются в литературе и позволяют решать задачи с помощью коммерческого программного обеспечения для оптимизации [15]. Большинство этих исследований были направлены на снижение затрат или увеличе-

ние прибыли компании. Модели различаются по сложности: от простых моделей размещения одного продукта без ограничений по мощности [16] до многоуровневых моделей размещения нескольких продуктов с ограничениями по мощности [17].

Привлекая больше внимания к новым концепциям, таким как экологичная логистическая сеть, оперативность реагирования на спрос клиентов и устойчивость, были рассмотрены новые целевые функции в рамках многоцелевого проектирования логистической сети. В [18] представлена стохастическая многоцелевая математическая модель для проектирования прямой/обратной логистической сети. Авторы применили новый метод для определения систематической структуры цепочки поставок с целью повышения прибыли и увеличения оперативности реагирования на запросы клиентов в качестве целевой функции. Они применили метод эпсилон-ограничений для достижения точных решений Парето. Кроме того, колебания деловой среды и спроса клиентов усилили важность учета устойчивости при проектировании сети. В [19] представлена устойчивая и надежная модель для проектирования интегрированной прямой/обратной логистической сети. Авторы одновременно учли неопределенные параметры, такие как объем спроса клиентов и сбои в работе объектов.

«Зеленая» логистика относится ко всем процессам, связанным с производством и распределением экологически чистым способом; то есть, учитываются экологические проблемы. Важность «зеленой» логистики подтверждается тем фактом, что применяемые в настоящее время подходы к логистике в компаниях, предоставляющих логистические услуги, не являются устойчивыми в долгосрочной перспективе. В [20] «зеленая» логистика разделена на три основные категории, включая «Проблему экологически чистой маршрутизации транспортных средств» (GVRP), «Проблему маршрутизации с учетом загрязнения» и «Проблему маршрутизации транспортных средств в обратной логистике». GVRP рассматривает проблемы, в которых исследуется оптимизация энергопотребления [21]. «Проблема маршрутизации с учетом загрязнения», предложенная в [5], предполагает сокращение выбросов парниковых газов, в частности CO₂. Эти газы оказывают негативное воздействие на экосистемы и здоровье человека. В последнее время активизировались усилия по сокращению этих газов [22]. Последний тип «зеленой» логистики – это «Проблема маршрутизации транспортных средств в обратной логистике» (VPRL). В [23] было предложено следующее определение обратной логистики: «Процесс планирования, реализации и контроля обратных потоков сырья в производственном процессе, упаковке, готовой продукции, а также от точки производства, распределения или использования до точки восстановления или надлежащей утилизации». С учетом этой категоризации, «зеленая» логистика делится на три категории; переработка продукции также рассматривается как одна из этих категорий. В данной работе исследуются как управление переработкой, так и сокращение выбросов CO₂, которые являются двумя основными особенностями вопросов «зеленой» логистики. Другими словами, сокращение выбросов CO₂ и применение предложенной модели к интегриро-

ванной прямой и обратной логистике являются «зелеными» особенностями представленной в данной работе модели.

3. Многоцелевая оптимизация

Задачи одноцелевой оптимизации обычно направлены на достижение оптимального или рационального решения (решений). В этом отношении сравнение доступных решений в допустимой области является простым, поскольку у нас есть только одно значение целевой функции, и мы можем сортировать эти решения по их целевым значениям. Например, если задача является задачей максимизации, $f(x) < f(y)$ показывает, что решение y лучше, чем решение x . В последние годы исследователи сталкиваются с задачами, где доступны две или более противоречащих друг другу целевых функций. Такие задачи называются задачами многоцелевой оптимизации. В результате больше внимания уделяется экологическим и социальным вопросам в задачах оптимизации, и к традиционным целевым функциям добавляются новые цели [24]. Таким образом, в задачах многоцелевой оптимизации с противоречащими целями практически невозможно найти одно решение, которое одновременно является наилучшим для всех целей.

Решение, удовлетворяющее всем ограничениям, называется допустимым решением и состоит из ряда переменных решения. В многоцелевых задачах термин «оптимизация» означает поиск множества решений, удовлетворяющих всем ограничениям и дающих приемлемые значения для всех целевых функций. Этот термин определяется следующим образом [25]:

Если $X = [x_1, x_2, \dots, x_N]^T$ — вектор переменных решения, то цель состоит в том, чтобы найти такой вектор, как $X^* = [x_1^*; x_2^*; \dots; x_N^*]^T$, который удовлетворяет всем ограничениям и оптимизирует вектор целевой функции:

$$F = [f_1(x), f_2(x), \dots, f_m(x)] \quad (3.1)$$

$$g_i(x) \geq 0, i = 1, 2, \dots, p \quad (3.2)$$

$$h_j(x) = 0, j = 1, 2, \dots, q \quad (3.3)$$

Как уже упоминалось, найти решение, оптимизирующее все целевые функции, обычно невозможно. Таким образом, определяется понятие доминирующих решений. Рассмотрим общую задачу многоцелевой максимизации с n переменными решения и m целевыми функциями, в которой ($m > 1$), решение x доминирует над решением y , если выполняются следующие условия:

$$f_i(x) \geq f_i(y) \quad \forall i \in \{1, 2, \dots, m\} \quad (3.4)$$

$$f_i(x) > f_i(y) \quad \exists i \in \{1, 2, \dots, m\} \quad (3.5)$$

Если существует решение, которое не превосходит ни одно другое решение, оно называется недоминируемым решением. Все решения, которые не превосходят ни одно другое решение в допустимой области, называются Парето-оптимальными решениями. Более того, их соответствующие образы в пространстве целей называются Парето-границами.

4. Постановка задачи

Рассматриваемая в данной работе интегрированная сеть прямой/обратной логистики состоит из нескольких групп узлов, включая узлы производства/восстановления, узлы распределения, зоны обслуживания клиентов, пункты сбора, пункты переработки и пункты утилизации. Помимо ре-

шения о местоположении объектов, необходимо определить уровень мощности каждого существующего объекта в потенциальном месте. Рис. 2 иллюстрирует взаимосвязи между узлами. Новые продукты транспортируются из производственных центров в места расположения открытых распределительных центров. Между производственными центрами нет взаимодействия, и они работают независимо (т.е. транспортировка между производственными центрами равна нулю). Затем эти продукты распределяются между зонами обслуживания клиентов в соответствии с их спросом. Следует отметить, что местоположение зон обслуживания клиентов известно.

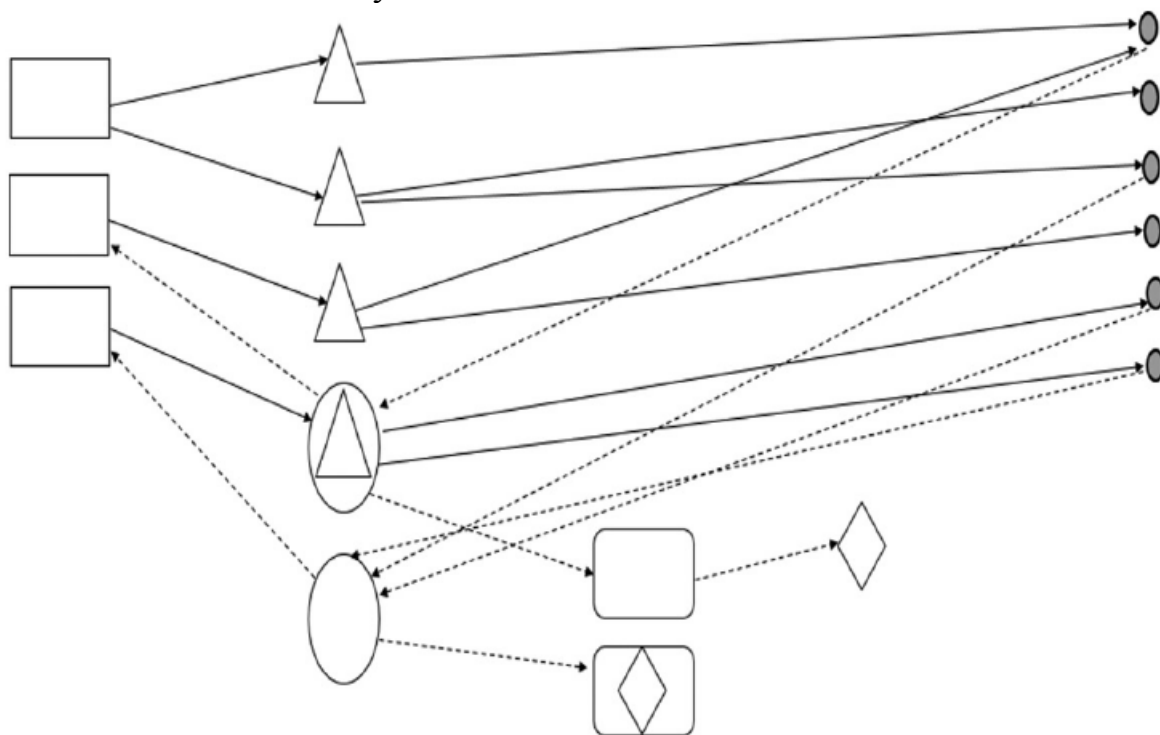


Рис. 2. Предлагаемая сеть: □ - производственный центр; ○ - центр сбора; □ - центр переработки отходов; △ - распределительный центр; ◇ - центр утилизации; - зона обслуживания клиентов; → - прямая логистика; - - - - - обратная логистика

На этом этапе начинается обратная логистика, и возвращенные продукты должны быть собраны из зон обслуживания клиентов в пункты сбора. Предполагается, что каждый клиент возвращает определенный процент продукции в качестве бракованной. Затем пригодные для восстановления продукты отправляются в производственные/перерабатывающие центры, после чего восстановленные продукты добавляются в прямую логистику. Таким образом, можно утверждать, что представленная нами логистическая сеть является замкнутой. Кроме того, пригодные для переработки и возвращенные продукты отправляются в центры переработки, а списанные — в центры утилизации. В процессе переработки также образуются отходы, которые необходимо транспортировать в центры утилизации. Другими словами, роль центров сбора заключается в тестировании и проверке для принятия решения о том, какое предприятие подходит для транспортировки возвращенных продуктов. В такой интегрированной сети объединение соответствующих цен-

тров обеспечивает экономию затрат для всей системы [26]. Для каждого типа возвращенных продуктов (т.е. пригодных для восстановления, перерабатываемых и списанных) устанавливается заранее определенный процент. Таким образом, в данной работе рассматриваются комбинированные объекты распределения, сбора и утилизации, а также комбинированная переработка; центры распределения, сбора, переработки и утилизации расположены в одном месте. Следовательно, экономия, полученная в результате этих комбинаций, должна быть отражена в целевой функции затрат.

Помимо вышеупомянутых проблем, в данной работе впервые рассматриваются некоторые новые концепции. Одним из главных аспектов исследования является принятие решения об уровне экологических инвестиций на этапе проектирования. Таким образом, нас интересует принятие экологических решений при проектировании сети и принятие мер по предотвращению загрязнения окружающей среды. Деятельность предприятий и компаний является одним из основных источников загрязнения окружающей среды. Кроме того, транспортная деятельность является значительным источником загрязнения воздуха и выбросов парниковых газов [27]. Следовательно, мы исследуем объем выбросов CO_2 в сети как единственный фактор воздействия на окружающую среду, который является очень популярным показателем. Учитывается объем выбросов CO_2 производственными, перерабатывающими и утилизационными центрами, а также выбросы CO_2 автопарком. В модель добавлена новая целевая функция, которая определяет объем выбросов CO_2 в сети и компромисс между общими затратами. Кроме того, учтены факторы воздействия на окружающую среду.

Кроме того, в данной работе рассматриваются социальные вопросы с учетом влияния нежелательных объектов, таких как центры переработки и утилизации отходов. Из-за нежелательного характера некоторых объектов в сети мы стремимся размещать их на значительном расстоянии от зон обслуживания клиентов. С другой стороны, корпорации могут принять решение повысить оперативность сети, разместив больше объектов вблизи зон обслуживания клиентов. Таким образом, представленная в данном исследовании сеть разработана с учетом одновременного учета стоимости сети, оперативности сети, а также экологических и социальных проблем.

Следует отметить, что все параметры, включая параметры, считаются детерминированными, и нам известно значение каждой переменной. Предсказание точного значения этих параметров представляет собой сложную задачу, и предсказание всегда сопряжено с ошибками. Предполагается, что спрос клиентов должен быть удовлетворен полностью. Кроме того, каждому узлу назначается только одно транспортное средство для перевозки всех грузов на каждом из существующих объектов. Проверяется баланс потоков в месте расположения каждого объекта; то есть, для каждого объекта необходимо учитывать баланс между входящими и выходными потоками. Наконец, для каждого существующего объекта учитывается ограничение по мощности, чтобы предотвратить превышение его возможностей. При этом каждый объект имеет не более одной установленной технологии и уровня мощности.

В целом, вопросы, на которые необходимо ответить в рамках этой сети, касаются местоположения и количества объектов, мощности каждого центра, технологий, внедренных в центрах производства, переработки и утилизации, уровня инвестиций в охрану окружающей среды путем приобретения экологически чистого автопарка для каждого центра, а также потока продукции между центрами. В табл. 1 приведено краткое описание проблемы.

Таблица 1

Атрибуты представленной сети

Цели	Выходы	Моделирование	Характеристики	Узлы сети
Расходы	Места расположения	MINLP	Емкость объектов	Производство/восстановление
Отзывчивость	Количество каждого объекта	MILP	Детерминированная модель	Распределение
Отношение к окружающей среде	Технологии		Поток с ограниченной пропускной способностью	Клиенты
Социальные	Емкость		Один продукт	Коллекция
	Автопарк		Неопределенное количество каждого объекта	Переработка отходов
			Один период	Утилизация

Для формулировки предлагаемой математической модели устойчиво интегрированной сети прямой/обратной логистики используются следующие обозначения:

Множества:

I - потенциальные места для размещения производственных/перерабатывающих центров;

J - потенциальные места для размещения распределительных центров;

K - наборы клиентских зон;

M - потенциальные места для размещения центров сбора;

E - потенциальные места для размещения объединенных распределительных и сборных центров; $E \cap J$, $E \cap M$;

R - потенциальные места для размещения центров переработки отходов;

D - потенциальные места для размещения центров утилизации;

N - потенциальные места для размещения объединенных центров переработки и утилизации отходов; $N \cap R$, $N \cap D$;

L - набор технологий, внедренных на предприятиях;

C - уровни мощности;

V - набор уровней инвестиций в автопарк;

R' - множество действующих центров переработки отходов;

D' - множество действующих центров утилизации отходов;

A - множество всех узлов.

Параметры:

de_k - спрос клиента k;

pe_k - процент возвращенных товаров у клиента k ;

q - процент возвращаемой продукции, подлежащей восстановлению;

qr - процент перерабатываемых изделий, передаваемых в пункты сбора;

sc - процент образующихся отходов в центрах переработки;

fp_{il}^c - стоимость создания производственного/перерабатывающего центра в потенциальном месте i с уровнем мощности c и технологией l ;

fd_j^c - стоимость создания распределительного центра j с уровнем мощности c ;

fc_m^c - стоимость создания распределительного центра в точке m с мощностью c ;

fr_{rl}^c - стоимость создания центра переработки отходов в месте r с уровнем мощности c и технологией l ;

fs_{dl}^c - стоимость создания центра утилизации отходов в месте d с уровнем мощности c и технологией l ;

$D_e^{cc\Phi}$ - экономия затрат, связанная с созданием объединенного объекта для распределительного центра с уровнем мощности c и центра сбора с уровнем мощности c' в точке e ;

$RD_{nll}^{cc\Phi}$ - экономия затрат, связанная с созданием комбинированного объекта, включающего центр переработки отходов с уровнем мощности c , технология l , и центр утилизации отходов с уровнем мощности c' , технология l' , в месте расположения p ;

Tr_{aa}^v - стоимость транспортировки одной единицы продукции от узла a до узла a' транспортными средствами с уровнем экологических инвестиций v ;

em_{aa}^v - Выбросы CO_2 , образующиеся при транспортировке одной единицы продукции из узла a в узел a' транспортными средствами с уровнем экологических инвестиций v ;

car_i^c - мощность производственного/перерабатывающего центра i с уровнем мощности c ;

cad_j^c - мощность распределительного центра j с уровнем мощности c ;

cas_m^c - вместимость пункта сбора m при уровне вместимости c ;

car_r^c - вместимость центра переработки r с уровнем вместимости c ;

cas_d^c - вместимость центра утилизации d с уровнем вместимости c ;

veh_a^v - фиксированные затраты на использование автопарка с уровнем экологических инвестиций v в узле a ;

$Resf$ - ожидаемое время доставки продукции;

$Resr$ - ожидаемое время сбора или возврата товаров;

tdc_{jk}^v - 1, если время доставки между распределительным центром j и клиентом k с использованием автопарка v меньше $Resf$; 0 в противном случае;

tcc_{km}^v - 1, если время в пути между клиентом k и пунктом сбора m с автопарком v меньше $Resr$; 0 в противном случае;

ρ - весовой коэффициент для прямой оперативности; $(1-\rho)$ обозначает вес обратной логистики;

$dis1_{rk}$ - расстояние между центром переработки r и зоной обслуживания клиентов k ;

$dis2_{dk}$ - расстояние между центром утилизации d и зоной обслуживания клиентов k ;

emp^1_i - выбросы CO_2 при производстве одной единицы продукции на производственном центре i с использованием технологии l ;

emp^1_r - выбросы CO_2 при переработке одной единицы продукции на центре переработки r с использованием технологии l ;

emp^1_d - выбросы CO_2 при переработке одной единицы продукции на центре утилизации i с использованием технологии l .

Переменные принятия решения

$x_{aa\Phi}^v$ - количество продукции, отгружаемой из узла a в узел a' транспортным средством v ;

ur^c_{il} - 1, если производственный центр s с уровнем мощности s и технологией l создан в потенциальном месте i ; 0 в противном случае;

ud^c_j - 1, если распределительный центр s с уровнем мощности s расположен в потенциальном месте j ; 0 в противном случае;

usc^c_m - 1, если пункт сбора s с уровнем мощности s расположен в потенциальном месте m ; 0 в противном случае;

ur^c_{rl} - 1, если пункт переработки s с уровнем мощности s и технологией l расположен в потенциальном месте r ; 0 в противном случае;

us^c_{dl} - 1, если центр захоронения s с уровнем мощности s и технологией l создан в потенциальном месте d ; 0 в противном случае;

z^v_a - 1, если парк транспортных средств s с уровнем экологических инвестиций v применяется в узле a .

В соответствии с вышеприведенными обозначениями, многоцелевая математическая модель проектирования устойчиво интегрированной логистической сети может быть сформулирована следующим образом:

$$\begin{aligned} \text{Min } Z1 = & \sum_i \sum_l \sum_c (fp^x_{il} \times ur^c_{il}) + \sum_j \sum_c (fd^c_i \times ud^c_j) + \\ & + \sum_m \sum_c (fc^c_m \times usc^c_m) + \sum_r \sum_c \sum_l (fr^c_{rl} \times ur^c_{rl}) + \sum_d \sum_c \sum_l (fs^c_{dl} \times us^c_{dl}) + \\ & + \sum_e \sum_c \sum_{c\Phi} (DC^{cc\Phi}_e \times yd^c_e \times yc^{c\Phi}_e) - \sum_n \sum_l \sum_{l\Phi} \sum_c \sum_{c\Phi} (RD^{cc\Phi}_{nll\Phi} \times ur^c_{nl} \times yc^{c\Phi}_{nl\Phi}) + \\ & + \sum_v \sum_a (veh^v_a \times z^v_a) + \sum_v \sum_a \sum_{a\Phi} (tr^v_{aa\Phi} \times x^v_{aa\Phi}) \end{aligned} \quad (4.1)$$

$$\text{Max } Z2 = \frac{\sum_k \sum_j \sum_v (x^v_{jk} \times dc^v_{jk})}{\sum_k de_k} + (1 - r) \frac{\sum_m \sum_v (x^v_{km} \times cc^v_{km})}{\sum_k (de_k \times e_k)} \quad (4.2)$$

$$\text{Min } Z3 = \text{Min}_{r\Phi, l, c, d\Phi} (dis1_{r\Phi} \times ur^c_{r\Phi}; dis2_{d\Phi} \times ys^c_{d\Phi}) \quad (4.3)$$

$$\begin{aligned}
\text{Min } Z_4 = & \sum_{a \in A} \sum_{v \in V} (em_{aa}^v \times x_{aa}^v) + \sum_{c \in C} \sum_{i \in I} \sum_{l \in L} \sum_{e \in E} \{ \sum_{j \in J} \sum_{v \in V} x_{ij}^v \times \{ \sum_{e \in E} \sum_{m \in M} \sum_{v \in V} x_{mr}^v \times \sum_{r \in R} \sum_{l \in L} \sum_{d \in D} \sum_{v \in V} x_{rd}^v + \sum_{m \in M} \sum_{d \in D} \sum_{v \in V} x_{md}^v \} \} \\
& + \sum_{c \in C} \sum_{r \in R} \sum_{l \in L} \sum_{e \in E} \{ \sum_{m \in M} \sum_{v \in V} x_{mr}^v \times \sum_{r \in R} \sum_{l \in L} \sum_{d \in D} \sum_{v \in V} x_{rd}^v + \sum_{m \in M} \sum_{d \in D} \sum_{v \in V} x_{md}^v \} \\
& + \sum_{c \in C} \sum_{d \in D} \sum_{l \in L} \sum_{e \in E} \{ \sum_{r \in R} \sum_{v \in V} x_{rd}^v + \sum_{m \in M} \sum_{d \in D} \sum_{v \in V} x_{md}^v \}
\end{aligned} \quad (4.4)$$

при ограничениях

$$\sum_{v \in V} x_{jk}^v = de_k \quad " k \in K \quad (4.5)$$

$$\sum_{v \in V} z_a^v \leq 1 \quad " a \in A \quad (4.6)$$

$$\sum_{a \in A} x_{aa}^v \leq M z_a^v \quad " a \in A, v \in V \quad (4.7)$$

$$\sum_{v \in V} \sum_{m \in M} x_{km}^v = de_k re_k \quad " k \in K \quad (4.8)$$

$$\sum_{i \in I} \sum_{v \in V} x_{ij}^v = \sum_{k \in K} \sum_{v \in V} x_{jk}^v \quad " j \in J \quad (4.9)$$

$$\sum_{k \in K} \sum_{v \in V} x_{km}^v = \sum_{i \in I} \sum_{v \in V} x_{mi}^v + \sum_{r \in R} \sum_{v \in V} x_{mr}^v + \sum_{d \in D} \sum_{v \in V} x_{md}^v \quad " m \in M \quad (4.10)$$

$$\sum_{i \in I} \sum_{v \in V} x_{mi}^v = q \sum_{k \in K} \sum_{v \in V} x_{km}^v \quad " m \in M \quad (4.11)$$

$$\sum_{r \in R} \sum_{v \in V} x_{mr}^v = q r \sum_{k \in K} \sum_{v \in V} x_{km}^v \quad " m \in M \quad (4.12)$$

$$\sum_{d \in D} \sum_{v \in V} x_{rd}^v = s c \sum_{m \in M} \sum_{v \in V} x_{mr}^v \quad " r \in R \quad (4.13)$$

$$\sum_{j \in J} \sum_{v \in V} x_{ij}^v \leq \sum_{l \in L} \sum_{c \in C} (cap_i^c \times y_{il}^c) \quad " i \in I \quad (4.14)$$

$$\sum_{k \in K} \sum_{v \in V} x_{ik}^v \leq \sum_{c \in C} (cad_j^c \times y_{jd}^c) \quad " j \in J \quad (4.15)$$

$$\sum_{k \in K} \sum_{v \in V} x_{km}^v \leq \sum_{c \in C} (cac_m^c \times y_{cm}^c) \quad " m \in M \quad (4.16)$$

$$\sum_{m \in M} \sum_{v \in V} x_{mr}^v \leq \sum_{c \in C} \sum_{l \in L} (car_r^c \times y_{rl}^c) \quad " r \in R \quad (4.17)$$

$$\sum_{m \in M} \sum_{v \in V} x_{md}^v + \sum_{r \in R} \sum_{v \in V} x_{rd}^v \leq \sum_{c \in C} \sum_{l \in L} (cas_d^c \times y_{dl}^c) \quad " d \in D \quad (4.18)$$

$$\sum_{c \in C} \sum_{l \in L} y_{il}^c \leq 1 \quad " i \in I \quad (4.19)$$

$$\sum_{c \in C} y_j^c = 1 \quad " j \in J \quad (4.20)$$

$$\sum_c y_m^c = 1 \quad " \quad m \in M \quad (4.21)$$

$$\sum_c \sum_l y_{rl}^c \leq 1 \quad " \quad r \in R \quad (4.22)$$

$$\sum_c \sum_l y_{dl}^c \leq 1 \quad " \quad d \in D \quad (4.23)$$

$$x_{aa}^v \leq 0 \quad " \quad a, a \in A, v \in V \quad (4.24)$$

$$y_{rl}^c, y_{dj}^c, y_{cm}^c, y_{rl}^c, y_{dl}^c, z_a^v \in \{0, 1\} \quad (4.25)$$

$$" \quad i \in I, j \in J, m \in M, c \in C, r \in R, d \in D, l \in L, a \in A, v \in V$$

Целевая функция (4.1) минимизирует общие затраты на проектирование сети, включая затраты на создание объектов с учетом используемой технологии и уровня мощности, транспортные расходы между узлами, использование автопарка и экономию затрат, связанных с объединением центров распределения, сбора, переработки и утилизации. Целевая функция (4.2) максимизирует прямую и обратную скорость реагирования сети. Мы модифицируем целевую функцию, представленную в [3]. Целевая функция (4.3) используется для увеличения расстояния между нежелательными объектами (т.е. центрами переработки и утилизации) и зоной обслуживания клиентов путем максимизации минимального расстояния между ними. В [9] эта целевая функция была применена для учета социального воздействия нежелательных объектов. Целевая функция (4.4) рассчитывает количество выбросов CO₂ в сети с учетом выбросов CO₂ в производственных, перерабатывающих и центрах утилизации, а также выбросов CO₂, вызванных автопарком.

Ограничение (4.5) гарантирует удовлетворение спроса клиентов. Ограничения (4.6) и (4.7) гарантируют, что каждому узлу назначается только один тип транспортного средства, и вся загрузка каждого существующего объекта осуществляется исключительно одним типом транспортного средства. Количество возвращенной продукции, инициирующее обратную логистику, определяется в ограничении (4.8). Ограничения (4.9) - (4.13) учитывают поток между узлами и обеспечивают сбалансированность потока между узлами. Ограничения (4.14) - (4.18) являются ограничениями по мощности, которые также запрещают передачу единиц продукции, возвращенной продукции, восстанавливаемой продукции, продукции, подлежащей переработке, и списанной продукции на неоткрытые объекты. Ограничения (4.19) - (4.23) гарантируют, что каждый объект имеет не более одного установленного уровня технологии и мощности. Ограничения (4.24) и (4.25) определяют диапазон переменных решения.

Некоторые члены в этой математической формулировке являются нелинейными, поэтому первоначально предложенная модель - это смешанное целочисленное нелинейное программирование (MINLP). Чтобы избежать сложности такой модели, вышеупомянутая модель линеаризуется путем введения некоторых новых переменных и новых ограничений. Члены

$\prod_{e \in E} \prod_{c \in C} (DC_e^{cc\Phi} \times y_d^c \times y_c^{c\Phi})$ и $\prod_{n \in N} \prod_{l \in L} \prod_{c \in C} (RD_{nl}^{cc\Phi} \times y_{r_{nl}}^c \times y_{c_{nl}}^{c\Phi})$ в первой целевой

функции являются нелинейными, поскольку они представляют собой умножение двух бинарных переменных. Переформулируем целевую функцию следующим образом:

$$Var1_e^{cc\Phi} = y_d^c y_c^{c\Phi} \quad (4.26)$$

$$Var2_{nl}^{cc\Phi} = y_{r_{nl}}^c y_{c_{nl}}^{c\Phi} \quad (4.27)$$

$$Var1_e^{cc\Phi}, Var2_{nl}^{cc\Phi} \in \{0,1\} \quad \forall e \in E; n \in N; c, c\Phi \in C; l \in L \quad (4.28)$$

$$\begin{aligned} \text{Min } Z1 = & \prod_{i \in I} \prod_{l \in L} \prod_{c \in C} (fp_{il}^x \times y_{p_{il}}^c) + \prod_{j \in J} \prod_{c \in C} (fd_j^c \times y_d_j^c) + \\ & \prod_{m \in M} \prod_{c \in C} \{fc_m^c \times y_c^c\} + \prod_{r \in R} \prod_{c \in C} \prod_{l \in L} (fr_{rl}^c \times y_{r_{rl}}^c) + \prod_{d \in D} \prod_{c \in C} \prod_{l \in L} (fs_{dl}^c \times y_{s_{dl}}^c) - \\ & - \prod_{e \in E} \prod_{c \in C} (DC_e^{cc\Phi} \times Var1_e^{cc\Phi}) - \prod_{n \in N} \prod_{l \in L} \prod_{c \in C} (RD_{nl}^{cc\Phi} \times Var2_{nl}^{cc\Phi}) + \\ & + \prod_{v \in V} \prod_{a \in A} (veh_a^v \times z_a^v) + \prod_{v \in V} \prod_{a \in A} \prod_{a\Phi \in A\Phi} (tr_{aa\Phi}^v \times x_{aa\Phi}^v) \end{aligned} \quad (4.29)$$

Поскольку целевая функция минимизируется, предполагается, что обе вспомогательные переменные должны принимать значение 1. Следовательно, мы не должны допускать, чтобы эти переменные принимали только значение 1, учитывая следующие условия: если y_d^c и $y_c^{c\Phi}$ равны 1, то $Var1_e^{cc\Phi}$ также должна быть равна 1, и те же условия должны применяться к $Var2_{nl}^{cc\Phi}$. Для учета вышеупомянутых условий добавим к модели следующие ограничения:

$$2Var1_e^{cc\Phi} \leq y_d^c + y_c^{c\Phi} \quad \forall e \in E; c, c\Phi \in C \quad (4.30)$$

$$2Var2_{nl}^{cc\Phi} \leq y_{r_{nl}}^c + y_{c_{nl}}^{c\Phi} \quad \forall n \in N; c, c\Phi \in C; l \in L \quad (4.31)$$

Кроме того, целевая функция (4.3) содержит нелинейный член, который можно линеаризовать, введя новую переменную $Var3$, чтобы исключить минимизацию члена в этой целевой функции. Переформулируем целевую функцию следующим образом, добавив два новых ограничения:

$$\text{Max } Z3 = Var3 \quad (4.32)$$

$$Var3 \leq dis1_{r\Phi} \times y_{r_{r\Phi}}^c \quad \forall r\Phi \in R; k \in K; l \in L; c \in C \quad (4.33)$$

$$Var3 \leq dis2_{d\Phi} \times y_{s_{d\Phi}}^c \quad \forall d\Phi \in D; k \in K; l \in L; c \in C \quad (4.34)$$

Наконец, четвертая целевая функция включает нелинейные члены, а именно произведение переменных решения. Например, в этой целевой функции член $y_{p_{il}}^c \prod_{j \in J} \prod_{v \in V} x_{ij}^v$ является нелинейным. Для устранения этой сложности

введем $Var4_{il}^c$ следующим образом:

$$Var4_{il}^c = y_{p_{il}}^c \prod_{j \in J} \prod_{v \in V} x_{ij}^v \quad (4.35)$$

$$\sum_{j=1}^n \sum_{v=1}^V x_{ij}^v \in \text{Var4}_{il}^c \in M \times \text{yp}_{il}^c \quad i \in I; l \in L; c \in C \quad (4.36)$$

В приведенных выше формулах M обозначает большое число. Если yp_{il}^c равно нулю, то $\sum_{j=1}^n \sum_{v=1}^V x_{ij}^v$ равно нулю, поэтому значение Var4_{il}^c будет равно нулю. И наоборот, если yp_{il}^c равно 1, Var4_{il}^c примет минимальное возможное значение; в результате Var4_{il}^c будет равно $\sum_{j=1}^n \sum_{v=1}^V x_{ij}^v$. Аналогичным образом,

другие нелинейные члены преобразуются в линейные следующим образом:

$$\text{Var5}_{rl}^c = \text{yp}_{il}^c \sum_{m=1}^M \sum_{v=1}^V x_{mr}^v \quad (4.37)$$

$$\sum_{m=1}^M \sum_{v=1}^V x_{mr}^v \in \text{Var5}_{rl}^c \in M \times \text{yp}_{rl}^c \quad r \in R; l \in L; c \in C \quad (4.38)$$

$$\text{Var6}_{dl}^c = \text{yp}_{dl}^c \sum_{r=1}^R \sum_{v=1}^V x_{rd}^v \quad (4.39)$$

$$\sum_{r=1}^R \sum_{v=1}^V x_{rd}^v \in \text{Var6}_{dl}^c \in M \times \text{yp}_{dl}^c \quad d \in D; l \in L; c \in C \quad (4.40)$$

$$\text{Var7}_{dl}^c = \text{yp}_{dl}^c \sum_{m=1}^M \sum_{v=1}^V x_{md}^v \quad (4.41)$$

$$\sum_{m=1}^M \sum_{v=1}^V x_{md}^v \in \text{Var7}_{dl}^c \in M \times \text{yp}_{dl}^c \quad d \in D; l \in L; c \in C \quad (4.42)$$

Целевая функция (4.4), преобразуется в следующее уравнение путем применения вышеуказанных преобразований:

$$\begin{aligned} \text{Min } Z_4 = & \sum_{a \in A} \sum_{a \in \Phi} \sum_{v=1}^V \left(\text{em}_{aa\Phi}^v \times x_{aa\Phi}^v \right) + \sum_{c \in C} \sum_{c \in \Phi} \sum_{v=1}^V \left(\text{emp}_i^l \times \text{Var4}_{il}^c \right) + \\ & + \sum_{c \in C} \sum_{r \in R} \sum_{l \in L} \left(\text{emr}_r^l \times \text{Var5}_{rl}^c \right) + \sum_{c \in C} \sum_{d \in D} \sum_{l \in L} \left(\text{emp}_d^l \times \left(\text{Var6}_{dl}^c + \text{Var7}_{dl}^c \right) \right) \end{aligned} \quad (4.43)$$

Список использованных источников

1. Rehman, S.T., Khan, S.A., Kusi-Sarpong, S. and Hassan, S.M. (2018), Supply chain performance measurement and improvement system: a MCDA-DMAIC methodology, *Journal of Modelling in Management*, Vol. 13 No. 3, pp. 522-549.
2. Beamon, B.M. (2008), Sustainability and the future of supply chain management, *Operations and Supply Chain Management*, Vol. 1 No. 1, pp. 4-18.
3. Ahi, P. and Searcy, C. (2013), A comparative literature analysis of definitions for green and sustainable supply chain management, *Journal of Cleaner Production*, Vol. 52, pp. 329-341.
4. Reid, R.D. and Sanders, N.R. (2011), *Operations Management an Integrated Approach*, John Wiley and Sons.
5. Dowlatshahi, S. (2000), Developing a theory of reverse logistics, *Interfaces*, Vol. 30 No. 3, pp. 143-155.
6. Üster, H., Easwaran, G., Akçali, E. and Çetinkaya, S. (2007), Benders decomposition with alternative multiple cuts for a multi-product closed-loop supply chain network design model, *Naval Research Logistics (Logistics)*, Vol. 54 No. 8, pp. 890-907.
7. Meade, L., Sarkis, J. and Presley, A. (2007), The theory and practice of reverse logistics, *International Journal of Logistics Systems and Management*, Vol. 3 No. 1, pp. 56-84.

8. Pishvae, M.S., Farahani, R.Z. and Dullaert, W. (2010), A memetic algorithm for bi-objective integrated forward/reverse logistics network design, *Computers and Operations Research*, Vol. 37 No. 6, pp. 1100-1112.
 9. Farrokhi-Asl, H., Tavakkoli-Moghaddam, R., Asgarian, B. and Sangari, E. (2016), Metaheuristics for a bi-objective location-routing-problem in waste collection management, *Journal of Industrial and Production Engineering*, pp. 1-14.
 10. Kramer, R., Subramanian, A., Vidal, T. and Lucídio dos Anjos, F.C. (2015), A metaheuristic approach for the pollution-routing problem, *European Journal of Operational Research*, Vol. 243 No. 2, pp. 523-539.
 11. Eskandarpour, M., Dejax, P., Miemczyk, J. and Péton, O. (2015), Sustainable supply chain network design: an optimization-oriented review, *Omega*, Vol. 54, pp. 11-32.
 12. Wang, F., Lai, X. and Shi, N. (2011), A multi-objective optimization for green supply chain network design, *Decision Support Systems*, Vol. 51 No. 2, pp. 262-269.
 13. Hanss, D., Böhm, G., Doran, R. and Homburg, A. (2016), Sustainable consumption of groceries: the importance of believing that one can contribute to sustainable development, *Sustainable Development*, Vol. 24 No. 6.
 14. Chanintrakul, P., Coronado Mondragon, A.E., Lalwani, C. and Wong, C.Y. (2009), Reverse logistics network design: a state-of-the-art literature review, *International Journal of Business Performance and Supply Chain Modelling*, Vol. 1 No. 1, pp. 61-81.
 15. Demirel, N.Ö. and Gökçen, H. (2008), A mixed integer programming model for re-manufacturing in reverse logistics environment, *The International Journal of Advanced Manufacturing Technology*, Vol. 39 Nos 11/12, pp. 1197-1206.
 16. Cornuéjols, G., Nemhauser, G.L. and Wolsey, L.A. (1983), *The Uncapacitated Facility Location Problem* (No. MSRR-493), Carnegie-Mellon Univ Pittsburgh Pa Management Sciences Research Group.
 17. Pasandideh, S.H.R., Niaki, S.T.A. and Asadi, K. (2015), Bi-objective optimization of a multi-product multi-period three-echelon supply chain problem under uncertain environments: NSGA-II and NREGA, *Information Sciences*, Vol. 292, pp. 57-74.
 18. Ramezani, M., Bashiri, M. and Tavakkoli-Moghaddam, R. (2013), A new multi-objective stochastic model for a forward/reverse logistic network design with responsiveness and quality level, *Applied Mathematical Modelling*, Vol. 37 No. 1-2, pp. 328-344.
 19. Hatefi, S.M. and Jolai, F. (2014), Robust and reliable forward–reverse logistics network design under demand uncertainty and facility disruptions, *Applied Mathematical Modelling*, Vol. 38 Nos 9/10, pp. 2630-2647.
 20. Lin, C., Choy, K.L., Ho, G.T., Chung, S.H. and Lam, H.Y. (2014), Survey of green vehicle routing problem: past and future trends, *Expert Systems with Applications*, Vol. 41 No. 4, pp. 1118-1138.
 21. Cirovic, G., Pamucar, D. and Božanic, D. (2014), Green logistic vehicle routing problem: routing light delivery vehicles in urban areas using a neuro-fuzzy model, *Expert Systems with Applications*, Vol. 41 No. 9, pp. 4245-4258.
 22. Demir, E., Bektas, T. and Laporte, G. (2014), The bi-objective pollution-routing problem, *European Journal of Operational Research*, Vol. 232 No. 3, pp. 464-478.
 23. Dekker, R., Moritz, F., Karl, I. and Luk, N.V.W. (Eds) (2013), *Reverse Logistics: Quantitative Models for Closed-Loop Supply Chains*, Springer Science and Business Media.
 24. Gamberi, M., Bortolini, M., Pilati, F. and Regattieri, A. (2015), Multi-objective optimizer for multimodal distribution networks: operating cost, carbon, Using Decision Support Systems for Transportation Planning Efficiency, *IGI Global*, p. 330.
 25. Rabbani, M., Farrokhi-Asl, H. and Asgarian, B. (2016), Solving a bi-objective location routing problem by a NSGA-II combined with clustering approach: application in waste collection problem, *Journal of Industrial Engineering International*, pp. 1-15.
 26. Mousazadeh, M., Torabi, S.A. and Zahiri, B. (2015), A robust possibilistic programming approach for pharmaceutical supply chain network design, *Computers and Chemical Engineering*, Vol. 82, pp. 115-128.
-

27. Fahimnia, B., Sarkis, J. and Davarzani, H. (2015), Green supply chain management: a review and bibliometric analysis, International Journal of Production Economics, Vol.162, pp. 101-114.

Кафтулина Ю.А.

ЦИФРОВИЗАЦИЯ ТАМОЖЕННОГО ДЕКЛАРИРОВАНИЯ ТОВАРОВ В РОССИЙСКОЙ ФЕДЕРАЦИИ: СОВРЕМЕННОЕ СОСТОЯНИЕ, ПРОБЛЕМЫ И ПЕРСПЕКТИВЫ РАЗВИТИЯ

Пензенский государственный университет

Введение. Современный этап развития мировой экономики характеризуется активной цифровой трансформацией государственного управления, затрагивающей практически все сферы таможенного администрирования. Одним из наиболее динамично развивающихся направлений является цифровизация таможенной деятельности, обеспечивающая повышение эффективности государственного контроля, ускорения движения товаров через таможенную границу и снижение административной нагрузки на участников внешнеэкономической деятельности.

В Российской Федерации цифровая трансформация таможенного администрирования реализуется в рамках комплексной модернизации Федеральной таможенной службы (ФТС России), предусматривающей внедрение современных информационных технологий, развитие электронного документооборота, совершенствование Единой автоматизированной информационной системы таможенных органов, расширение применения технологий автоматической регистрации и автоматического выпуска деклараций на товары.

Развитие цифровых технологий позволило практически полностью отказаться от бумажного документооборота при совершении таможенных операций, централизовать процессы оформления товаров посредством создания сети электронного декларирования, повысить уровень автоматизации обработки деклараций и существенно сократить время проведения таможенных операций. Вместе с тем дальнейшее совершенствование цифрового таможенного администрирования требует решения комплекса организационных, технологических и нормативно-правовых проблем, препятствующих переходу к полностью интеллектуальной модели принятия решений.

Несмотря на значительное количество исследований, сохраняется необходимость комплексной оценки эффективности цифровизации таможенного декларирования на современном этапе, анализа достигнутых результатов внедрения автоматизированных технологий, выявления существующих ограничений и определения перспектив дальнейшего развития цифровой модели таможенного администрирования.

Целью исследования является оценка эффективности и систематизация проблем цифровизации таможенного декларирования на основе анализа его современного состояния.

Научная новизна исследования заключается в оценке эффективности на цифровизации таможенного декларирования на основе анализа динамики основных показателей автоматизации таможенных процедур, а также в систе-

матизации проблем и разработке практических рекомендаций по дальнейшему развитию цифровой экосистемы таможенных органов.

Практическая значимость исследования определяется возможностью использования полученных результатов при совершенствовании деятельности ФТС России, развитии цифровых технологий таможенного контроля, а также при разработке мероприятий по повышению эффективности взаимодействия таможенных органов с участниками внешнеэкономической деятельности.

Результаты исследования. Современный этап развития российской таможенной системы характеризуется переходом от отдельных элементов автоматизации к комплексной цифровой модели таможенного администрирования. Основу данной модели составляет интеграция информационных ресурсов таможенных органов, электронного документооборота, интеллектуальных алгоритмов обработки данных и автоматизированных процедур принятия решений.

В настоящее время практически весь объем деклараций на товары представляется в электронной форме, что позволило отказаться от традиционного бумажного документооборота и создать единое цифровое пространство взаимодействия между таможенными органами и участниками внешнеэкономической деятельности. Использование электронного декларирования обеспечивает не только ускорение совершения таможенных операций, но и повышение достоверности представляемых сведений за счет применения автоматизированного форматно-логического контроля.

Ключевым элементом современной цифровой инфраструктуры выступает Единая автоматизированная информационная система таможенных органов (ЕАИС ТО), обеспечивающая централизованную обработку сведений, информационный обмен между структурными подразделениями ФТС России, а также взаимодействие с федеральными органами исполнительной власти и информационными системами государств - членов Евразийского экономического союза.

Существенное влияние на эффективность таможенного декларирования оказало создание сети центров электронного декларирования (ЦЭД) [9]. Концентрация полномочий по оформлению деклараций в специализированных подразделениях позволила унифицировать правоприменительную практику, повысить качество контроля и обеспечить более равномерное распределение нагрузки между таможенными органами.

Важнейшим направлением цифровизации стало внедрение технологий автоматической регистрации деклараций на товары и автоматического выпуска товаров. Автоматическая регистрация осуществляется без участия должностного лица на основе алгоритмической проверки полноты и корректности сведений, содержащихся в декларации. Автоматический выпуск представляет собой следующий этап цифровизации, при котором решение о выпуске товаров принимается информационной системой при отсутствии выявленных рисков и соблюдении установленных критериев.

Одновременно развивается автоматизированная система управления рисками, основанная на анализе больших массивов данных, статистических

моделей и профилей риска. Использование риск-ориентированного подхода позволяет сосредоточить контрольные мероприятия на наиболее рискованных поставках при минимальном вмешательстве в оформление добросовестных участников внешнеэкономической деятельности.

В последние годы существенное развитие получили технологии искусственного интеллекта, интеллектуального анализа данных и предиктивной аналитики (табл. 1). Их применение позволяет выявлять аномалии в товарных потоках, прогнозировать возможные нарушения таможенного законодательства и совершенствовать механизмы автоматизированного принятия решений.

Таблица 1

Основные цифровые технологии, применяемые при таможенном декларировании товаров [составлено по 1; 5; 7]

Цифровая технология	Основное назначение	Практический эффект
Электронное декларирование	Представление деклараций и документов в электронной форме	Полный отказ от бумажного документооборота
Автоматическая регистрация деклараций	Регистрация деклараций без участия инспектора	Сокращение времени регистрации, снижение нагрузки на должностных лиц
Автоматический выпуск товаров	Автоматизированное принятие решения о выпуске товаров	Ускорение таможенного оформления товаров низкого уровня риска
ЕАИС ТО	Централизованная обработка и хранение информации	Интеграция информационных ресурсов таможенных органов
Система управления рисками	Выборочный контроль поставок	Повышение эффективности таможенного контроля
Центры электронного декларирования	Централизация оформления деклараций	Унификация правоприменительной практики и повышение производительности
<i>Big Data</i> и искусственный интеллект	Анализ больших массивов данных и прогнозирование рисков	Развитие интеллектуального таможенного администрирования

Таким образом, современная модель цифрового таможенного декларирования представляет собой многоуровневую систему взаимосвязанных информационных технологий, обеспечивающих автоматизацию большинства административных процедур. Ее дальнейшее развитие связано с повышением уровня интеллектуализации процессов принятия решений, расширением автоматического выпуска товаров и интеграцией цифровых сервисов с информационными системами других государственных органов (табл. 2).

Проведенный анализ показывает, что цифровизация таможенного декларирования стала одним из наиболее успешных направлений модернизации государственного управления в Российской Федерации. Создание современной цифровой инфраструктуры позволило существенно повысить эффективность деятельности таможенных органов, обеспечить высокий уровень автоматизации основных процедур и сформировать предпосылки для дальнейше-

го перехода к интеллектуальной модели таможенного администрирования, основанной на технологиях искусственного интеллекта, предиктивной аналитики и цифровых платформ.

Таблица 2

Современные направления цифровизации таможенного декларирования в Российской Федерации [составлено автором по 2; 3; 4]

Направление цифровизации	Основные результаты	Значение для участников ВЭД
Электронное взаимодействие	Полный переход на электронное декларирование	Снижение административных издержек
Автоматизация таможенных операций	Развитие автоматической регистрации и автоматического выпуска	Сокращение времени оформления товаров
Централизация оформления	Развитие сети центров электронного декларирования	Повышение единообразия принимаемых решений
Межведомственная интеграция	Электронный обмен сведениями с государственными органами	Сокращение количества представляемых документов
Аналитические технологии	Использование Big Data и искусственного интеллекта	Повышение качества управления рисками
Развитие цифровых сервисов	Электронные кабинеты участников ВЭД, удаленная уплата платежей	Повышение удобства взаимодействия бизнеса с таможенными органами

Оценка эффективности цифровизации таможенного декларирования основывается на анализе количественных показателей, характеризующих уровень автоматизации таможенных процедур, скорость совершения таможенных операций и результативность внедрения современных информационных технологий. К числу наиболее значимых индикаторов относятся доля деклараций, зарегистрированных в автоматическом режиме, доля автоматически выпущенных деклараций, среднее время регистрации декларации и среднее время выпуска товаров.

Применение указанных показателей позволяет оценить не только уровень цифровой зрелости таможенной системы, но и эффективность функционирования Единой автоматизированной информационной системы таможенных органов, центров электронного декларирования и автоматизированной системы управления рисками.

За исследуемый период наблюдалось последовательное совершенствование цифровых технологий таможенного администрирования. Несмотря на отдельные колебания показателей в 2024 г., общая тенденция свидетельствует о повышении уровня автоматизации процессов, сокращении продолжительности таможенного оформления и расширении применения интеллектуальных алгоритмов обработки деклараций (табл. 3).

Анализ табл. 3 свидетельствует о поступательном развитии цифрового таможенного администрирования. Наиболее заметные положительные изменения произошли в сфере автоматической регистрации деклараций. Если в 2021 г. без участия должностных лиц регистрировалось 81,6% деклараций, то к 2025 г. данный показатель достиг 86,4% [10]. Это означает, что большинст-

во деклараций проходит первоначальную обработку исключительно средствами информационной системы, что значительно сокращает административную нагрузку на инспекторский состав и ускоряет начало таможенного оформления. По итогам 2025 г. автоматически зарегистрировано более 3,3 млн. деклараций из 3,88 млн. электронных деклараций на товары [10].

Таблица 3

Динамика основных показателей цифровизации таможенного декларирования в Российской Федерации за 2021-2025 г. [составлено автором по 6; 8]

Показатель	Годы				
	2021	2022	2023	2024	2025
Автоматическая регистрация деклараций, %	81,6	83,0	85,0	82,0	86,4
Автоматический выпуск товаров, %	26,5	26,8	27,0	26,0	27,9
Среднее время регистрации декларации, мин.	7	6	5	5	<5
Среднее время выпуска товаров, ч.	5-6	5	4,5	4	3-4

Несколько более медленными темпами развивается технология автоматического выпуска товаров. В течение рассматриваемого периода доля автоматически выпущенных деклараций увеличилась с 26,5 до 27,9%. Несмотря на сравнительно небольшие темпы роста, данный показатель отражает постепенное расширение возможностей автоматизированного принятия решений при сохранении необходимого уровня таможенного контроля. Ограниченный рост объясняется тем, что решение о выпуске товаров связано с анализом рисков, контролем соблюдения запретов и ограничений, а также проверкой правильности определения таможенной стоимости и классификации товаров.

Одним из наиболее значимых результатов цифровизации стало сокращение продолжительности совершения таможенных операций. Среднее время регистрации декларации уменьшилось с 7 мин. в 2021 г. до менее 5 мин. в 2025 г., а среднее время выпуска товаров сократилось с 5-6 ч до 3-4 ч [8]. Для поставок низкого уровня риска время автоматического выпуска может составлять менее двух минут, что свидетельствует о высокой эффективности современных цифровых технологий.

Для оценки интенсивности цифровой трансформации целесообразно рассмотреть изменение основных показателей за весь исследуемый период (табл. 4).

Таблица 4

Изменение показателей эффективности цифровизации таможенного декларирования за 2021-2025 г. [составлено автором по 6; 8]

Показатель	2021	2025	Абсолютное изменение	Темп изменения
Автоматическая регистрация, %	81,6	86,4	+4,8 п.п.	+5,9 %
Автоматический выпуск, %	26,5	27,9	+1,4 п.п.	+5,3 %
Среднее время регистрации, мин	7,0	4,5	-2,5 мин	-35,7 %
Среднее время выпуска, ч	5,5	3,5	-2,0 ч	-36,4 %

Расчеты показывают, что наиболее существенные изменения связаны с сокращением временных затрат при совершении таможенных операций. Снижение среднего времени регистрации деклараций почти на 36% обуслов-

лено совершенствованием алгоритмов автоматической проверки сведений, развитием электронного документооборота и оптимизацией функционирования Единой автоматизированной информационной системы таможенных органов.

Сопоставимые результаты наблюдаются и по времени выпуска товаров. За пять лет продолжительность оформления сократилась более чем на треть, что существенно повысило скорость движения товаров через таможенную границу и снизило издержки участников внешнеэкономической деятельности.

Следует отметить, что динамика автоматической регистрации значительно опережает темпы роста автоматического выпуска. Подобная ситуация объясняется различием содержания рассматриваемых процедур. Регистрация декларации представляет собой преимущественно формальную проверку соблюдения требований к заполнению документа и комплектности представленных сведений. Автоматический выпуск, напротив, требует комплексной оценки рисков, анализа профилей участников внешнеэкономической деятельности, проверки соблюдения мер нетарифного регулирования и иных контрольных мероприятий. Именно поэтому доля автоматически зарегистрированных деклараций существенно превышает показатель автоматического выпуска.

Временное снижение отдельных показателей в 2024 г. нельзя рассматривать как свидетельство ухудшения эффективности цифровизации. Оно обусловлено изменением структуры внешнеторговых потоков, расширением перечня контролируемых товарных категорий, а также адаптацией информационных систем к новым условиям функционирования. Уже в 2025 г. наблюдалось восстановление положительной динамики: доля автоматической регистрации достигла максимального значения за исследуемый период, а автоматический выпуск увеличился до 27,9% всех оформленных деклараций.

В целом проведенный анализ свидетельствует о высокой эффективности проводимой цифровой трансформации таможенного декларирования. За 2021-2025 г. была сформирована современная цифровая инфраструктура, обеспечивающая практически полный переход на электронное взаимодействие с участниками внешнеэкономической деятельности, широкое применение технологий автоматической регистрации деклараций и последовательное расширение автоматического выпуска товаров. Одновременно значительно сократились сроки совершения таможенных операций, повысилась производительность таможенных органов и были созданы предпосылки для дальнейшего внедрения технологий искусственного интеллекта, предиктивной аналитики и интеллектуальной системы управления рисками.

Несмотря на значительные результаты цифровой трансформации таможенного администрирования, современная система таможенного декларирования продолжает сталкиваться с рядом проблем, ограничивающих дальнейшее повышение эффективности автоматизированных процедур. Эти проблемы имеют технологический, организационный, нормативно-правовой и кадровый характер, что требует комплексного подхода к их решению.

Одной из наиболее существенных проблем является зависимость эффек-

тивности таможенного декларирования от стабильности функционирования информационной инфраструктуры. Сбои в работе Единой автоматизированной информационной системы таможенных органов, нарушения каналов передачи данных и временная недоступность отдельных электронных сервисов способны приводить к увеличению сроков совершения таможенных операций и росту финансовых издержек участников внешнеэкономической деятельности.

Другой актуальной проблемой остается сравнительно невысокая доля автоматического выпуска товаров. Несмотря на высокий уровень автоматической регистрации деклараций, окончательное решение о выпуске значительной части товаров по-прежнему принимается должностными лицами таможенных органов. Это обусловлено необходимостью проведения дополнительных проверок, применением системы управления рисками, контролем соблюдения запретов и ограничений, а также проверкой правильности классификации товаров и определения их таможенной стоимости.

Определенные трудности связаны и с различиями в уровне цифровой зрелости отдельных таможенных органов. Несмотря на функционирование центров электронного декларирования, отдельные регионы характеризуются неодинаковыми показателями автоматизации и различной скоростью обработки деклараций, что снижает единообразие правоприменительной практики и эффективность функционирования цифровой системы в целом.

Важным ограничением дальнейшей цифровизации является необходимость совершенствования нормативно-правовой базы. Развитие технологий искусственного интеллекта, автоматизированного принятия решений и интеллектуального анализа данных требует определения правового статуса таких технологий, распределения ответственности за принимаемые автоматизированные решения и формирования единых требований к использованию цифровых инструментов в деятельности таможенных органов.

Дополнительными факторами, сдерживающими развитие цифрового таможенного администрирования, являются возрастающие угрозы информационной безопасности, необходимость импортозамещения программного обеспечения, дефицит квалифицированных специалистов в сфере информационных технологий, а также недостаточный уровень цифровой подготовки отдельных участников внешнеэкономической деятельности. Все перечисленные обстоятельства требуют дальнейшего совершенствования цифровой экосистемы Федеральной таможенной службы России.

Современные тенденции развития государственного управления свидетельствуют о том, что дальнейшая цифровизация таможенного декларирования будет связана не столько с увеличением количества электронных сервисов, сколько с повышением интеллектуального уровня информационных систем и расширением автоматизированного принятия решений.

Одним из наиболее перспективных направлений является развитие технологий искусственного интеллекта и машинного обучения при функционировании системы управления рисками. Использование интеллектуальных алгоритмов позволит повысить точность выявления потенциальных нарушений

таможенного законодательства, сократить количество ложных срабатываний профилей риска и увеличить долю деклараций, выпускаемых в автоматическом режиме.

Не менее важным направлением остается модернизация Единой автоматизированной информационной системы таможенных органов. Переход к современной микросервисной архитектуре, создание резервных центров обработки данных и дальнейшее развитие отечественного программного обеспечения будут способствовать повышению устойчивости цифровой инфраструктуры и снижению технологических рисков.

Значительный потенциал имеют развитие межведомственного электронного взаимодействия и формирование единого цифрового пространства в рамках Евразийского экономического союза. Расширение электронного обмена сведениями между таможенными органами, налоговыми органами, Россельхознадзором, Роспотребнадзором и другими государственными структурами позволит минимизировать объем документов, представляемых участниками внешнеэкономической деятельности, и ускорить проведение контрольных мероприятий.

Перспективным направлением является дальнейшее совершенствование деятельности центров электронного декларирования. Использование интеллектуальной диспетчеризации деклараций с учетом текущей загрузки центров позволит более равномерно распределять информационные потоки, повысить производительность обработки деклараций и обеспечить единообразие правоприменительной практики.

Особое значение приобретает развитие цифровых сервисов для участников внешнеэкономической деятельности. Внедрение интеллектуальных помощников при заполнении деклараций, расширение функциональных возможностей личного кабинета участника ВЭД, интеграция информационных систем бизнеса с Единой автоматизированной информационной системой таможенных органов посредством программных интерфейсов (API) будут способствовать снижению количества ошибок при декларировании и дальнейшему сокращению сроков совершения таможенных операций.

Комплексная реализация указанных направлений позволит сформировать интеллектуальную цифровую модель таможенного администрирования, основанную на автоматизированном анализе данных, риск-ориентированном контроле и минимальном участии должностных лиц при совершении стандартных таможенных операций.

Заключение. Проведенное исследование показало, что цифровизация таможенного декларирования является одним из наиболее значимых направлений модернизации системы таможенного администрирования Российской Федерации. За последние годы создана современная цифровая инфраструктура, обеспечивающая практически полный переход на электронное декларирование товаров, широкое применение автоматической регистрации деклараций, развитие автоматического выпуска товаров и централизацию процессов оформления посредством центров электронного декларирования.

Анализ динамики показателей за 2021-2025 г. свидетельствует об устой-

чивом повышении уровня автоматизации таможенных процедур, сокращении времени регистрации деклараций и выпуска товаров, а также повышении эффективности деятельности таможенных органов. Достигнутые результаты подтверждают положительное влияние цифровых технологий на снижение административной нагрузки, ускорение движения товаров через таможенную границу и повышение качества государственного контроля.

Вместе с тем дальнейшее развитие цифрового таможенного администрирования требует решения ряда технологических, организационных и нормативно-правовых проблем, связанных с совершенствованием Единой автоматизированной информационной системы, расширением применения технологий искусственного интеллекта, повышением уровня информационной безопасности и дальнейшей модернизацией системы управления рисками.

В перспективе развитие цифровой модели таможенного декларирования должно быть ориентировано на интеллектуализацию процессов принятия решений, расширение автоматического выпуска товаров, развитие межведомственного информационного взаимодействия и создание единой цифровой экосистемы таможенного администрирования. Реализация указанных направлений позволит обеспечить дальнейшее повышение эффективности деятельности таможенных органов, укрепление экономической безопасности государства и создание благоприятных условий для развития внешнеэкономической деятельности Российской Федерации.

Список использованных источников

1. Бондаренко А.О. Тенденции развития организационной структуры таможенных органов в условиях перехода к цифровой таможне// Академический вестник Ростовского филиала Российской таможенной академии. 2021. № 3. С. 18-25.
2. Кафтулина Ю.А. Информационные технологии в системе таможенного контроля: современная практика противодействия обороту контрафактной продукции// Экономика и управление: проблемы, решения. 2025. Т. 1, № 11 (164). С. 46-55.
3. Кафтулина Ю.А., Пальченков Д.А., Нагорнова Д.А. Особенности оптимизации таможенных процедур декларирования товаров в условиях цифровизации// Экономика и управление: проблемы, решения. 2026. Т. 12, № 2 (167). С. 233-241.
4. Кузякин Ю.П., Михин Ю.П. Цифровая трансформация таможенного контроля: состояние и перспективы// Юридическая наука. 2023. № 8. С. 111-117.
5. Орел М.Н. Развитие цифровых платформ для оптимизации таможенных операций// Экономические науки. 2024. № 239. С. 152-156.
6. <https://customs.gov.ru>.
7. Протасов М.Н. Состояние и перспективы развития современной системы показателей оценки процесса оказания государственных таможенных услуг// Вестник евразийской науки. 2024. Т. 16, № 2. - <https://esj.today/PDF/39ECVN224.pdf>.
8. <https://movi-st.ru/company/articles/sroki-tamozhennogo-oformleniya-v-2025-godu-polnyy-obzor-vremennykh-ramok/>.
9. <https://axillc.ru/news/novinki/tsentralnaya-elektronnaya-tamozhnya-itogi-2025-418-tysdt-i-452-mlrd-rub/>.
10. <https://ed2inteh.ru/фмс-итоги-декларирования-в-2025-году/>.

Сальникова А.В., Каузов А.И.

ТАМОЖЕННОЕ АДМИНИСТРИРОВАНИЕ УТИЛИЗАЦИОННОГО СБОРА В РОССИЙСКОЙ ФЕДЕРАЦИИ: ПРОБЛЕМЫ И ПУТИ СОВЕРШЕНСТВОВАНИЯ

**Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевич Столетовых**

Введение

Утилизационный сбор, введенный в Российской Федерации в 2012 г., представляет собой один из ключевых инструментов таможенного регулирования, позволяющий компенсировать снижение таможенных пошлин после вступления страны во Всемирную торговую организацию [24]. За прошедшие годы утилизационный сбор претерпел значительные изменения: ставки неоднократно повышались, совершенствовался порядок расчета и взимания, расширялся круг плательщиков. С 1 октября 2024 г. произошло повышение коэффициентов на 70-85%, а с 1 декабря 2025 г. вступили в силу новые правила расчета, привязывающие размер сбора к мощности двигателя [26].

В 2024 г. поступления утилизационного сбора составили рекордные 1085,9 млрд руб., что в 1,65 раза превысило показатель 2023 г. [11]. Однако в 2025 г. наблюдается существенное снижение фактических поступлений: бюджет недополучил около 890 млрд руб. по сравнению с плановыми показателями свыше 2 трлн руб. [25]. Это свидетельствует о достижении предела фискальной емкости инструмента и обуславливает необходимость критического переосмысления механизмов администрирования.

Цель статьи состоит в выявлении особенностей таможенного администрирования утилизационного сбора, выявлении проблем и определении основных направлений его совершенствования.

1. Правовая природа и история института утилизационного сбора в России

Согласно части 1 статьи 24.1 Федерального закона от 24.06.1998 № 89-ФЗ, утилизационный сбор представляет собой обязательный платеж, взимаемый при ввозе транспортных средств на территорию Российской Федерации или при их производстве на территории страны, который выполняет экологическую функцию [27]. Бюджетный кодекс РФ относит утилизационный сбор к числу неналоговых доходов, полностью поступающих в федеральный бюджет [2]. В Таможенном кодексе ЕАЭС утилизационный сбор не упоминается, поскольку не отнесен к категории таможенных платежей, что порождает регуляторный вакуум на союзном уровне.

М.С. Гордиенко анализирует утилизационный сбор как элемент системы неналоговых платежей, связанных с экологической деятельностью [6]. Е.В. Лунева и Л.Р. Мингазова делают вывод о двойственной правовой природе сбора, одновременно обладающего признаками налога и таможенного платежа [8, с. 181]. М.Б. Худжатов определяет утилизационный сбор как инструмент протекционистской политики, направленный на защиту отечественного автопрома [24]. С.Г. Белев и Е.О. Матвеев утверждают, что сбор осуществляет протекционистскую функцию, «хоть и прикрываемую необходимостью

соблюдения экологических стандартов» [1, с. 25]. Д.С. Власенко справедливо отмечает, что «действительная цель взимания утилизационного сбора выражается в реализации фискальных задач государства» [3, с. 8].

Изначально обязанность по уплате утилизационного сбора возлагалась исключительно на импортеров, а российские производители освобождались от уплаты при условии принятия обязательств по переработке автомобилей [9]. Однако такой подход вызвал протесты, и в ноябре 2012 г. Европейский союз обратился в ВТО с иском к России. В ответ на международное давление с 1 января 2014 г. сбор стал взиматься и с отечественных производителей. В 2016 г. произошло первое значительное повышение ставок на 65%, был введен сбор в отношении самоходных машин [8]. Период 2020-2024 г. ознаменовался масштабной индексацией: в 2020 г. коэффициенты выросли на 112-145%, а 1 октября 2024 г. произошло повышение на 70-85% [14]. С 1 декабря 2025 г. размер сбора привязан к мощности двигателя, запланировано ежегодное повышение на 10-20% до 2030 г. [4].

2. Порядок расчета, взимания и контроля уплаты утилизационного сбора

Утилизационный сбор исчисляется плательщиком самостоятельно путем умножения базовой ставки на повышающий коэффициент. Для легковых автомобилей категории М1 базовая ставка составляет 20000 руб., для прочих колесных транспортных средств - 150000 руб. [16], для самоходных машин - 172500 руб. [15]. Коэффициенты дифференцированы в зависимости от типа транспортного средства, возраста, объема и мощности двигателя. Для физических лиц предусмотрен льготный режим: 3400 руб. для транспортных средств до 3 лет и 5200 руб. для старше 3 лет при соблюдении ряда условий, включая ограничение мощности до 160 л.с. [13].

Администрирование утилизационного сбора на ввозимые транспортные средства осуществляет ФТС России, на транспортные средства отечественного производства - ФНС России. При ввозе из стран ЕАЭС применяется компенсационная формула, включающая недоуплаченные таможенную пошлину, НДС и акциз [12]. Плательщик обязан уплатить сбор не позднее принятия решения о выпуске товара. Таможенный орган в течение 5 рабочих дней проверяет правильность исчисления. При выявлении неуплаты в течение 3 лет таможенный орган информирует плательщика о доначислении и пенях в размере 1/300 ключевой ставки ЦБ РФ за каждый день просрочки [16].

Анализ динамики поступлений сумм утилизационного сбора в федеральный бюджет РФ в 2012-2025 г. позволяет выявить четыре периода: становление (2012-2014 г.), кризис и восстановление (2015-2016 г.), стабильный рост (2017-2022 г.), взрывной рост (2023-2024 г.). Совокупные поступления за исследованный период превысили 4 трлн руб. (табл. 1). За 13 лет структура администрирования утилизационного сбора претерпела полный цикл от монополии ФТС в 2012-2013 г. через устойчивое доминирование ФНС в 2015-2022 г. обратно к доминированию ФТС в 2023-2024 г. (табл. 1).

В 2021-2024 г. суммы поступлений пени за просрочку уплаты утилизационного сбора выросли в 59,5 раз (табл. 2), причем доминирование пени за

колесные транспортные средства (71,7%) свидетельствует о концентрации проблем в наиболее массовом сегменте.

Таблица 1

Динамика поступлений утилизационного сбора в федеральный бюджет РФ в 2012-2024 г.

Год	ФТС России, млрд руб.	ФНС России, млрд руб.	Всего, млрд руб.	Доля ФТС России, %	Доля ФНС России, %
2012	18,7	-	18,7	100,0	0,0
2013	49,5	-	49,5	100,0	0,0
2014	43,7	58,8	102,5	42,6	57,4
2015	22,6	62,1	84,7	26,7	73,3
2016	47,3	89,8	137,1	34,5	65,5
2017	24,7	139,3	164,0	15,1	84,9
2018	84,4	178,8	263,2	32,1	67,9
2019	93,4	225,8	319,2	29,3	70,7
2020	97,2	267,1	364,3	26,7	73,3
2021	138,1	337,9	476,0	29,0	71,0
2022	136,3	225,0	361,3	37,7	62,3
2023	448,8	210,1	658,9	68,1	31,9
2024	668,8	417,1	1 085,9	61,6	38,4
Итого	1 873,5	2 211,8	4 085,3	-	-

Примечания: а) с 2012 - данные только о колесных транспортных средствах (шасси) и прицепах к ним, ввозимых в Российскую Федерацию; б) с 2014 - данные только о колесных транспортных средствах (шасси) и прицепах к ним, ввозимых в Российскую Федерацию и производимых на территории РФ; в) с 2016 - данные о самоходных машинах и прицепах к ним, ввозимых в Российскую Федерацию и производимые на территорию Российской Федерации; источник: здесь и далее составлено автором по данным отчетов по форме 0507011 за 2012-2024 г. Федерального казначейства РФ [11]

3. Зарубежный опыт администрирования утилизационных платежей

В Европейском союзе Директива 2000/53/ЕС заложила основы системы расширенной ответственности производителя. Ключевая особенность - отсутствие прямого сбора с покупателей: ответственность за финансирование утилизации возлагается на производителей и импортеров, обязанных обеспечить бесплатный прием утилизируемых транспортных средств. В Германии действует залоговая система (200-300 евро), в Нидерландах - сбор в размере 45 евро со специализированным оператором, во Франции экологические взносы включаются в цену автомобиля [7]. Япония обладает одной из наиболее совершенных систем. Закон об утилизации автомобилей (вступил в силу 1 января 2005 г.) вводит систему предварительной оплаты: размер сбора устанавливается индивидуально для каждой модели (6000 - 18000 йен), средства управляются некоммерческой организацией и направляются на переработку конкретных категорий отходов [22]. Уникальной особенностью является электронная система учета, обеспечивающая полную прослеживаемость на всех этапах. В отличие от России, японская система гарантирует прямое целевое использование средств [29]. Китай взимает сбор с производителей и импортеров, в Республике Корея применяется субсидиарная модель, США не

имеют федеральной системы утилизационного сбора [28].

Сравнительный анализ выявляет четыре модели: расширенной ответственности производителя (ЕС, Япония), фискально-протекционистская (Россия, Китай), субсидиарная (Республика Корея) и рыночная (США). Наиболее эффективные системы строятся на прозрачности, прослеживаемости и прямой связи между уплаченными сборами и экологическими результатами.

Таблица 2

Динамика поступлений сумм пени за просрочку уплаты сумм утилизационного сбора за транспортные средства, ввозимые в РФ, поступивших в федеральный бюджет РФ в 2019-2024 г.

Год	Пени за просрочку уплаты суммы утилизационного сбора за ввозимые в РФ колесные транспортные средства, млн руб.	Пени за просрочку уплаты суммы утилизационного сбора за ввозимые в РФ самоходные машины, млн руб.	Всего пени, млн руб.	Темп роста суммы пени за просрочку уплаты суммы утилизационного сбора за ввозимые в РФ колесные транспортные средства, %	Темп роста суммы пени за просрочку уплаты суммы утилизационного сбора за ввозимые в РФ самоходные машины, %
2019	7,0	17,1	24,1	-	-
2020	4,8	20,6	25,4	68,6	120,5
2021	1,3	11,3	12,6	27,1	54,9
2022	8,6	25,1	33,7	661,5	222,1
2023	81,0	47,5	128,5	941,9	189,2
2024	537,6	212,6	750,2	663,7	447,6
Итого	640,3	334,2	974,5	-	-

4. Проблемы таможенного администрирования утилизационного сбора

Первая проблема - нормативная фрагментарность правового регулирования. Порядок взимания регулируется множеством разрозненных нормативных правовых актов. Для одной категории транспортных средств применяются разные правила в зависимости от страны ввоза, цели использования и статуса лица. Таможенные органы при постконтроле нередко ориентируются на нормы, действовавшие на момент проверки, а не на момент возникновения обязанности, что равносильно обратной силе закона [20].

Вторая проблема - чрезмерная сложность расчета. Сумма сбора зависит от множества параметров. В 2025 г. ФТС обязала 30 тыс. автовладельцев, ввезших транспортные средства из Казахстана, доплатить сбор от 750 тыс. до 3,5 млн руб. из-за нарушения порядка применения формулы [23].

Третья проблема связана с доминирующей ролью постконтроля. Таможенные органы выявляют ошибки после уплаты сбора, выставя требования о доплате и пенях в течение 3 лет. Для плательщиков это создает ситуацию правовой неопределенности: обязанность формально исполнена, но окончательный размер платежа остается неопределенным. Генеральная прокуратура РФ выявила нарушения при контроле за полнотой исчисления сбора: к дон-

числению привела ненадлежащая проверка расчетов за транспортные средства, ввезенные физическими лицами [5].

Четвертая проблема - несвоевременная оценка действий при коммерческом ввозе под видом личного пользования. Отдельные физические лица декларируют автомобили для личного использования, тогда как фактически они перепродаются. Выявление таких нарушений происходит, когда транспортное средство уже поставлено на учет или перепродано до истечения 12 месяцев со дня выдачи таможенного приходного ордера [17-19].

Пятая проблема - функциональная перегрузка таможенных органов, выполняющих широкий круг функций от приема платежа до судебного взыскания. Действующий порядок исчисления слишком сложен, а организационная модель не соответствует масштабу платежа [21].

5. Предложения по совершенствованию механизма администрирования

Во-первых, необходимо принятие единого системного нормативного документа, например, постановления Правительства РФ с отменой множества действующих, консолидирующего все правила взимания, исчисления, уплаты, контроля, возврата и зачета сбора. Это устранил разрозненные правила и повысит предсказуемость.

Во-вторых, представляется целесообразным разработка и внедрение публичного официального автоматизированного калькулятора утилизационного сбора. Он должен обеспечивать: автоматический расчет на основе обязательного набора параметров; формирование шаблона расчета в формате PDF; интеграцию с реестрами Минпромторга РФ, Госавтоинспекции, Ростехнадзора; разъяснительную информацию с ссылками на законодательство; ведение журнала расчетов. Внедрение данного цифрового инструмента снизит количество ошибок плательщиков.

В-третьих, требуется переход к предварительному контролю расчета перед выпуском транспортного средства. Процедура включает: подачу расчета с декларацией, проверку таможенным органом в течение 5 рабочих дней, при обнаружении ошибки - возврат с указанием на ошибку, при отсутствии - подтверждение и выпуск. Это сместит акцент с выявления нарушений на их предупреждение.

В-четвертых, необходимо законодательное закрепление и цифровизация критериев транспортного средства для личного пользования, в частности, количественное ограничение ввоза (1-2 транспортных средства в год), обязательная регистрация на плательщика в течение 12 месяцев, отсутствие продажи в течение 12 месяцев, отсутствие множественных ввозов, отсутствие признаков коммерческого использования. Критерии должны применяться для автоматической проверки в информационной системе ФТС России.

В-пятых, следует перераспределить полномочия между таможенными органами и специализированным оператором. Предлагается делегировать оператору функции технического расчета, формирования электронного документа, ведения реестра и аудита. Таможенные органы сосредотачиваются на приеме платежа, предварительном и постконтроле, взыскании. Это снизит

административную нагрузку и повысит точность расчетов. Дополнительно необходима систематизация информационно-просветительской деятельности ФТС России и создание единого информационного портала.

Заключение

Утилизационный сбор в России - сложный правовой институт с расхождением между декларируемой экологической целью и фактически реализуемыми фискально-протекционистскими функциями. За 2012-2025 гг. поступления выросли в 58 раз, превысив 5,2 трлн руб. Однако снижение поступлений в 2025 г. свидетельствует о достижении предела фискальной емкости. Необходима комплексная модернизация системы администрирования с переходом от реактивной модели к профилактической и цифровизированной. Это позволит повысить правовую определенность, сократить споры, снизить нагрузку на таможенные органы и обеспечить устойчивое выполнение сбором его функций.

Список использованных источников

1. Белев С.Г., Матвеев Е.О. Последствия повышения утилизационного сбора для транспортных средств в РФ// *Ars Administrandi*. - 2022. - Т. 14, № 1. - С. 25-43.
 2. Бюджетный кодекс Российской Федерации от 31.07.1998 № 145-ФЗ (ред. от 25.05.2026). - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=535012&cacheid=C1FC68AB0DD1CA2300C05180A55445CC&mode=splus&rnd=0.8771844608510118#bnXWHNV3Ngd7hAV9>.
 3. Власенко Д.С. Проблемные вопросы правового регулирования утилизационного сбора в РФ// *Юрист ВЭД*. - 2025. - № 1. - С. 8-30.
 4. https://www.alta.ru/external_news/111632/.
 5. <https://epp.genproc.gov.ru/ru/gprf/mass-media/news/main/e8318604/?ysclid=mps6tjf3ga961629857>.
 6. Гордиенко М.С. Утилизационный сбор в системе фискальных неналоговых платежей РФ// *Налоги и налогообложение*. - 2019. - № 10. - С. 25-37.
 7. Есть ли такой сбор в других странах?// *Крымская правда*. - 18.10.2025.
 8. <https://tass.ru/info/7196227>.
 9. Кравцова С.А., Лобанова Н.Л. Утилизационный сбор: пополнение бюджета или обеспечение экологической безопасности// *Проблемы социально-экономического развития Сибири*. - 2019. - № 1 (5). - С. 59-64.
 10. Лунева Е.В., Мингазова Л.Р. Экологическая функция утилизационного сбора: проблемы правового регулирования// *Вестник Саратовской государственной юридической академии*. - 2018. - № 4 (123). - С. 174-182.
 11. https://roskazna.gov.ru/ispolnenie-byudzheto/federalnyj-byudzheto?filter_type=43&page=2.
 12. Письмо Министерства финансов РФ от 10.01.2025 № 03-13-06/617. - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=QUEST&n=229220&cacheid=D0A8D73A19DF009A9E3AAF8A84E9E652&mode=splus&rnd=0.8771844608510118#Ns3XHN VWXPXXeCHm>.
 13. Постановление Правительства РФ от 1.11.2025 № 1713 «О внесении изменений в Постановление Правительства Российской Федерации от 26.12.2013 г. № 1291». - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=517977&cacheid=598C8B2C4F4E48A876096F349941EA3D&mode=splus&rnd=0.8771844608510118#NOyXHNVC L8lguJjc>.
 14. Постановление Правительства РФ от 13.09.2024 № 1255 «О внесении изменений в некоторые акты Правительства Российской Федерации». - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=485724&cacheid=BC7>
-

DB4DABB470C384EB4AEE14836FEB1&mode=splus&rnd=0.8771844608510118#hmJYHN VZ7OFN9TeU.

15. Постановление Правительства РФ от 06.02.2016 № 81 (ред. от 12.02.2026) «Об утилизационном сборе в отношении самоходных машин и (или) прицепов к ним и о внесении изменений в некоторые акты Правительства Российской Федерации» (вместе с «Правилами взимания, исчисления, уплаты и взыскания утилизационного сбора в отношении самоходных машин и (или) прицепов к ним, а также возврата и зачета излишне уплаченных или излишне взысканных сумм этого сбора»). - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=526500&cacheid=B487B4F3D304FD3530A4F29FFA341450&mode=splus&rnd=0.8771844608510118#FOYYHNVl4gQWFTDF>.

16. Постановление Правительства РФ от 26.12.2013 № 1291 (ред. от 06.02.2026) «Об утилизационном сборе в отношении колесных транспортных средств (шасси) и прицепов к ним и о внесении изменений в некоторые акты Правительства Российской Федерации» (вместе с «Правилами взимания, исчисления, уплаты и взыскания утилизационного сбора в отношении колесных транспортных средств (шасси) и прицепов к ним, а также возврата и зачета излишне уплаченных или излишне взысканных сумм этого сбора»). - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=526497&cacheid=4DB9C6BE071337DBD6888D30CE19DF60&mode=splus&rnd=0.8771844608510118#dVyYHNV2PUy8L1U91>.

17. Решение Zubovo-Polyanskogo районного суда Республики Мордовия от 20.11.2024 по делу № 2а-908/2024 от 05.09.2024. - <https://reputation.su/sudrf/235481157>.

18. Решение Ленинского районного суда г. Махачкалы (Республика Дагестан) по делу № 2-2141/2025 от 08.05.2025. - <https://www.zakonrf.info/gorsud/doc-3fdeaf9e-cb7e-571e-82cc-88c47e2a3f4e/?ysclid=mpretad9mp56683204>.

19. Решение Октябрьского районного суда г. Липецка (Липецкая область) № 2А-1716/2024 2А-1716/2024~М-885/2024 М-885/2024 от 20.06.2024 по делу № 2А-1716/2024. - <https://sudact.ru/regular/doc/Z05uHexLs6Nc/?ysclid=mprfp8tg9n598714651>.

20. Решение Раменского городского суда Московской области по делу № 2а-2225/2025 от 22.05.2025. - <https://www.zakonrf.info/gorsud/doc-52af7041-4f7c-5db6-8201-1ddc9d055be1/?ysclid=mprfm3kmz8299104543>.

21. <https://customs.gov.ru/press/aktual-no/document/659555?ysclid=mprg4scpj6248436382>.

22. Стукалов Д. Утилизация и переработка вышедших из эксплуатации транспортных средств (ВЭТС) в Японии. - <https://rusmet.ru/press-center/utilizacziya-i-pererabotka-vyshedshih-iz-ekspluataczii-transportnyh-sredstv-vets-v-yaponii/>.

23. Харчевникова П. ФТС назвала причину доначислений за утильсбор на автомобили из Казахстана. - <https://www.rbc.ru/society/30/10/2025/69039bf99a794751a8e0af28>.

24. Худжатов М.Б. Актуальные проблемы уплаты утилизационного сбора при ввозе транспортных средств в Российскую Федерацию// Маркетинг и логистика. - Сентябрь-октябрь 2019. - № 5 (25). - <https://marklog.ru/aktualnye-problemy-uplaty-utilizacionnogo-sbora-pri-vvoze-transportnyh-sredstv-v-rossijskuju-federaciju/>.

25. <https://naavtotrasse.ru/auto-news/utillsbor-ne-srabotal-byudzheth-rf-nedoshchitalsya-09-trln.html>.

26. <https://www.garant.ru/article/1901763/>.

27. Федеральный закон от 24.06.1998 № 89-ФЗ (ред. от 23.03.2026) «Об отходах производства и потребления». - <https://www.consultant.ru/cons/cgi/online.cgi?req=doc&base=LAW&n=529662&cacheid=314C03DD8AB60439A7EF22CEFAFB2026&mode=splus&rnd=0.8771844608510118#lzEaHNVgBgLXvzmi1>.

28. <https://www.aea.or.kr/eng/biz/sub208.do>.

29. <https://autorecyclingworld.com/end-of-life-vehicle-recycling-a-comparative-analysis-of-china-and-japan/>.

2. Информатика, вычислительная техника и управление

Аль-Имари М.Д.Б.

МЕЖУРОВНЕВАЯ СХЕМА КООРДИНАЦИИ, ОРИЕНТИРОВАННАЯ НА МИКРОСЕРВИСЫ

Воронежский государственный технический университет

1. Введение

Современные сервисы облачных вычислений предъявляют жесткие требования к производительности, удовлетворение которых требует применения передовых технологий и значительных исследовательских усилий. В стремлении к гибкому управлению облачной инфраструктурой и разработке сервисов, многие ведущие провайдеры, включая Netflix [1], Amazon [2] и Microsoft [3], внедряют микросервисную архитектуру. Микросервисы могут разрабатываться с использованием различных языков программирования и моделей, а также развертываться и управляться независимо, что не оказывает влияния на штатное функционирование друг друга.

Оптимизация производительности облачных вычислительных систем в эпоху микросервисов представляет собой сложную многогранную задачу. Во-первых, центры обработки данных становятся всё более гетерогенными и децентрализованными на инфраструктурном уровне из-за внедрения специализированного оборудования [4], ускорителей [5] и новых серверных архитектур [6]. Во-вторых, приложения на основе микросервисов характеризуются сложной динамикой времени выполнения. Сервисные единицы обычно развертываются распределенно с использованием облегченной виртуализации [7]. В-третьих, по сравнению с монолитными приложениями, микросервисные системы предъявляют более строгие требования к производительности.

В статье проводится сравнительный анализ методов оптимизации производительности микросервисов в облачных средах. Проблема оптимизации производительности разделена на три ключевых аспекта, для каждого из которых исследуются последние достижения с целью формирования целостного представления о проблеме. Основное внимание уделяется методам, направленным на решение задач управления ресурсами облачных систем для обеспечения высокой производительности. В заключение обобщаются открытые вопросы и сложности, связанные с особенностями микросервисной архитектуры.

Для исследователей и разработчиков крайне важно получить глубокое понимание потенциальных узких мест производительности и исследовательских задач микросервисной архитектуры, а также последних достижений в области методов оптимизации производительности. В данной статье предпринята попытка всестороннего изучения различных методов, направленных на повышение производительности микросервисов. В целом, вклад настоящей работы заключается в следующем:

1) Проведено тщательное сравнение важных терминов и концепций, тесно связанных с облачными вычислительными системами в эпоху микро-

сервисов.

2) Представлена классификация методов оптимизации производительности микросервисов в облаке, выполнен анализ преимуществ и недостатков существующих проектов, а также обозначены возможности для исследований в этих областях.

3) Обобщены открытые проблемы и вызовы, с которыми сталкивается управление облачными микросервисами.

2. Определения

В данном разделе представлен обзор парадигмы микросервисов и ее сравнение с другими родственными концепциями.

2.1. Микросервис: краткий обзор

Термин «микросервис» был впервые введен в контексте сервис-ориентированной архитектуры (SOA) как «гранулированный» подход [8], что было обусловлено распределенной природой облачных систем и приложений [9]. В общем смысле микросервис определяется как архитектурный стиль проектирования программного обеспечения, противопоставляемый «монолитному» стилю. В данной парадигме приложение состоит из многоуровневых, функционально простых служб, каждая из которых выполняется в независимом процессе и взаимодействуют через открытые порты для предоставления комплексных услуг. Более конкретно, архитектура микросервисов отличается от традиционных облачных приложений следующими аспектами:

- Разделенная функциональность.
- Независимая разработка.
- Децентрализованное управление.

2.2. Сравнение родственных понятий

Появление микросервисов является результатом повсеместного распространения облачных вычислений, а также потребности в децентрализованном управлении приложениями и оперативном обновлении сервисов. Помимо термина «микросервис», для разработки программного обеспечения и управления облаками были предложены и другие архитектурные модели.

SOA. Подобно микросервисам, сервис-ориентированная архитектура (SOA) характеризуется модульным дизайном и коммуникацией на основе сообщений [10]. В SOA сервисы выполняют небольшие функции. Корпоративная сервисная шина (ESB) применяется для соединения служб и потребителей, а также для обмена данными между всеми службами, что делает SOA более интегрированным решением [11]. Микросервисы отличаются от SOA большей степенью развязанности:

1) В архитектуре микросервисов сервисы в основном взаимодействуют друг с другом посредством более простых сетевых запросов, тогда как в SOA ESB в основном отвечает за внутреннюю коммуникацию.

2) В микросервисах хранение данных децентрализовано в соответствии с принципом «разделять как можно меньше данных», тогда как в SOA ESB отвечает за централизованный обмен данными.

3) Сервисные компоненты в микросервисной системе имеют индивидуальный порт доступа для пользователей и обеспечивают независимые функции.

Бессерверные вычисления. Бессерверные вычисления были впервые предложены для одноранговой программной платформы и привлекли внимание исследователей с момента появления AWS Lambda [12]. Как правило, в облачной платформе выделяют два варианта: «Функция как услуга» (FaaS) и «Серверная часть как услуга» (BaaS) [13]. Микросервис может быть развернут бессерверным способом, при этом его отличие от бессерверной функции заключается в том, что микросервис может состоять из нескольких функций [14].

Архитектура, управляемая событиями (EDA). Событийно-ориентированная архитектура считается основой микросервисов и FaaS [15]. В этой модели услуга вызывается через вызов функции или запрос от другой службы или конечных пользователей. Службы координируют свои действия друг с другом, публикуя события и подписываясь на них, вместо привязки потоков к запросам на протяжении всего процесса выполнения.

Архитектура, управляемая моделями (MDA). MDA часто предлагается при разработке микросервисов [16]. Впервые этот подход был представлен как подход к разработке, основанный на моделях [17], при котором абстракция системы используется для проектирования программного приложения [18]. Модель представляет собой набор утверждений о составе системы и ее связях с другими системами [19]. В MDA микросервисные системы могут моделироваться с помощью определенных языков на различных уровнях абстракции. При этом программные реализации могут автоматически генерироваться с помощью языков описания.

2.3. Оптимизация производительности

Методы оптимизации производительности облачных вычислительных систем в эпоху микросервисов являются более сложными по сравнению с традиционными проектами, ориентированными на монолитные приложения. Особенно сложно отслеживать состояние производительности и задержку запросов мелкогранулярных микросервисов в крупных облачных системах [20]. Факторы, влияющие на производительность микросервисов, охватывают все уровни, включая архитектуру программного обеспечения и разработку приложений, согласование сервисов, планирование емкости ресурсов, изменение времени выполнения и динамическую конфигурацию базовой аппаратной системы. Тщательно рассматриваются предложенные методы и подходы в предыдущих работах, и представляются возможности и проблемы для исследований в этих трех ключевых областях.

Оценка и мониторинг производительности. Микросервисные приложения имеют более гранулярную архитектуру и большое разнообразие в программной архитектуре, режиме взаимодействия и среде выполнения.

Предоставление ресурсов и управление ими. На протяжении многих лет предоставление ресурсов в центрах обработки данных остается серьезной проблемой. Для управления крупномасштабной системой микросервисов облачные провайдеры предлагают различные инструменты и платформы для организации приложений и управления распределением ресурсов. Такие платформы, как правило, требуют от пользователей настройки требований к ресурсам и применения базовых стратегий для размещения служб и коррек-

тировки ресурсов. Более того, в высокодинамичной среде выполнения производительность некоторых критически важных служб может напрямую влиять на другие службы.

Настройка и оптимизация системы. Микросервисы имеют меньшую задержку по сравнению с традиционными облачными сервисами. Для оптимизации производительности на системном уровне одной из возможностей исследования является снижение нагрузки на операционную систему и сеть, чтобы адаптироваться к задержке в микросекундном масштабе.



Рис. 1.

3. Оценка эффективности и мониторинг

Пре предыдущие работы по оценке производительности и мониторингу микросервисов были направлены на разработку различных компонентов, которые в основном подразделяются на три подкатегории: сравнительный анализ, анализ и характеристика, мониторинг и обнаружение аномалий.

3.1. Оценки

Поставщики облачных услуг разработали простые тесты с открытым исходным кодом для облегчения исследований микросервисов, такие как [51]. Поскольку в предыдущих исследованиях использовались простые тесты, их результаты не проверялись в рабочей среде. Поэтому необходимы более сложные тесты [22].

С переходом от монолитных к микросервисам современные приложения с интенсивным использованием данных в режиме реального времени (OLDI) теперь сталкиваются с новыми требованиями к субмиллисекундной задержке [23]. В [24] впервые предложен тест для изучения фундаментального проек-

тирования микросервиса.

3.2. Анализ и характеристика

Производительность микросервисного приложения отличается от производительности его монолитных аналогов. На более низком системном уровне исследования были сосредоточены на характеристике микросервисов, включая их детализированные показатели производительности, поведение ресурсов и архитектурные последствия. На уровне виртуализации существует ряд работ, в которых изучается влияние различных технологий виртуализации на производительность. На прикладном уровне в некоторых недавних работах рассматриваются различные варианты разработки, такие как архитектурный дизайн, протокол связи, потоковая модель, и их влияние на производительность приложения.

Влияние на систему и архитектуру. В [25] сравнивалась пропускная способность монолитного и микросервисного приложений, а также проводились эксперименты по длине пути, промахам в кэше и циклам на команду в микросервисной системе с использованием разных языков программирования. В [26] была проведена подробная характеристика пакета тестов для потоковой передачи фильмов, касающаяся распределения циклов, количества команд в секунду и нагрузки на кэш. Этот анализ дает представление об управлении облаком, а также об архитектурном проектировании на уровне инфраструктуры.

Сравнение методов виртуализации. Накладные расходы, связанные с уровнем виртуализации, побудили многих исследователей обратить внимание на эффективную среду виртуализации. Несколько популярных платформ виртуализации, таких как Unikernel, Docker и KVM, сравниваются и оцениваются с точки зрения эффективности одновременной подготовки нескольких экземпляров [27]. В некоторых работах основное внимание уделяется системным издержкам и влиянию на производительность сети, которые оказывают различные технологии виртуализации и контейнеризации. В [28] было проведено исследование, посвященное накладным расходам и влиянию на производительность Docker. В [29] проанализирована производительность сети, в частности, в отношении трафика между хостами. Авторы сравнили несколько моделей развертывания микросервиса: «голое железо», контейнеризация «ведущий-подчиненный», вложенная контейнеризация и виртуальная машина. Они также представили сетевой стек каждого механизма.

3.3. Мониторинг и обнаружение аномалий

Крупные облачные центры обработки данных обычно используют мониторинг производительности и отслеживание аномалий. Эти темы были широко изучены в [30]. Для приложений, основанных на микросервисах, существует ряд новых проблем, которые необходимо решить. Во-первых, поскольку запросы на микросервисы обычно распространяются на несколько уровней сервиса и каждый сервис имеет разное состояние и поведение в различных конфигурациях, необходимо разработать новые инструменты для отслеживания, профилирования и сбора данных. Во-вторых, приложения для микросервисов, как правило, имеют сложную архитектуру. Нам нужны механизмы для

получения информации о поведении сервиса и аномалиях производительности. С этой целью в некоторых предыдущих работах использовались сложные модели для анализа трассировки данных.

4. Предоставление ресурсов и управление ими

Надлежащее выделение ресурсов и управление ими являются ключом к обеспечению высокой производительности. Мы разделяем предыдущие исследования на три подкатегории:

- 1) моделирование и прогнозирование,
- 2) размещение и оркестровка,
- 3) адаптация во время выполнения.

4.1. Моделирование и прогнозирование

Одним из аспектов оптимизации выделения ресурсов является принятие решений по планированию пропускной способности на основе надежных моделей прогнозирования. Это требует понимания производительности системы при различных конфигурациях ресурсов. Часто службы, чувствительные к задержкам, имеют широкий диапазон рабочих нагрузок, с большой долей низкого и умеренного спроса [31] и небольшой долей пиковых нагрузок [32]. Поэтому построение точной модели прогнозирования рабочей нагрузки может быть сложной задачей.

Одним из важных направлений работы является построение нелинейной модели для прогнозирования производительности. Например, в [92] задача разделена на две части: анализ платформы микросервиса и анализ инфраструктуры макросервиса. В этой работе построена модель цепочки Маркова для платформы микросервисов, подготовки виртуальных машин и RM provisioning соответственно. Например, для подмодели платформы микросервисов используется количество запросов, контейнеров и виртуальных машин для обозначения состояния системы. В работе [33] была смоделирована микросервисная система в виде ориентированного графа.

4.2. Размещение и оркестровка

Микросервисы, как правило, развертываются на легких виртуальных машинах (VM) или контейнерах. Размещение и согласование служб оказывают нетривиальное влияние на производительность соответствующего приложения.

В некоторых предыдущих работах оркестровка микросервисов в основном рассматривалась как задача оптимизации, и для оценки предлагаемого подхода использовалась платформа моделирования. В [34] были рассмотрены три ключевых показателя оркестровки для контейнеров и сервисов в мультиоблачной среде: стоимость эксплуатации, время выполнения и время восстановления. В этой работе разработан алгоритм для определения распределения виртуальной машины и размещения контейнера. В работе [35] была поставлена цель минимизировать количество узлов, используемых для микросервиса, в соответствии с требованиями QoS. Ключевым методом является прогнозирование использования ресурсов вычислительными узлами на основе исторических данных. Затем используется модель попарного ранжирования для принятия решения о развертывании службы.

4.3. Адаптация к среде выполнения в реальном времени

Оптимизация производительности микросервисов во время выполнения требует сочетания передовых методов адаптации ресурсов. На прикладном уровне планирование запросов изучалось с использованием различных моделей определения задержек. Контроль перегрузки является еще одним важным аспектом для служб, позволяющим поддерживать нормальный отклик или масштабирование до насыщения, включая такие механизмы, как снижение нагрузки [36] и увеличение ресурсов [37]. На уровне базовой инфраструктуры в последние годы были предложены различные онлайн-планировщики для монолитных облачных сервисов [38].

5. Настройка и координация системы

Разукрупнение на прикладном уровне микросервиса приводит к разнообразию требований к системе и архитектуре. В то же время, приложения-микросервисы имеют гораздо меньшее время обработки запросов, обычно в масштабе микросекунд [39]. Эти факторы требуют переосмысления проектов на системном уровне, которые могут повысить эффективность выполнения и взаимодействия, а также обеспечить более гибкую системную координацию по всему вычислительному стеку.

Небольшая задержка запросов на микросервисы создает новые проблемы для существующих систем управления энергопотреблением. Для служб миллисекундного масштаба традиционные схемы часто используют преимущества низкой задержки и интервалов поступления запросов для управления работой процессорных ядер в различных режимах энергопотребления (уровни частоты/напряжения) [40]. Однако эффективность этих подходов может быть весьма ограниченной, поскольку интервал запроса в микросекундном масштабе приводит к фрагментированным периодам простоя и недостаточному времени для перехода в режим питания. Для решения проблемы задержки реализована контейнерная система с помощью:

- 1) использование пространства имен и изоляции групп управления для снижения уровня производительности;
- 2) использование полных языковых репозиторий на рабочих компьютерах, чтобы избежать затрат на инициализацию Python;
- 3) разработка многоуровневой системы кэширования. Представленная система значительно улучшила время ожидания при холодном запуске и пропускную способность.

6. Возможности для проведения исследований

Эффективная поддержка приложений на базе микросервисов в облаке - непростая задача. В этом разделе мы суммируем основные нерешенные проблемы, связанные с микросервисами. На рис. 2 мы представляем основные проблемы, связанные с различными уровнями облачных вычислений, и обсуждаем возможные решения.

6.1. Новые модели и абстракции

Хотя микросервисная архитектура обеспечивает большую гибкость при разработке и развертывании приложений, она также снижает эффективность существующих подходов к оптимизации системы, используемых для моно-

литных приложений. Например, приложения сталкиваются с сервисными компонентами с различными зависимостями, каждый из которых может быть написан на разных платформах программирования и развернут с использованием разных контейнерных технологий. Поскольку будущие приложения будут демонстрировать кардинально отличающиеся схемы рабочей нагрузки, потребуются методы моделирования и прогнозирования рабочей нагрузки на уровне микросервисов. Более того, по мере того, как мы внедряем более специализированное и разукрупненное оборудование/ускорители, неоднородность центров обработки данных возрастает. Важно создавать новые модели для анализа взаимодействия между приложениями и базовыми ресурсами.

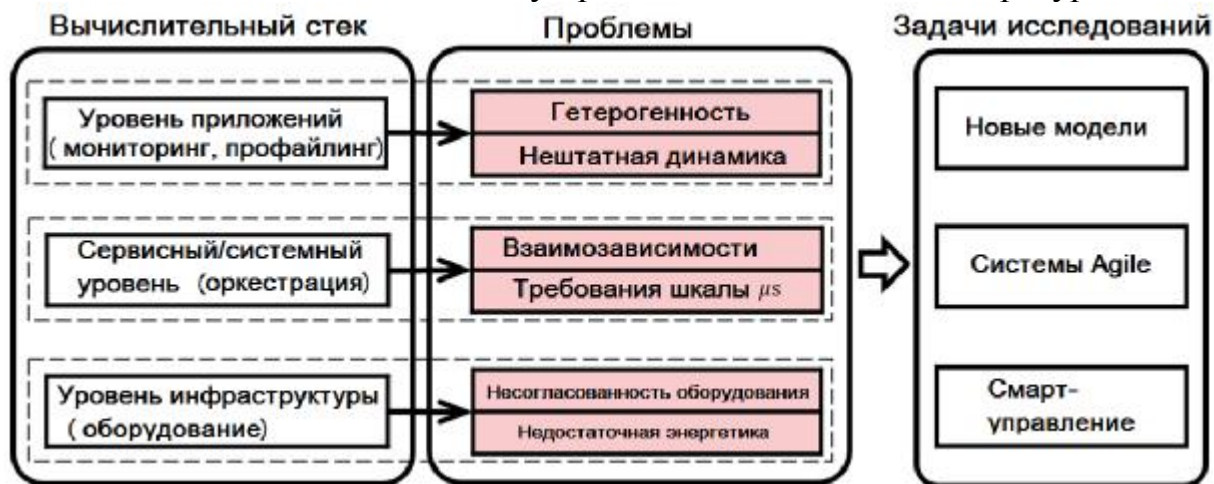


Рис. 2. Нерешенные проблемы в эпоху микросервисов

6.2. Системная оркестровка в масштабе мкс

Микросервис перенес компьютерную систему в «эру микросекунд-убийц». Программистам легко уменьшить задержку событий в наносекундном и миллисекундном масштабе времени (например, доступ к DRAM занимает десятки или сотни наносекунд, а дисковый ввод-вывод - несколько миллисекунд). Однако было проделано мало работы с точки зрения поддержки событий микросекундного масштаба. Предыдущие исследования показали, что субмиллисекундные системные издержки (например, переключение потоков) оказывают нетривиальное влияние на задержку микросервиса. Поэтому важно разрабатывать системы, которые могут лучше поддерживать параллелизм, масштабирование частоты и механизм ввода-вывода. В частности, традиционное управление питанием в крупномасштабных системах приводит к длительной задержке. Также представляет большой интерес количественная оценка задержки при управлении питанием, выявление причины снижения производительности и устранение задержки.

6.3. Автономное управление ресурсами

Разработка системных механизмов, которые могут адаптивно и автономно управлять базовыми вычислительными ресурсами для удовлетворения потребностей микросервисов, приобретает все большее значение. Часто операторам и проектировщикам облачных центров обработки данных приходится искать нужную конфигурацию оборудования и распределять ресурсы. Требования к производительности различаются для разных служб и ресур-

сов, и количество конфигураций может быть огромным. Без надлежащего проектирования поиск решений по планированию при определенных ограничениях производительности может занять много времени. Было бы полезно использовать самые современные методы машинного обучения для управления базовыми вычислительными ресурсами. Кроме того, вычислительные среды могут быстро меняться по мере масштабирования или миграции приложений. Чтобы постоянно принимать разумные решения по планированию в сложной операционной среде, необходимо разрабатывать новые методы автономного управления. Другими словами, система, разработанная для приложений на базе микросервисов, должна обладать улучшенными возможностями самостоятельного управления.

6.4. Многослойная разработка

В конечном счете, мы ожидаем, что в ближайшем будущем для вычислений в масштабах центров обработки данных потребуется межуровневая схема координации, ориентированная на микросервисы. Сегодня различные методы управления ресурсами применяются отдельно на разных уровнях стека облачных вычислений. В то время как новые аппаратные компоненты развертываются на нижнем уровне инфраструктуры, программные системы верхнего уровня не могут использовать эту гибкость из-за отсутствия координации. Поскольку мы переходим от традиционного монолитного подхода к микросервисам, крайне важно выработать глубокое понимание поведения совместного использования ресурсов на разных уровнях вычислений (аппаратное обеспечение, архитектура, операционная система, промежуточное программное обеспечение и т.д.). В частности, большое значение имеет разработка гибких и адаптивных механизмов координации для мелкозернистых, недолговечных сервисов.

7. Заключение

Широкое распространение различных облачных приложений продолжает стимулировать развитие вычислений в масштабах центров обработки данных. В последнее время, с развитием технологий виртуализации и реализацией микросекундной задержки в сети, архитектура микросервисов стала многообещающей при разработке облачных приложений. Микросервис не только открывает новые возможности, но и усложняет выполнение требований к производительности.

Список использованных источников

1. Adopting microservices at netflix: Lessons for team and process design. <https://www.nginx.com/learn/microservices/>.
2. Microservices architectures on amazon web services. <https://docs.aws.amazon.com/whitepapers/latest/microservices-on-aws/simple-microservices-architecture-on-aws.html>.
3. Microservices in Azure. <https://azure.microsoft.com/en-us/solutions/microservice-applications/>.
4. Norman P Jouppe, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In 2017 ACM/IEEE 44th Annual Int. Symp. on Computer Architecture (ISCA), pp. 1–12. IEEE, 2017.
5. Eric Chung, Jeremy Fowers, Kalin Ovtcharov, Michael Papamichael, Adrian Caulfield,

Todd Massengill, Ming Liu, Daniel Lo, Shlomi Alkalay, Michael Haselman, et al. Serving dnns in real time at datacenter scale with project brainwave. *IEEE Micro*, 38(2):8–20, 2018.

6. Vlad Nitu, Boris Teabe, Alain Tchana, Canturk Isci, and Daniel Hagimont. Welcome to zombieland. In *Proceedings of the Thirteenth EuroSys Conf. on - EuroSys '18*, pp. 1–12, New York, New York, USA, 2018. ACM Press.

7. Serverless deployment. <https://microservices.io/patterns/deployment/serverless-deployment.html>.

8. James Lewis. Main sponsor Micro Services-Java the Unix way. Technical report.

9. Learn from SOA: 5 lessons for the microservices era. <https://www.infoworld.com/article/3080611/>.

10. Bob Familiar. Microservice Architecture. In *Microservices, IoT, and Azure*, pp. 21–31. 2015.

11. Pooyan Jamshidi, Claus Pahl, Nabor C Mendonça, James Lewis, and Stefan Tilkov. Microservices: The journey so far and challenges ahead. *IEEE Software*, 35(3):24–35, 2018.

12. NEW LAUNCH: Getting Started with AWS Lambda - YouTube.

13. Microservice and Serverless Computing. <https://www.endava.com/en/blog/Engineering/2019/Microservices-and-Serverless-Computing>.

14. What Is a Serverless Microservice? <https://www.cloudflare.com/learning/serverless/glossary/serverless-microservice/>.

15. Geoffrey C. Fox, Vatche Ishakian, Vinod Muthusamy, and Aleksander Slominski. Status of serverless computing and function-as-a-service(faas) in industry and research. *CoRR*, abs/1708.08028, 2017.

16. Florian Rademacher, Sabine Sachweh, and Albert Zündorf. Differences between model-driven development of service-oriented and microservice architecture. In *2017 IEEE Int. Conf. on Software Architecture Workshops (ICSAW)*, pp. 38–45. IEEE, 2017.

17. Richard Soley et al. Model driven architecture. *OMG white paper*, 308(308):5, 2000.

18. Alberto Rodrigues da Silva. Model-driven engineering: A survey supported by the unified conceptual model. *Computer Languages, Systems Structures*, 43:139 – 155, 2015.

19. Edwin Seidewitz. What models mean. *IEEE software*, 20(5):26–32, 2003.

20. Rajiv Ranjan, Chang Liu, Lydia Chen, Maria Fazio, Massimo Villari, and Antonio Celesti. Open Issues in Scheduling Microservices in the Cloud. *IEEE Cloud Computing*, 3(5):81–88, 2016.

21. Socks shop – a microservices demo application. <https://microservices-demo.github.io/>.

22. Nuha Alshuqayran, Nour Ali, and Roger Evans. A systematic mapping study in microservice architecture. In *Proceedings - 2016 IEEE 9th Int. Conf. on Service-Oriented Computing and Applications, SOCA 2016*, pp. 44–51. IEEE, nov 2016.

23. A. Sriraman and T. F. Wenisch. μ suite: A benchmark suite for microservices. In *2018 IEEE Int. Symp. on Workload Characterization (IISWC)*, pp. 1–12, Sep. 2018.

24. Nane Kratzke and Peter-Christian Quint. Ppbench. In *Proceedings of the 6th Int. Conf. on Cloud Computing and Services Science - Volume 1 and 2, CLOSER 2016*, page 223–231, Setubal, PRT, 2016. SCITEPRESS - Science and Technology Publications, Lda.

25. T. Ueda, T. Nakaike, and M. Ohara. Workload characterization for microservices. In *2016 IEEE Int. Symp. on Workload Characterization (IISWC)*, pp. 1–10, Sep. 2016.

26. Yu Gan and Christina Delimitrou. The Architectural Implications of Cloud Microservices. *IEEE Computer Architecture Letters*, 17(2):155–158, 2018.

27. Bruno Xavier, Tiago Ferreto, and Luis Jersak. Time Provisioning Evaluation of KVM, Docker and Unikernels in a Cloud Platform. In *Proceedings 2016 16th IEEE/ACM Int. Symp. on Cluster, Cloud, and Grid Computing, CCGrid 2016*, pp. 277–280, 2016.

28. Anders Västberg. Container overhead in microservice systems Container overhead in microservice systems. 2018.

29. Marcelo Amaral, Jordà Polo, David Carrera, Iqbal Mohomed, Merve Unuvar, and Malgorzata Steinder. Performance evaluation of microservices architectures using containers. In *Proceedings - 2015 IEEE 14th Int. Symp. on Network Computing and Applications, NCA 2015*,

pp. 27–34. IEEE, sep 2016.

30. Hyunwook Baek, Abhinav Srivastava, and Jacobus Van der Merwe. Cloudsight: A tenant-oriented transparency framework for cross-layer cloud troubleshooting. In Proceedings of the 17th IEEE/ACM Int. Symp. on Cluster, Cloud and Grid Computing, pp. 268–273. IEEE Press, 2017.

31. Xiao Zhang, Eric Tune, Robert Hagmann, Rohit Jnagal, Vrigo Gokhale, and John Wilkes. Cpi 2: CPU performance isolation for shared compute clusters. In Proceedings of the 8th ACM European Conf. on Computer Systems, pp. 379–391. ACM, 2013.

32. Artemiy Margaritov, Siddharth Gupta, Reikai Gonzalez-Alberquilla, and Boris Grot. Stretch: Balancing qos and throughput for colocated server workloads on smt cores. In 2019 IEEE Int. Symp. on High Performance Computer Architecture (HPCA), pp. 15–27. IEEE, 2019.

33. Marco Gribaudo, Mauro Iacono, and Daniele Manini. Performance Evaluation Of Massively Distributed Microservices Based Applications. 6(Cd):598–604, 2017.

34. Carlos Guerrero, Isaac Lera, and Carlos Juiz. Resource optimization of container orchestration: a case study in multi-cloud microservices-based applications. Journal of Supercomputing, 74(7):2956–2983, 2018.

35. Xue Leng, Tzung-Han Juang, Yan Chen, and Han Liu. Aomo: An ai-aided optimizer for microservices orchestration. In Proceedings of the ACM SIGCOMM 2019 Conf. Posters and Demos, pp. 1–2, 2019.

36. Pieter J. Meulenhoff, Dennis R. Ostendorf, Miroslav Živkovic, Hendrik B. Meeuwissen, and Bart M. M. 't Gijsen. Intelligent overload control for composite web services. In Luciano Baresi, Chi-Hung Chi, and Jun Suzuki, editors, Service-Oriented Computing, pp. 34–49, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

39. Chih-Hsun Chou, Laxmi N Bhuyan, and Daniel Wong. μ dpm: Dynamic power management for the microsecond era. In 2019 IEEE Int. Symp. on High Performance Computer Architecture (HPCA), pp. 120–132. IEEE, 2019.

40. Harshad Kasture, Davide B Bartolini, Nathan Beckmann, and Daniel Sanchez. Rubik: Fast analytical power management for latency-critical systems. In 2015 48th Annual IEEE/ACM Int. Symp. on Microarchitecture (MICRO), pp. 598–610. IEEE, 2015.

Баталов Д.И., Рындин А.А.

СТРАТЕГИИ СЕТЕВОГО МОНИТОРИНГА ДЛЯ СЕТЕЙ КРИТИЧЕСКОЙ ИНФРАСТРУКТУРЫ

**Петербургский государственный университет путей сообщения
Императора Александра I**

Воронежский государственный технический университет

1. Введение

Современное общество демонстрирует беспрецедентный уровень зависимости от функционирования критически важных инфраструктур, охватывающих такие секторы, как энергетика, водоснабжение, телекоммуникации и транспорт. Надежность и непрерывность работы этих систем являются фундаментальным условием экономической стабильности, национальной безопасности и общественного благосостояния. В силу своей значимости, данные инфраструктуры представляют собой первостепенные цели для различного рода деструктивных воздействий, будь то физические диверсии, кибератаки или технологические сбои, способные привести к каскадному отказу взаимосвязанных компонентов.

Процесс модернизации этих инфраструктур, часто обозначаемый терми-

ном "Индустрия 4.0" или "умная инфраструктура", подразумевает глубокую интеграцию киберфизических систем и повсеместное внедрение коммуникационных технологий для повышения ситуационной осведомленности, эффективности управления и надежности. Однако эта интеграция, обеспечивая несомненные преимущества, приводит к неуклонному расширению и повышению доступности поверхности атаки. Ранее изолированные и функционировавшие на основе проприетарных протоколов промышленные системы управления (ICS) и системы диспетчерского управления и сбора данных (SCADA) теперь все чаще используют стандартные протоколы TCP/IP и подключаются к корпоративным сетям и облачным платформам, что делает их уязвимыми для атак, изначально разработанных для IT-сред.

Анализ известных инцидентов в данной области наглядно демонстрирует эволюцию угроз. Атака Stuxnet, поразившая иранские предприятия ядерной программы, впервые показала возможность целенаправленного физического воздействия на промышленное оборудование через киберпространство. Впоследствии вредоносное ПО BlackEnergy и Crash Override (также известное как Industroyer), примененное против украинских энергетических компаний, продемонстрировало способность злоумышленников к прямому манипулированию коммутационными аппаратами и вызыванию масштабных отключений электроэнергии. В свою очередь, кампания Havex, нацеленная на промышленные предприятия в Соединенных Штатах и Европе, показала методы разведки и сбора информации из внутренних сетей ICS с использованием легитимных функций протоколов. Эти инциденты свидетельствуют о возросшей изощренности злоумышленников, их глубоком понимании технологических процессов и умении использовать возросшую поверхность атаки для достижения физических последствий.

Дальнейшее усугубление ситуации связано с экспоненциальным ростом числа подключенных к облаку промышленных устройств Интернета вещей (Industrial Internet of Things, IIoT) в этих инфраструктурах. Многообразие датчиков, исполнительных механизмов и интеллектуальных электронных устройств (IED) создает огромное количество потенциальных точек входа. Более того, высокая степень взаимозависимости между компонентами инфраструктуры приводит к потенциальным каскадным и эскалирующим сбоям, когда отказ или компрометация одного элемента может вызвать цепную реакцию, выводящую из строя значительные сегменты сети или даже всю инфраструктуру. Совокупность этих факторов - расширение поверхности атаки, рост числа подключенных устройств и взаимозависимость - формирует настоятельную необходимость в повышении уровня кибербезопасности и разработке эффективных механизмов защиты для данных сред.

Ключевым элементом любой стратегии киберзащиты является мониторинг сетевого трафика и анализ поведения устройств. Именно мониторинг позволяет обнаруживать вторжения, выявлять аномалии в работе протоколов, диагностировать сбои и собирать данные для последующего криминалистического анализа. Однако разработка эффективной и практически реализуемой стратегии мониторинга сопряжена с рядом фундаментальных проблем,

требующих формального решения.

Во-первых, существует множество различных типов мониторов, каждый из которых предназначен для решения специфических задач: системы обнаружения вторжений (IDS), анализирующие сетевой трафик на наличие сигнатур атак; системы предотвращения вторжений (IPS), функционирующие в активном режиме; антивирусные решения, сканирующие файловые системы и память хостов; системы анализа журналов событий (SIEM), агрегирующие и коррелирующие данные из различных источников; специализированные решения для мониторинга целостности файлов и контроля конфигураций. Каждый тип монитора обладает своими функциональными возможностями, ограничениями по производительности и фокусируется на различных аспектах поведения системы. Таким образом, выбор того, что именно отслеживать, какой тип монитора для этого использовать и в каком месте сети его развернуть, является нетривиальной задачей оптимизации.

Во-вторых, развертывание и эксплуатация мониторов сопряжены со значительными капитальными (CAPEX) и операционными (OPEX) затратами. Стоимость монитора не ограничивается только ценой оборудования и его установкой. Существенным фактором является стоимость анализа генерируемых ими предупреждений. Каждое срабатывание, будь то истинное или ложноположительное, требует внимания оператора, время которого ограничено. Высокий уровень ложных срабатываний может привести к "усталости от предупреждений", когда операторы начинают игнорировать сигналы системы безопасности, что критически снижает эффективность защиты. Следовательно, стратегия мониторинга должна быть экономически обоснованной и минимизировать издержки, связанные с обработкой событий безопасности.

В-третьих, мониторинг сетей критической инфраструктуры подчиняется ряду специфических ограничений, которые редко встречаются в традиционных корпоративных ИТ-средах. Существуют жесткие нормативные требования. Например, в системах, ответственных за выработку и передачу значительных объемов электроэнергии, действуют требования, установленные Североамериканской корпорацией по надежности электроснабжения (NERC) в рамках стандартов защиты критической инфраструктуры (CIP). Эти стандарты предписывают конкретные меры безопасности, включая мониторинг и аудит. Следствием этого является то, что большее количество отслеживаемых устройств приводит к возрастанию объема работ и затрат, связанных с учетом этих устройств при проведении аудитов NERC-CIP, так как каждое устройство должно соответствовать определенным требованиям по управлению доступом, ведению журналов и реагированию на инциденты.

Кроме того, существуют жесткие временные ограничения на обмен данными (Quality of Service, QoS). Коммуникационные задержки в сетях с протоколами реального времени, такими как GOOSE (Generic Object Oriented Substation Event) или SV (Sampled Values) в подстанциях, строго регламентированы. Нарушение этих временных ограничений, например, из-за внесения дополнительной задержки сетевым монитором, может привести к нестабильности физических процессов, управляемых этими сетями, и, как следствие,

перевести систему в аварийное или небезопасное состояние. Мониторинг не должен вносить неприемлемых задержек в критические потоки данных. Фактически, рекомендации NERC прямо указывают на необходимость осуществления мониторинга датчиков, журналов, систем обнаружения вторжений, антивирусного программного обеспечения, систем управления исправлениями и других механизмов безопасности в режиме реального времени или с минимально возможной задержкой, что накладывает дополнительные требования к производительности и месту развертывания мониторов.

Учитывая вышеизложенные проблемы, возникает центральный вопрос исследования: какова наилучшая стратегия мониторинга для данной киберсети с учетом заданных ограничений на бюджет, нормативные требования и допустимые задержки? Формально, стратегия мониторинга представляет собой отображение, которое определяет, какие сетевые потоки должны отслеживаться, с помощью каких типов мониторов и в каких точках сети (на каких узлах или ребрах) эти мониторы должны быть развернуты.

Для оценки эффективности различных стратегий необходимо ввести количественный показатель, который позволяет сравнивать их между собой и определять оптимальный вариант. В данной работе мы определяем такой показатель, который, по сути, количественно оценивает "ценность" каждого конкретного сопоставления потоков и мониторов. Данный показатель является комплексным и учитывает два ключевых аспекта. Первый аспект - это важность отслеживаемых потоков. Потоки могут быть критически важными для управления технологическим процессом, содержать конфиденциальную информацию или быть частью системы безопасности. Более важным потокам должен быть присвоен более высокий приоритет в стратегии мониторинга. Второй аспект - это местоположение монитора относительно конечных точек потока. Монитор, расположенный на одном из концов потока, имеет доступ ко всему трафику этого потока, в то время как монитор, установленный на промежуточном узле, может видеть только часть трафика или только его метаданные. Кроме того, мониторинг на конечных точках позволяет коррелировать события на отправителе и получателе, что повышает точность обнаружения.

Выбор наилучшего отображения из комбинаторно большого пространства всех возможных стратегий является вычислительно сложной задачей. Поиск точного оптимального решения, как мы покажем далее, является NP-трудной задачей, что делает невозможным применение методов полного перебора даже для сетей среднего размера в приемлемое время. Это требует разработки приближенных алгоритмов, которые могут гарантировать нахождение решения, близкого к оптимальному, за полиномиальное время.

В данной работе мы описываем и анализируем такой детерминированный алгоритм. Этот алгоритм может использоваться сетевыми операторами для принятия обоснованных решений о почти оптимальном развертывании мониторов, исходя из их бюджета безопасности и жестких требований к QoS приложений критической инфраструктуры. Алгоритм обеспечивает корректное распределение ограниченных ресурсов, определяя, какие потоки следует

отслеживать, какие типы мониторов использовать и где их разместить. Это позволяет администратору перейти от интуитивных или эвристических решений к формализованному и математически обоснованному подходу, повышая эффективность инвестиций в безопасность и снижая общий кибер-риск.

В качестве мотивирующего примера, иллюстрирующего практическую значимость поставленной задачи, рассмотрим локальную вычислительную сеть (LAN) передающей подстанции электросети. Схема такой сети с одним коммутатором представлена на рис. 1. Мы специально рассмотрим применение функции проверки синхронизма для управления замыканием выключателя. Для стабильного, надежного и безопасного функционирования энергосистемы необходимо, чтобы все генераторы и нагрузки работали синхронно, то есть с одинаковой частотой и фазой. Приложение проверки синхронизма гарантирует, что перед замыканием выключателя, соединяющего две части энергосистемы, напряжения с обеих сторон выключателя совпадают по фазе, амплитуде и частоте в заданных пределах. На высоком уровне приложение измеряет угол между однофазными напряжениями с обеих сторон выключателя (со стороны шины и со стороны линии) и определяет, находятся ли они в заданных пределах.

Рассмотрим коммуникационные потоки, необходимые для работы этого приложения, и потенциальные точки мониторинга.

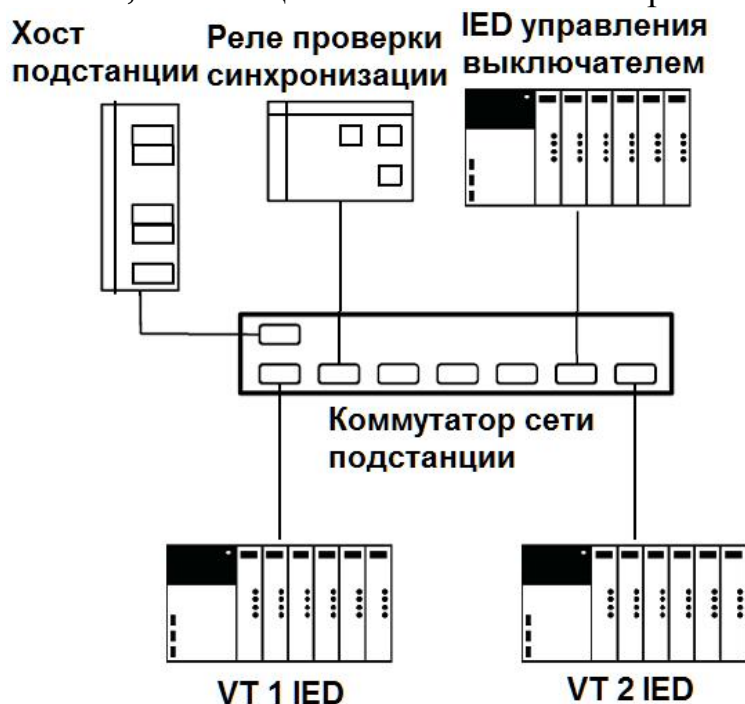


Рис. 1. Локальная сеть с одним коммутатором подстанции

Владелец подстанции определяет, какие датчики (трансформаторы напряжения, VT) должны сообщать о параметрах напряжения, и опрашивает соответствующие интеллектуальные электронные устройства (IED), подключенные к этим датчикам.

IED, связанные с трансформаторами напряжения (VT IED), добавляют временную метку (полученную с центрального сервера времени, например,

через протокол RTP - Precision Time Protocol) к информации о напряжении и частоте, формируют пакет данных и отправляют его на реле проверки синхронизации через сетевой коммутатор.

Реле проверки синхронизации получает эти пакеты, анализирует временные метки и значения напряжения, определяет, синхронизированы ли напряжения. В случае успешной проверки, оно формирует управляющую команду и отправляет ее на устройство управления выключателем.

Устройство управления выключателем интерпретирует полученное управляющее сообщение и выполняет соответствующее управляющее действие - подает сигнал на привод выключателя для его замыкания.

Требование к крайнему сроку для приложения проверки синхронизации составляет 100 микросекунд. Этот срок отсчитывается с момента принятия решения о необходимости закрытия выключателя (событие, которое запускает приложение) до момента фактического замыкания контактов выключателя. Это жесткое ограничение реального времени накладывает серьезные ограничения на допустимое время обработки и передачи каждого сообщения в цепочке.

В идеале, учитывая критичность управления выключателем, каждое из сообщений в этой цепочке должно было бы проходить через различные проверки безопасности, прежде чем устройство-получатель отреагирует на него. Сообщения должны быть аутентифицированы для подтверждения их источника, проверены на предмет возможных искаженных или поддельных пакетов, их происхождение должно быть верифицировано, а полезная нагрузка проверена на наличие вредоносных программ или аномальных значений. Однако применение всех этих мер безопасности к каждому сообщению может внести значительную задержку, которая сделает невозможным соблюдение 100-микросекундного лимита.

Возникает нетривиальная задача оптимизации: какие подмножества этих функций безопасности могут быть применены к каждому из сообщений? Где в сети должно происходить это вмешательство - на конечных устройствах (IED, реле), на сетевом коммутаторе или на выделенном пограничном устройстве безопасности? Мы исследуем эти вопросы в остальной части работы, используя формальные методы, разработанные в данном документе.

Следует отметить, что на реальной подстанции существует множество IED, хостов и коммутаторов, а владелец передающей сети управляет несколькими подстанциями, которые взаимодействуют друг с другом и с центральным центром управления по глобальной сети (WAN). Это создает проблему огромного масштаба, где количество потенциальных потоков для мониторинга и возможных мест развертывания мониторов становится чрезвычайно большим, что делает задачу оптимизации еще более сложной и требует применения эффективных алгоритмов, представленных в данной работе.

2. Постановка задачи

2.1. Модель системы

Киберсеть моделируется как неориентированный граф $G=(V,E)$, где V - это набор $|V|$ сетевых элементов: коммутаторов, маршрутизаторов и хостов, и

где множество ребер E представляет связи между сетевыми элементами. Такое представление является стандартным для анализа сетевой инфраструктуры, поскольку оно абстрагируется от физических деталей реализации (таких как тип кабеля, пропускная способность канала или протоколы канального уровня) и фокусируется на топологической структуре, которая определяет возможные пути прохождения трафика. В контексте критически важных инфраструктур, таких как электрические подстанции, граф G обычно имеет относительно небольшой размер и характеризуется высокой степенью связности в пределах одной подстанции, при этом межподстанционные связи образуют разреженный граф, отражающий физическую топологию линий электропередачи.

Монитор - это устройство (развернутое либо в качестве промежуточного устройства, функции виртуальной сети, либо на хосте), которое может выполнять ряд функций безопасности (брандмауэр, система обнаружения/предотвращения вторжений, аутентификатор, средство проверки происхождения и т.д.) в потоках, которые ему назначено отслеживать. Каждый монитор имеет набор функций безопасности, которые он поддерживает. Следует подчеркнуть, что в реальных развертываниях мониторы могут быть реализованы как программные модули, работающие на существующем сетевом оборудовании (например, на коммутаторах с поддержкой OpenFlow или на выделенных серверах безопасности), так и как специализированные аппаратные устройства, включенные в разрыв сети. Выбор конкретной реализации зависит от требований к производительности, стоимости и доступности оборудования. Мониторы размещаются на линиях связи, предположение, которое допускает реализацию монитора в устройствах на конечных точках соединения или размещение устройств беспроводного мониторинга на линиях связи. Это предположение является ключевым для нашей модели, поскольку оно позволяет рассматривать мониторинг как операцию, привязанную к ребру графа, что значительно упрощает формализацию задачи. Мы предполагаем, что можно отслеживать только подмножество ссылок L , $L \subseteq E$. Это ограничение отражает практические соображения: не все каналы связи могут быть оборудованы мониторами из-за физических ограничений, стоимости развертывания или политики безопасности. Если монитор развернут на канале $l \in L$, то любой поток, проходящий по этому каналу, может быть проанализирован одной или всеми функциями безопасности на мониторе. Важно отметить, что мы не рассматриваем возможность частичного мониторинга потока (например, только заголовков пакетов) - предполагается, что если поток назначен на монитор, то мониторинг осуществляется в полном объеме для выбранных функций безопасности.

Стратегия мониторинга сети определяется как назначение A потоков в F этим $L \times M$ парам функций связи и безопасности, таким образом, что поток $f \in F$ отслеживается сетевой функцией $m \in M$ в канале $l \in L$. Другими словами, стратегия A является отображением из множества потоков в множество допустимых пар (канал, функция безопасности), при этом каждый поток может быть назначен на несколько пар, что позволяет реализовать многоуров-

невую защиту. Осуществимая стратегия мониторинга сети - это та, которая удовлетворяет всем ограничениям. К числу таких ограничений относятся:

- поток может быть назначен только на канал, через который он фактически проходит (топологическое ограничение);
- функция безопасности должна поддерживаться монитором, развернутым на данном канале (ограничение совместимости);
- суммарная задержка, вносимая мониторингом, не должна превышать предельный срок потока (временное ограничение);
- при наличии бюджетных ограничений - суммарная стоимость мониторинга не должна превышать заданный бюджет (экономическое ограничение).

Сетевой трафик в критически важных инфраструктурах часто очень хорошо изучен. Сетевые администраторы знают, какие источники сообщают о связи с каким пунктом назначения, они знают тип передаваемой информации (например, показания датчиков для агрегаторов данных и системные команды от станции управления для системных исполнительных механизмов). Они знают маршрут, по которому проходит каждый поток. Такая детерминированность трафика является следствием архитектурных принципов построения АСУ ТП (автоматизированных систем управления технологическим процессом), где взаимодействие между устройствами жестко регламентировано технологическими процессами и не допускает спонтанных изменений без санкции оператора. В отличие от корпоративных сетей, где трафик характеризуется высокой степенью случайности и непредсказуемости, в сетях критически важных инфраструктур преобладают регулярные, периодические потоки данных (например, телеметрия с периодом опроса 100 мс) и редкие, но критически важные события (например, команды релейной защиты). Разумно предположить знание набора F потоков трафика, каждое из описаний которых представляет собой кортеж {исходное устройство, целевое устройство, исходное приложение}, причем последний компонент необходим, если несколько приложений на исходном устройстве взаимодействуют с целевым устройством. Это требование обусловлено тем, что одно и то же физическое устройство (например, программируемый логический контроллер) может одновременно выполнять несколько функций: передавать телеметрию на сервер SCADA, принимать команды от станции управления и обмениваться служебной информацией с соседними контроллерами. Каждое из этих взаимодействий имеет свои характеристики критичности и требования к качеству обслуживания, поэтому они должны рассматриваться как отдельные потоки. Осознав это различие, в оставшейся части статьи мы оставляем описание исходного приложения, понимая, что мы можем преобразовать проблему, когда исходный код имеет несколько приложений, в проблему, когда каждый исходный код имеет только одно приложение, путем создания виртуальных источников, по одному на приложение. Такое преобразование является стандартным приемом в теории массового обслуживания и позволяет унифицировать описание потоков без потери общности.

Хотя разработка рациональных средств оценки относительных преимуществ мониторинга различных потоков с использованием разных мониторов

является работой, ортогональной нашей, мы описываем формулировку, которая позволяет использовать эти оценки для определения стратегии мониторинга сети. Важно понимать, что задача оценки критичности потока и выбора оптимальной функции безопасности для него является самостоятельной исследовательской проблемой, которая решается методами анализа рисков, теории принятия решений и экспертного оценивания. Наша задача - предоставить формальную основу, в которую могут быть интегрированы результаты таких оценок. Во-первых, ценность команд мониторинга для некоторых исполнительных механизмов будет выше, чем для других, в зависимости от важности исполнительного механизма для системы в целом и потенциального воздействия на систему передачи ему фальсифицированных команд. Например, команда на отключение высоковольтного выключателя имеет значительно более высокую критичность, чем команда на изменение уставки регулятора напряжения, поскольку неправильное срабатывание первого может привести к каскадному отключению электроэнергии, в то время как ошибка во втором случае может быть скорректирована автоматической системой регулирования. Поскольку не все потоки одинаково критичны, мы предполагаем существование функции w , которая количественно определяет с помощью положительного действительного числа $w(f)$ критичность мониторинга потока f . Значения $w(f)$ могут быть определены экспертным путем, на основе анализа последствий нарушения безопасности для каждого типа потока, или с использованием формальных методов, таких как анализ деревьев отказов и оценки ущерба. В общем случае $w(f)$ могут быть нормированы так, что их сумма по всем потокам равна единице, что позволяет интерпретировать их как вероятностную меру важности потока.

Во-вторых, различные функции безопасности $m \in M$ по-разному влияют на отслеживаемые потоки, в зависимости от конечных точек, данных, которые несет поток, и того, что именно способен обнаружить монитор. Рассмотрим потоки, передающие команды исполнительным механизмам, если функция безопасности может проверить целостность команды, то мониторинг может оказать большее влияние, чем если бы функция просто проверяла, что форматирование команд в потоке является допустимым в соответствии со спецификацией протокола. В первом случае монитор способен обнаружить попытку модификации команды злоумышленником, что является критически важным для предотвращения несанкционированных действий в системе. Во втором случае монитор может выявить только ошибки, вызванные случайными сбоями или неправильной работой оборудования, но не злонамеренные атаки. Аналогично, для потока телеметрии более важной может быть функция аутентификации источника, поскольку подмена данных от датчика может привести к неправильным решениям оператора или автоматической системы управления. Мы увидим, что постановка задачи в этой статье рассматривает возможность выбора одной или многих функций безопасности из M таким образом, что ценность мониторинга конкретного потока с помощью функции безопасности определенного типа зависит от атрибутов потока и функции. Пусть $\alpha_m(f)$ - значение потока контроля $f \in F$ с функцией безопасно-

сти $m \in M$. Функция $\alpha_m(f)$ может быть определена как математическое ожидание предотвращенного ущерба, как вероятность обнаружения атаки, как экспертная оценка или как комбинация этих факторов. В общем случае $\alpha_m(f)$ не обязана быть аддитивной по функциям безопасности, что отражает тот факт, что совокупный эффект от применения нескольких функций может быть как больше, так и меньше суммы отдельных эффектов.

Расстояние между каналом, на котором отслеживается поток, и его пунктом назначения также влияет на эффективность системы стратегия мониторинга. Рассмотрим поток f , который требуется для прохождения через систему предотвращения вторжений (IPS) и монитор происхождения данных. В случае IPS полезно для пропускной способности и безопасности сети, если поток отслеживается ближе к точке входа в сеть. Это объясняется тем, что IPS предназначена для блокирования вредоносного трафика до того, как он достигнет критически важных устройств. Если IPS размещена далеко от точки входа, атакующий трафик может распространиться по внутренней сети, заражая или нарушая работу промежуточных устройств, прежде чем будет заблокирован. Кроме того, раннее обнаружение атак позволяет снизить нагрузку на сеть, поскольку вредоносные пакеты не передаются дальше по сети, экономя пропускную способность каналов. Что касается проверки происхождения данных, то ценность мониторинга ближе к месту назначения значительно выше для обнаружения изменений данных внутри сети. Это связано с тем, что проверка происхождения данных предназначена для обнаружения модификаций, которые могут быть внесены в данные в процессе их передачи через сеть. Если мониторинг осуществляется вблизи источника, то проверяется только исходная целостность данных, но не их состояние после прохождения через промежуточные узлы. В то же время мониторинг вблизи пункта назначения позволяет обнаружить все модификации, внесенные на любом участке пути. Пусть $\gamma(d(l, f_t), m)$ - функция, которая количественно оценивает влияние применения функции безопасности m на звене l , которое находится в $d(l, f_t)$ переходах от пункта назначения f_t потока f . В общем случае функция γ может быть немонотонной, что отражает специфику различных функций безопасности: для одних функций (например, IPS) оптимальное положение - в начале пути, для других (например, проверка целостности) - в конце пути, а для третьих (например, анализ аномалий) - может быть равномерно распределенным по всему пути.

Наконец, некоторые функции безопасности могут извлечь выгоду из мониторинга потока в нескольких местах сети. Рассмотрим, например, систему обнаружения вторжений: традиционно IDS размещались в точке входа в сеть, однако в связи с растущей опасностью атак на уровне сетевых данных предпринимались попытки также разместить датчики IDS внутри системы. Многоуровневое размещение IDS позволяет обнаруживать атаки на разных этапах их развития: на этапе проникновения в сеть, на этапе распространения внутри сети и на этапе воздействия на целевые устройства. Каждый уровень обнаружения имеет свои сильные стороны: пограничная IDS эффективна против внешних атак, в то время как внутренняя IDS может обнаружить дей-

ствия легитимных пользователей, которые используют свои учетные записи в злонамеренных целях, или атаки, проникшие в обход пограничной защиты. Однако другие функции безопасности, такие как аутентификация, необходимо применять к потоку только один раз. Повторная аутентификация на каждом промежуточном узле не только не добавляет безопасности, но и может создать ложное чувство защищенности, поскольку злоумышленник может перехватить аутентификационные данные после первого успешного прохождения проверки. Кроме того, повторная аутентификация увеличивает задержку потока и вычислительную нагрузку на мониторы. Пусть β_m будет показателем того, выиграет ли монитор от повторного применения ($\beta_m=0$) или нет ($\beta_m=1$). Формально, если $\beta_m=1$, то поток f может быть назначен функции m не более чем на одном канале в стратегии A ; если $\beta_m=0$, то ограничений на количество назначений не накладывается.

2.2. Модель производительности монитора

Пусть $T \in R^{p \times q}$ обозначает временную матрицу, где $p=|F|$ и $q=|L \times M|$. Элемент T_{ij} представляет собой наихудшую задержку, с которой сталкивается поток $i \in F$ при функции безопасности $m \in M$ на канале $l \in L$. Задержка в наихудшем случае - это время, затраченное на обработку потока, учитывая, что все другие потоки, проходящие по каналу l , назначаются функции безопасности m . Формально, если обозначить через $S(l)$ множество всех потоков, проходящих через канал l , то T_{ij} вычисляется для наихудшего сценария, когда все потоки из $S(l)$ одновременно обрабатываются функцией безопасности m . Такая пессимистическая оценка оправдана в контексте критически важных инфраструктур, где даже редкие превышения допустимых задержек могут привести к тяжелым последствиям, вплоть до отказа системы управления. Хотя это очень консервативная оценка, большинство исследований систем реального времени использует этот подход и рассматривает время выполнения в наихудшем случае в качестве показателя для оценки ограничений по срокам. Альтернативные подходы, основанные на средних задержках или вероятностных гарантиях, могут быть более точными с точки зрения среднего поведения системы, но они не обеспечивают детерминированных гарантий, которые необходимы для систем с жесткими требованиями реального времени. Методы вычисления времени выполнения в наихудшем случае ортогональны нашей работе и широко изучаются в исследованиях систем реального времени. В литературе предложено множество подходов к оценке наихудшего времени выполнения, включая статический анализ кода, измерение времени выполнения на реальном оборудовании, моделирование на основе формальных моделей и комбинации этих методов. В нашей работе мы предполагаем, что значения T_{ij} известны или могут быть оценены с достаточной точностью для всех пар (поток, пара канал-функция).

Каждый поток $f \in F$ имеет крайний срок δ_f . Существуют определенные недостатки, которые имеют гибкие крайние сроки (определенные в диапазоне $\min$ - $\max$), в то время как большинство критических потоков имеют строгие верхние границы требований к срокам, поскольку несоблюдение этого крайнего срока может привести к нестабильности системы. В сетях критиче-

ски важных инфраструктур большинство потоков являются периодическими или спорадическими с известными максимальными частотами передачи, что позволяет определить жесткие требования к задержке. Например, поток телеметрии от измерительного трансформатора тока должен доставлять данные в центр управления не позднее чем через 10 мс после момента измерения, в противном случае данные могут быть признаны устаревшими и не использоваться для принятия решений о состоянии системы. Мы предполагаем, что все крайние сроки являются строгими, т.е. δ_f - это абсолютное последнее время, когда поток может прибыть в пункт назначения. Это предположение упрощает формализацию задачи и соответствует требованиям большинства протоколов, используемых в критически важных инфраструктурах, таких как IEC 61850 для подстанций или DNP3 для SCADA-систем. $\delta \in \mathbb{R}^{p \times 1}$ - вектор крайнего срока, где δ_i - крайний срок потока $i \in F$.

2.3. Задача поиска стратегии мониторинга сети

Учитывая сеть, потоки в сети и требования к мониторингу потоков, наша цель состоит в том, чтобы найти стратегию мониторинга сети, которая максимизирует качество мониторинга, получаемого по сети, при соблюдении сроков выполнения потоков. Формально мы должны определить стратегию отображения от F к $L \times M$, которая максимизирует выгоду от мониторинга, $Q(A)$. Мы определяем показатель качества мониторинга, $Q(A)$, формально как:

$$Q(A) = \sum_{f \in A(F)} w(f) \left[\sum_{m \in A_f(M)} \alpha_m(f) \left[\beta_m \left(\max_{l \in A_{f,m}(L)} \gamma(d(l, f_t), m) \right) + (1 - \beta_m) \left(\sum_{l \in A_{f,m}(L)} \gamma(d(l, f_t), m) \right) \right] \right] \quad (1)$$

где A - назначение потоков парам линк-монитор, а $A(F)$ - набор потоков в назначении, $A_f(M)$ - набор функций безопасности в A , которые выбраны для потока f , и $A_{f,m}(L)$ - набор потоков в A , где поток f контролируется функцией безопасности m . Первое слагаемое внутри скобок соответствует ценности мониторинга потока f функцией безопасности m на единственном канале (если $\beta_m=1$) или на оптимальном канале (если $\beta_m=0$ и требуется выбрать один канал). Второе слагаемое соответствует дополнительной ценности от мониторинга на нескольких каналах для функций с $\beta_m=0$. По сути, при заданном задании A , $Q(A)$ возвращает выгоду от мониторинга этой стратегии мониторинга, A . Функция $Q(A)$ является аддитивной по потокам, что позволяет рассматривать задачу оптимизации как задачу максимизации суммы индивидуальных вкладов, однако ограничения, связывающие потоки (например, ограничения на пропускную способность канала или бюджет), делают задачу нетривиальной.

Задача поиска стратегии сетевого мониторинга заключается в поиске подмножества местоположений для развертывания мониторов и выборе функций безопасности в мониторах для активации потоков, проходящих через них. Для формального понимания проблемы стратегии сетевого монито-

ринга рассмотрим следующую оптимизационную задачу:

$$\underset{A \subseteq L \times M}{\text{maximize}} \quad Q(A) \quad (2)$$

при ограничении

$$Tx_A \leq \beta$$

Текущая формулировка задачи позволяет добавить ограничение по затратам в дополнение к временным ограничениям. Это может, например, использоваться для описания денежных затрат на мониторинг потока. Может быть добавлена новая строка в матрице T , где каждый элемент представляет собой стоимость мониторинга в паре линк-монитор, скажем, из-за аналитика, который должен изучить сгенерированные предупреждения. Вектор ограничений δ будет иметь соответствующий бюджет, ниже которого должна быть общая стоимость. Такой подход позволяет моделировать различные практические сценарии: например, если организация имеет ограниченный бюджет на кибербезопасность и должна выбрать, какие потоки мониторить в первую очередь. Аналогично, если существует ограничение на общее количество пар линк-монитор, которые должны быть выбраны (чтобы уменьшить количество развернутых устройств), сужая T таким образом, чтобы $T_{1j}=1$ могло быть добавлено, а ограничение количества добавлено к δ . Это ограничение может быть актуально, когда количество доступных мониторов ограничено физически или когда администраторы хотят минимизировать сложность управления системой мониторинга.

Задача, описанная в уравнении 2, является NP-сложной, как мы покажем позже в этом разделе. Сложность задачи обусловлена комбинаторной природой выбора подмножества пар (канал, функция безопасности) для каждого потока при наличии глобальных ограничений. Даже частный случай, когда каждый поток может быть назначен только на одну пару, сводится к задаче о назначениях с дополнительными ограничениями, которая является NP-трудной. Предлагаемый нами алгоритм будет вычислять назначения жадным способом, и мы раскрываем и используем базовую структуру подмодульности задачи для разработки и анализа алгоритма принципиальным образом. Подмодульность функции $Q(A)$ означает, что предельная выгода от добавления новой пары (канал, функция безопасности) в стратегию мониторинга уменьшается с ростом уже выбранного множества пар. Это свойство позволяет применять жадные алгоритмы с гарантированными аппроксимационными коэффициентами, а также использовать методы ускоренного вычисления предельных выгод для повышения эффективности алгоритма на больших экземплярах задачи. В следующих разделах мы формально докажем свойства функции $Q(A)$ и представим наш алгоритм с анализом его гарантий качества.

2.4. Свойства $Q(A)$

Функция оценки стратегии мониторинга, формализованная в уравнении (1), по построению обладает рядом фундаментальных свойств, которые обуславливают возможность применения к ней методов дискретной оптимизации и теории субмодулярных функций. По своей конструкции $Q(A)$ неотрицательна, поскольку каждая из составляющих функций, входящих в её опре-

деление, принимает только неотрицательные значения. Это свойство является необходимым условием для корректной интерпретации $Q(A)$ как меры качества стратегии мониторинга: отрицательные значения не имели бы содержательного смысла в контексте оценки эффективности сетевого мониторинга. Кроме того, Q является нормализованной функцией, т.е. $Q(\emptyset)=0$, что означает: при отсутствии выбранных пар линк-монитор оценка стратегии равна нулю, что согласуется с интуитивным представлением о том, что без активированных средств мониторинга не может быть достигнуто никакого положительного эффекта. Помимо указанных тривиальных свойств, мы показываем, что $Q(A)$ обладает ещё двумя существенными для алгоритмического анализа свойствами: субмодулярностью и монотонностью. Эти свойства позволяют применять к задаче оптимизации стратегии мониторинга богатый арсенал теоретических результатов, разработанных для максимизации субмодулярных функций при линейных ограничениях.

Определение 1. При заданном конечном множестве X вещественнозначная функция f на множестве подмножеств X называется субмодулярной, если для любых подмножеств $A, B \subseteq X$ выполняется неравенство:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$$

В качестве эквивалентной альтернативы, субмодулярные функции могут быть определены посредством свойства уменьшения предельных значений (diminishing returns), которое формулируется следующим образом:

$$f(A \cup \{x\}) - f(A) \leq f(B \cup \{x\}) - f(B), \text{ " } \forall A \subseteq B \subseteq X, x \in X \setminus A$$

Выражение $f(A \cup \{x\}) - f(A)$ известно как добавочный предельный доход от элемента x при заданном множестве A и обозначается как $f_A(x)$. Свойство уменьшения предельных значений интуитивно интерпретируется следующим образом: предельный вклад любого элемента в значение функции не возрастает по мере расширения множества уже выбранных элементов. Иными словами, чем больше элементов уже включено в решение, тем меньший дополнительный эффект приносит добавление нового элемента. Это свойство отражает явление насыщения, характерное для многих практических задач, включая задачи сетевого мониторинга, где добавление новых пар линк-монитор при уже существующем обширном покрытии даёт всё меньший прирост в качестве мониторинга.

В литературе по алгоритмам аппроксимации определяется обширный класс задач, целью которых является максимизация монотонной субмодулярной функции при наличии нескольких линейных ограничений упаковочного типа (packing constraints). Эти задачи имеют следующую стандартную форму:

$$\max f(S) \text{ такой, что } Ax_S \leq b \text{ и } S \subseteq [n]$$

где $A \in [0, 1]^{m \times n}$, $b \in [1, \infty)^m$ и x_S - характеристический вектор множества S .

Каждый элемент A_{ij} матрицы A интерпретируется как стоимость включения элемента j в решение относительно ограничения i , а каждый элемент b_i вектора b представляет собой бюджетное ограничение для i -го ресурса. Наиболее распространённым и хорошо изученным примером задачи, принимающей такую форму, является классическая задача о рюкзаке (knapsack

problem), в которой целевая функция является модульной, то есть аддитивной по элементам. Однако задачи с субмодулярными целевыми функциями представляют значительно больший интерес с теоретической и практической точек зрения, поскольку они охватывают широкий спектр прикладных ситуаций, включая задачи выбора сенсоров, задачи влияния в социальных сетях, задачи покрытия и многие другие. Дополнительным важным свойством, которым должна обладать целевая функция для применимости разработанных алгоритмов, является монотонность.

Определение 2. Функция f , определённая на множестве подмножеств, называется монотонной (невозрастающей по включению), если для любых $B \subseteq A$ выполняется неравенство $f(B) \leq f(A)$.

Функция на множествах называется монотонной неубывающей, если при расширении аргументного множества путём добавления одного или нескольких новых элементов значение функции никогда не уменьшается в результате такого включения. Иными словами, увеличение объёма выбранных элементов не приводит к ухудшению значения целевой функции. Ясно, что это свойство выполняется для функции Q , поскольку добавление новой пары линк-монитор к заданию либо улучшит текущую оценку мониторинга, либо, в крайнем случае, не изменит её (если $w(f) = 0$ для всех потоков f , затронутых данным добавлением, или $\alpha_m(f) = 0$ для всех $m \in A(M)$, то есть добавление не приносит дополнительной пользы). Формально это следует из того, что каждая компонента функции Q является неубывающей по включению пар линк-монитор в задание, а сумма неубывающих функций также является неубывающей функцией.

Теорема 2.1. Функция $Q: 2^{L \times M} \rightarrow \mathbb{R}^+$, определённая в уравнении (1), является субмодулярной.

Доказательство. Для доказательства субмодулярности функции Q рассмотрим два произвольных назначения $A \subseteq A' \subseteq L \times M$. Чтобы установить субмодулярность, согласно определению через предельные значения, необходимо показать, что для любой пары $(l_0, m_0) \notin A'$ выполняется неравенство:

$$Q_A(l_0, m_0) = Q(A \dot{\cup} (l_0, m_0)) - Q(A) \geq Q_{A'}(l_0, m_0).$$

Когда добавляется новая пара линк-монитор (l_0, m_0) , отслеживаются все потоки, которые проходят через нее. Если монитор имеет тип $\beta_{m_0} = 0$, то функция является модульной (линейной), поскольку эффект добавления пары совпадает для обоих назначений A и A_0 . Более интересным случаем является случай, когда $\beta_{m_0} = 1$. Когда этот тип монитора существует, может быть три случая.

При добавлении новой пары линк-монитор (l_0, m_0) в задание происходит отслеживание всех потоков, которые проходят через данный линк и удовлетворяют условиям, определяемым типом монитора. Ключевым фактором, определяющим характер предельного вклада, является тип монитора β_{m_0} . Если монитор имеет тип $\beta_{m_0} = 0$, то функция ведёт себя как модульная (линейная), поскольку эффект от добавления пары оказывается одинаковым для обоих назначений A и A' . Это связано с тем, что мониторы типа $\beta = 0$ обеспе-

чивают фиксированный прирост оценки, не зависящий от того, какие другие пары уже включены в задание.

Более интересным и содержательным является случай, когда $\beta_{m_0} = 1$. В этой ситуации монитор способен обнаруживать все потоки, проходящие через линк, однако степень его полезности может зависеть от уже существующего покрытия. При наличии монитора такого типа могут возникнуть три различных случая, каждый из которых необходимо рассмотреть отдельно.

Случай 1: Новый линк не максимизирует значение γ ни для назначения A , ни для назначения A' . В этом случае предельные вклады равны: $Q_{A'}(l_0, m_0) = Q_A(l_0, m_0)$, что делает функцию модулярной на данном участке. Интуитивно это означает, что добавление пары (l_0, m_0) приносит одинаковый прирост независимо от того, какое из двух назначений рассматривается, поскольку в обоих случаях данный линк не является критическим с точки зрения максимизации покрытия.

Случай 2: Новый линк максимизирует значение γ для назначения A , но не для назначения A' . В данной ситуации предельный вклад для большего назначения A' оказывается нулевым: $Q_{A'}(l_0, m_0) = 0$, тогда как для меньшего назначения A предельный вклад строго положителен: $Q_A(l_0, m_0) > 0$. Это делает функцию строго субмодулярной, поскольку предельный доход от добавления элемента уменьшается с ростом базового множества. Содержательно такая ситуация возникает, когда в назначении A' уже присутствует пара, обеспечивающая лучшее покрытие того же линка, и добавление (l_0, m_0) не даёт дополнительной информации.

Случай 3: новый линк максимизирует γ для обоих назначений. В этом случае $Q(A' \dot{\cup} (l_0, m_0)) = Q(A \dot{\cup} (l_0, m_0)) > 0$, но из монотонности мы знаем, что $Q(A') \geq Q(A)$, таким образом,

Случай 3: Новый линк максимизирует значение γ для обоих назначений A и A' . В этом случае предельные вклады связаны соотношением: $Q(A' \dot{\cup} (l_0, m_0)) = Q(A \dot{\cup} (l_0, m_0)) > 0$, однако из свойства монотонности мы знаем, что $Q(A') \geq Q(A)$. Таким образом, получаем:

$$Q(A \dot{\cup} (l_0, m_0)) - Q(A) \geq Q(A' \dot{\cup} (l_0, m_0)) - Q(A')$$

что в точности соответствует определению субмодулярности через предельные значения. Рассмотрение всех трёх случаев показывает, что для произвольной пары (l_0, m_0) и произвольных назначений $A \subseteq A'$ выполняется требуемое неравенство, следовательно, функция Q является субмодулярной. ■

Установленная субмодулярность функции Q имеет фундаментальное значение для разработки эффективных алгоритмов решения задачи мониторинга. Как показано в работах Немхаузера, Вольси и Фишера, а также в более поздних исследованиях Краузе и Головина, монотонные субмодулярные функции допускают эффективную аппроксимацию с гарантированной точностью посредством жадных алгоритмов.

Лемма 2.2. Задача определения оптимальной стратегии мониторинга сети является NP-сложной.

Доказательство. Максимизация монотонной субмодулярной функции с линейными ограничениями упаковочного типа является NP-сложной задачей,

что установлено в классической работе [2]. Поскольку задача мониторинга сети сводится к максимизации функции Q (которая, как показано в теореме 2.1, является монотонной и субмодулярной) при наличии линейных ограничений на ресурсы (представленных бюджетными ограничениями, вытекающими из требований к времени выполнения потоков), она также является NP-сложной. Формально, любая задача максимизации монотонной субмодулярной функции с линейными ограничениями может быть сведена к задаче мониторинга сети путём соответствующего выбора параметров матрицы T и вектора крайних сроков δ , что и устанавливает NP-сложность рассматриваемой задачи [2]. ■

Следует отметить, что NP-трудность задачи не делает её неразрешимой на практике: она лишь указывает на то, что точное решение за полиномиальное время невозможно (при условии $P \neq NP$), однако аппроксимационные алгоритмы с гарантированными коэффициентами могут быть разработаны и успешно применены.

3. Алгоритм

Предлагаемое решение задачи определения стратегии мониторинга сети описано в алгоритме 1. Данный алгоритм принимает в качестве входных данных матрицу времени наихудшего случая T , элементы которой характеризуют задержку, вносимую парой линк-монитор j для потока f , а также вектор крайних сроков δ , компоненты которого задают предельно допустимое время выполнения каждого потока. Выходными данными алгоритма является назначение $A \subseteq L \times M$, определяющее, какие пары линк-монитор должны быть активированы для обеспечения требуемого уровня мониторинга при соблюдении всех ограничений на время выполнения.

Алгоритм начинает работу с пустого назначения $A = \emptyset$ и выполняется циклически (в цикле `while`), при этом на каждой итерации цикла к текущему назначению добавляется одна пара линк-монитор, выбранная в соответствии с жадным критерием. Цикл завершает свою работу при выполнении одного из двух условий: (1) текущее назначение A таково, что нарушается крайний срок выполнения хотя бы одного потока, то есть существует поток $f \in F$, для которого суммарная задержка, вносимая выбранными парами, превышает δ_f ; (2) все потоки были полностью проконтролированы, то есть для каждого потока f достигнуто максимально возможное значение функции $\alpha_m(f)$ при выбранных мониторах. Первое условие обеспечивает соблюдение всех ограничений, второе - гарантирует завершение алгоритма даже в случае, когда ограничения не являются активными.

Жадный шаг алгоритма устроен таким образом, что на каждой итерации выбирается пара линк-монитор j , которая максимизирует соотношение полезности и затрат. Напомним, что $Q_{A(j)}$ обозначает добавочное предельное значение добавления пары линк-монитор j к заданию A , то есть $Q_{A(j)} = Q(A \cup \{j\}) - Q(A)$. Стоимость добавления пары j определяется с учётом текущих весов ограничений и вычисляется как сумма по всем потокам величин $w_f \cdot T_{fj}$, где T_{fj} - задержка, вносимая парой j для потока f . Такой подход является стандартным для задач максимизации субмодулярных функций с линейными

ограничениями и обобщает классический жадный алгоритм для задачи о рюкзаке.

Как только пара линк-монитор будет выбрана в цикле, необходимо обновить веса w_f для ограничений. Такое обновление гарантирует, что ограничения, приближающиеся к своим предельным значениям, получают экспоненциально возрастающие веса, что в свою очередь делает их более «дорогими» в жадном критерии и тем самым предотвращает их нарушение. Выбор параметра λ является критическим для гарантии качества аппроксимации, как будет показано ниже.

Для учёта множества линейных ограничений, вытекающих из требований к времени выполнения потоков, необходимо присвоить веса каждому ограничению и рассматривать линейную комбинацию затрат как истинную стоимость каждого шага алгоритма. Инициализируется вес каждого ограничения следующим образом: $w_f = 1/\delta_f$ для каждого потока $f \in F$. Такая инициализация означает, что потоки с самыми дальними (наибольшими) крайними сроками имеют меньший вес, и наоборот: потоки с более жёсткими ограничениями по времени получают больший вес. Формально такое взвешивание отражает, насколько близко ограничение находится к тому, чтобы быть нарушенным при данном решении: чем ближе суммарная задержка к предельному значению δ_f , тем больше вес соответствующего ограничения, что делает его более чувствительным к дальнейшему добавлению пар.

Algorithm 1: Greedy monitor deployment algorithm

Input : Worst case timing matrix T , deadlines δ ,
update factor λ
Output: A set of link monitor pairs $A \subseteq L \times M$ with
feasible deadlines

$A \leftarrow \{\}$;
 $w_f \leftarrow 1/\delta_f$ for all $f \in F$;
 $k \leftarrow 1$;
while $\sum_{f \in F} \delta_f w_f \leq \lambda$ **and** $A \neq L \times M$ **do**
 $j \leftarrow \operatorname{argmax}_{j \in (L \times M) \setminus A} \frac{Q_A(j)}{\sum_{f \in F} T_{fj} w_f}$;
 $A \leftarrow A \cup \{j\}$;
 $w_f \leftarrow w_f \lambda^{T_{fj}/\delta_f}$ for all $f \in F$;
 $k \leftarrow k + 1$;
 if $T x_A \leq \delta$ **then**
 | return A ;
 else
 | return $A \setminus \{j\}$;
 end
end

Теорема 3.1. Алгоритм 1 аппроксимирует оптимальный алгоритм в пределах коэффициента $(1-e)(1-1/e)$, т.е. $ALG1 \geq (1-e)(1-1/e)$ для некоторого фиксированного значения $e > 0$.

Для обоснования этой теоремы введём вспомогательную величину $\eta = \min\{\delta_f/T_{fj}\}$. Отметим, что $T_{fj} > 0$ всегда выполнено, поскольку задержка, добавляемая парой линк-монитор, по своему физическому смыслу всегда отлична от нуля - любой мониторинг вносит ненулевую дополнительную за-

держку в обработку потока. Величина η по существу представляет собой коридор ограничений: она показывает, насколько велик запас между крайними сроками и минимальными задержками, вносимыми парами. Чем больше η , тем более «свободны» ограничения, и тем легче алгоритму найти допустимое решение с высоким качеством.

Теорема 1.1 из работы [1] утверждает, что при использовании мультипликативного правила обновления весов жадный алгоритм достигает гарантии аппроксимации $\Omega(1/|F|^{1/\eta})$, и эта гарантия является константой, поскольку число ограничений $|F|$ является фиксированным для данной матрицы T . Иными словами, для любой конкретной задачи с заданными параметрами коэффициент аппроксимации не зависит от размера входных данных, что делает алгоритм пригодным для практического применения. При этом коэффициент обновления λ играет ключевую роль в гарантии аппроксимации: от его выбора зависит, насколько быстро растут веса ограничений и, следовательно, насколько точно алгоритм соблюдает бюджетные ограничения при максимизации целевой функции.

В общей настройке можно использовать коэффициент обновления $\lambda = e^\eta |F|$ (см. лемму 2.4 в [1]), чтобы получить приближение с постоянным коэффициентом $\Omega(1/|F|^{1/\eta})$. Однако для достижения более точной гарантии, сформулированной в теореме 3.1, необходимо выбрать коэффициент обновления более тщательно. Конкретно, коэффициент аппроксимации $(1 - \varepsilon)(1 - 1/e)$ достигается, когда в алгоритме 1 используется коэффициент обновления $\lambda = e^{\varepsilon n}/4$, и когда рассматриваемый вариант задачи таков, $\eta = \Omega(\ln |F|/\varepsilon^2)$ для любого фиксированного $\varepsilon > 0$. Это условие выполняется, когда временные матрицы T являются плотными с небольшим смещением, то есть когда задержки T_{fj} для всех пар (f, j) имеют сравнимые значения и не слишком сильно различаются между собой. В таких случаях коридор ограничений η достаточно велик, что позволяет алгоритму достигать почти оптимального качества решения.

4. Заключение

В данной работе была формализована проблема поиска стратегии мониторинга сети как задача определения множества сетевых соединений и функций безопасности, которые необходимо активировать для обеспечения мониторинга заданных потоков при соблюдении ограничений на время выполнения. Мы показали, что эта проблема является NP-сложной, что обосновывает необходимость разработки аппроксимационных алгоритмов. Был определён новый показатель безопасности - функция оценки $Q(A)$, которая может быть использована для количественной оценки качества стратегии мониторинга. Используя специальные свойства этой метрики, а именно её субмодулярность и монотонность, мы показали, что задача может быть решена с помощью жадного алгоритма за полиномиальное время с фиксированной границей аппроксимации. Предложенный алгоритм основан на классическом подходе к максимизации субмодулярных функций с линейными ограничениями и использует мультипликативное обновление весов ограничений для обеспечения соблюдения бюджетных ограничений. Теоретический анализ демонст-

рирует, что алгоритм достигает гарантии аппроксимации $(1 - \varepsilon)(1 - 1/e)$ при выполнении определённых условий на структуру временных матриц, что делает его пригодным для практического применения в задачах мониторинга сетевой безопасности.

Перспективными направлениями будущих исследований являются:

- адаптация предложенного алгоритма к динамическим сценариям, где множество потоков и требования к мониторингу изменяются во времени;
- исследование возможностей распараллеливания вычислений для ускорения работы алгоритма на больших сетях;
- разработка более точных оценок параметра η для специальных классов сетевых топологий;
- экспериментальное исследование эффективности предложенного алгоритма на реальных данных сетевого трафика.

Список использованных источников

1. Azar Y., Gamzu I. Efficient submodular function maximization under linear packing constraints// International Colloquium on Automata, Languages, and Programming. - Springer, 2012, p. 38-50.
2. Krause A., Golovin D. Submodular function maximization// Tractability, 2014.

Гетманская Д.В.

АЛГОРИТМ ВЫБОРА ПРИЗНАКОВ БОЛЬШИХ НАБОРОВ ДАННЫХ С ВЫСОКОЙ РАЗМЕРНОСТЬЮ И НЕСБАЛАНСИРОВАННОСТЬЮ, ОСНОВАННЫЙ НА ГРАНУЛЯРНОМ СЛИЯНИИ

Воронежский государственный технический университет

1. Введение

Селекция признаков, являясь одним из ключевых этапов предварительной обработки данных, находит широкое применение в областях машинного обучения и распознавания образов [1]. Интенсивное развитие сетевых инфраструктур и технологий сбора данных обуславливает непрерывное появление наборов данных сверхвысокой размерности, характеризующихся дисбалансом классов (несбалансированные данные). Подобные данные обычно содержат значительное число избыточных и неинформативных независимых признаков, что существенно осложняет процесс выбора релевантных переменных. Большой объем обрабатываемой информации оказывает критическое влияние на вычислительную эффективность алгоритмов селекции, при этом стандартные вычислительные средства зачастую неспособны обеспечить загрузку и обработку полного массива данных в оперативной памяти. Таким образом, разработка более эффективных алгоритмов выбора признаков для массивных несбалансированных данных высокой размерности приобретает важное теоретическое и прикладное значение. Теория гранулярных вычислений слияния (granular fusion computing) представляет собой методологию неточного решения задач, связанных с анализом больших данных. Обеспечивая сохранение информационной ценности данных, данная теория позволяет уменьшить масштаб обрабатываемых данных, трансформируя ис-

ходный большой массив в множество информационных гранул (фрагментов). Ввиду способности к существенному сокращению объема данных [2], применение теории гранулярных вычислений слияния для крупномасштабной обработки информации привлекает внимание значительного числа исследователей. В работе Liang и др. для гранулирования больших данных на облачной платформе применялась технология кластеризации, что позволило снизить потери информации и повысить эффективность использования временных и вычислительных ресурсов [3]. Yuan и др. для обработки крупномасштабных временных рядов использовали метод гранулирования нечеткой информации, применив для регрессионного анализа и прогнозирования отдельных гранул метод опорных векторов, что повысило скорость анализа временных рядов [4].

Основываясь на результатах вышеупомянутых исследований, в данной работе предлагается алгоритм выбора признаков, основанный на гранулярном синтезе. Используя преимущества теории гранулярного синтеза, процесс отбора информативных переменных из высокоразмерных несбалансированных наборов данных преобразуется в последовательность операций выбора признаков из небольших рассеянных фрагментов данных. Экспериментальная валидация эффективности разработанного алгоритма показывает, что метод, основанный на слиянии гранул, наследует преимущества традиционных подходов, при этом его точность и скорость воспроизведения (recall) превосходят аналогичные показатели традиционных алгоритмов. Полученные результаты свидетельствуют об эффективности и практической применимости предложенного алгоритма.

2. Разработка алгоритма выбора признаков для многомерных данных на основе гранулярного синтеза

В ответ на существенные ограничения традиционных алгоритмов выбора признаков, возникающие при обработке массивов высокоразмерных несбалансированных данных, в данной работе предложен алгоритм селекции признаков, основанный на теории гранулярного синтеза. Алгоритм структурно разделен на три ключевых этапа: гранулирование (Granulation), выбор признаков на основе гранулированных данных и объединение гранулированных признаков для последующей селекции [5]. Ниже представлено детальное описание каждого этапа.

2.1. Обработка крупномасштабных данных для гранулирования

При осуществлении выбора признаков из многомерного несбалансированного набора данных на основе теории слияния гранул, для процесса гранулирования применяется статистическая теория случайной выборки [6]. На начальном этапе определения размера набора данных производится расчет дисперсии для всего массива данных. Впоследствии, в соответствии с атрибутивными характеристиками массивных данных, для процедуры гранулирования применяется теория иерархической выборки [7]. После завершения процесса гранулирования высокоразмерный несбалансированный набор данных представляется в виде, схематически иллюстрированном на рис. 1.

В данной работе гранулирование массивных данных осуществляется на

основе теории слияния гранул. Размер гранул (particle size) может быть вычислен непосредственно, исходя из размера исходного набора данных N и заданных параметров r , что позволяет существенно повысить эффективность процедуры гранулирования массивных данных. При этом объем данных, содержащихся в каждой отдельной грануле, значительно сокращается, а возможность последующего использования гранул данных повышается, обеспечивая оптимизацию вычислительной среды [8].

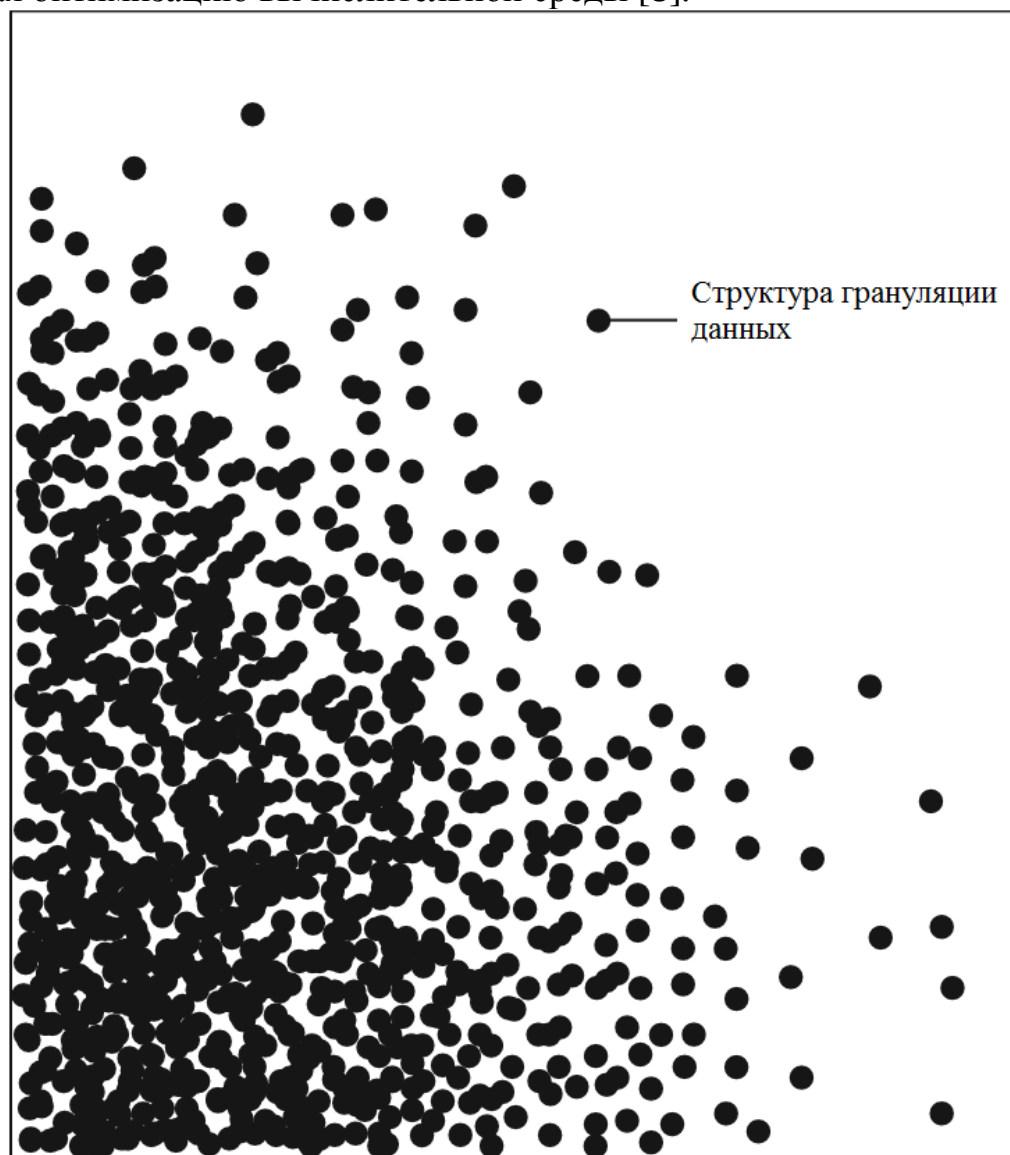


Рис. 1. Представление гранулированных больших наборов данных

На этапе вычислений вводится набор данных X с размером выборки N , после чего вычисляются собственные значения P гранул данных в соответствии с выражением (1):

$$p = \frac{X - X_p}{N^r} \quad (1)$$

p представляет собственные значения гранул данных. Согласно теории слияния гранул, диапазон параметра r следующий: $0.5 \leq r \leq 0.9$.

Приведенный выше расчет реализует процесс гранулирования больших наборов неравновесных данных с высокой размерностью, в котором собственное значение p задействованных фрагментов данных может быть вычис-

лено как параметр выбора признака. Анализ показывает, что значение p определяется с помощью X_p и N_p , и получается точное значение p , затем выполняется резервный вывод, который обеспечивает основу для расчета характеристик зернистости данных на следующем шаге.

2.2. Выбор объекта - частицы данных

В соответствии с исходным пространством признаков генерируется подмножество потенциальных признаков, соответствующих гранулам данных, которое служит входными данными для алгоритма выбора признаков [9]. Для поиска признаков-объектов начальная точка поиска детерминирует его направление. При инициализации поиска с пустого набора H_i , последующий процесс представляет собой последовательное добавление выбранных объектов в подмножество кандидатов, при этом функциональное выражение прямого поиска имеет вид (2):

$$H_i = \frac{x|pw_i|}{q_1} + \frac{x|pw_i|}{q_2} + \dots + \frac{x|pw_i|}{q_n} \quad (2)$$

Здесь H представляет собой условие прямого поиска в пустом наборе H_i ; $\frac{x|pw_i|}{q_1}$ представляет степень детализации несбалансированного коэффициента x , когда вероятность точки рассеяния равна 1. Таким же образом, $\frac{x|pw_i|}{q_n}$

представляет степень детализации коэффициента неравновесия x , когда вероятность точки рассеяния равна n .

Для предотвращения попадания в локальный оптимум, что характерно для линейных стратегий поиска, вычисление ограничений для условия прямого поиска H реализуется с использованием стратегии случайного поиска. Предположим, что исходный набор признаков содержит N_q векторов признаков, тогда подмножество признаков-кандидатов может содержать $2N_q$ векторов. При $H \geq 2N_q$, используя исчерпывающий метод поиска, осуществляется доступ ко всем подмножествам объектов в соответствии с заданным направлением, целевое подмножество отмечается, и определяются критерии оценки. В случае $H < 2N_q$, для получения оптимальных результатов для каждого подмножества признаков применяется полный поиск, а также вычисляется дополнительный коэффициент балансировки данных с использованием N_q . На данной основе в набор потенциальных признаков добавляется вектор признаков, при этом случайным образом удаляется не векторный признак, что повышает эффективность выбора признаков в алгоритме. Одновременно нивелируется неопределенность, обусловленная высокой вычислительной сложностью [10-13].

Посредством вышеуказанных вычислений получается начальный набор признаков N_q для гранул данных, при этом N_q используется исключительно в качестве референтного коэффициента для подготовки к расчетам слияния и последующей селекции подмножеств признаков.

2.3. Вычисление слияния подмножеств объектов

Оценка подмножества объектов является одним из ключевых этапов в процессе выбора признаков для большого набора данных. Каждое потенциальное подмножество объектов должно быть оценено в соответствии с заданными критериями. Благодаря интеграции соответствующего содержания теории гранулярного синтеза, процесс вычисления слияния для выбора признаков из больших массивов данных трансформируется в процесс оценки подмножества признаков. Схематическое представление формы оценки слияния для подмножества показано на рис. 2.

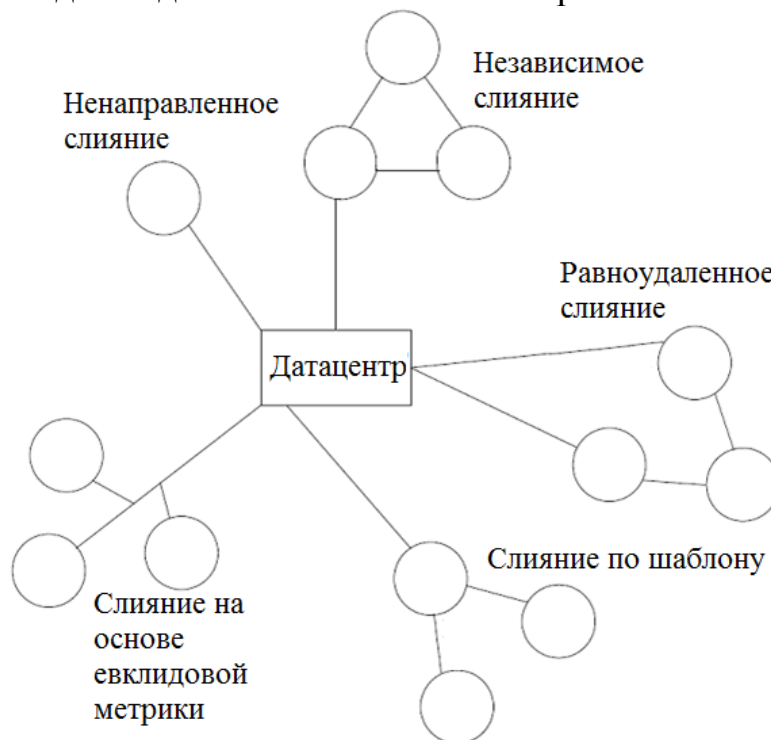


Рис. 2. Варианты слияния подмножеств данных

Рисунок 2 иллюстрирует, что слияние подмножеств гранул данных в основном включает следующие типы: однонаправленное слияние, слияние по шаблону, слияние на основе евклидова расстояния, слияние на основе равного расстояния и независимое слияние. Введем вышеприведенную модель в процесс вычисления и обозначим ее как U_i , где $U_i = \{u_1, u_2, u_3, \dots, u_i\}$. Независимый стандарт и стандарт слияния объединяются для оценки характеристик по внутренним атрибутам данных.

Одновременно, в соответствии с конкретным алгоритмом обучения, критерий расстояния используется для измерения сходства между данными выборки, что позволяет представить вклад или эффективность признаков в задачи классификации и распознавания. Селекция признаков с использованием мер расстояния, как правило, базируется на следующих допущениях.

Выборки, относящиеся к различным классам подмножеств признаков, принимаются в качестве критерия расстояния f_1 , а коэффициенты абсолютных значений f_1 вычисляются и измеряются в виде $f_1 \otimes f_n$. Если обнаруживается, что данный набор мер является нулевым множеством, то есть $f_1 \otimes f_n = 0$, передача многомерных данных прекращается, и продолжается ожидание

функции интеллектуального анализа данных в буфере до тех пор, пока $f_1 \otimes f_n \neq 0$, после чего вычисляется коэффициент отбора многомерных неравновесных данных, не являющихся гранулами, согласно выражению (3):

$$s_i = \frac{1}{f_1 \otimes f_n - 1} \sum_{j=1}^i U_{ij} \quad (3)$$

Здесь σ - характерные коэффициенты отбора крупномерных неравновесных данных; u_{ij} - характеристики информационной энтропии данных.

Если выбранный признак большого массива данных может быть ограничен значением g , это свидетельствует о наличии у признака данных метрики и возможности его эффективного восстановления. При этом критерий корреляции g применяется для выбора подмножества признаков данных. Если выбираемый признак больших данных не соответствует ограничивающему критерию g , производительность измерения объекта данных оценивается как низкая, эффективная частота отзыва недостаточна, в связи с чем выбор признака отменяется, а диапазон селекции переопределяется до тех пор, пока выбранный признак не будет удовлетворять ограничению g . Таким образом, завершается процесс отбора признаков для большого набора несбалансированных данных высокой размерности на основе коэффициента корреляции между признаками.

3. Эксперимент

Для подтверждения достоверности разработанного алгоритма выбора характеристик, основанного на грануляции, был проведен экспериментальный анализ. Условия проведения эксперимента детально представлены в табл. 1.

Таблица 1

Информация о конфигурации среды разработки

Имя параметра	Конфигурация
Операционная система	Microsoft Windows 10
Процессор	Intel(R)Celeron(R) 2.6 GHz
Memory	8.0 GB
Жесткий диск	400 GB
СУБД	Microsoft SQL Server 2010 R2
JDK	1.6
Математический пакет	MATLAB

Объектом эксперимента является высокоразмерный несбалансированный и объемный набор данных, для которого проводится проверка тестируемости. Результаты проверки соответствуют установленным стандартам, что демонстрирует практическую значимость экспериментальных данных. Для обеспечения точности эксперимента для сравнительного анализа используется традиционный алгоритм выбора признаков данных, также вычисляются показатели точности и скорости воспроизведения для обоих алгоритмов. В данном эксперименте исследуются точность и частота повторения выбора признаков из многомерного несбалансированного большого набора данных с использованием двух алгоритмов, при этом процесс эксперимента контролируется с помощью шкалы Фишера, что гарантирует наглядность и объектив-

ность результатов.

Формулы для расчета коэффициента точности и скорости воспроизведения при выборе признаков большого набора данных имеют вид (4) и (5):

$$g = \frac{A}{(A + B)^2} \quad (4)$$

$$l = \frac{C}{(C + B)^2} \quad (5)$$

Здесь γ представляет собой коэффициент точности выбора характеристики для большого набора данных, λ представляет частоту повторных вызовов, выбранную для функции большого набора данных, γ и λ учитываются в процентах; A представляет количество ограничений, представляющих данные; B представляет коэффициенты кластеризации с отрицательными ограничениями данных; C представляет параметр набора данных; n представляет собой константу, количество экспериментов.

Экспериментальный процесс организован следующим образом: на первом этапе с использованием двух типов алгоритмов выбора признаков производится селекция наиболее ценного подмножества признаков, затем исходные данные проецируются в подпространство малоразмерных признаков для кластеризации, для которой применяется алгоритм простого среднего значения. На заключительном этапе результаты выбора признаков из больших наборов данных модифицируются и создаются их резервные копии.

Сравнение частоты воспроизведения (recall) выбора признаков между традиционным и новым алгоритмами для большого набора несбалансированных данных высокой размерности представлено в табл. 2.

Таблица 2

Сравнение максимальной частоты отзывов объектов набора данных (%)

Набор данных	Образец	Признаки	Категории	Традиционный алгоритм	Новый алгоритм
Heart	270	16	2	66.7	74.6
Sphere	362	36	2	72.1	92.3
Sonar	246	49	3	59.6	86.4
Digits	189	16	4	78.3	91.6
Wine	4563	42	8	92.1	99.6
Image	961	26	9	59.3	86.4
Zoo	143	15	10	89.3	98.9

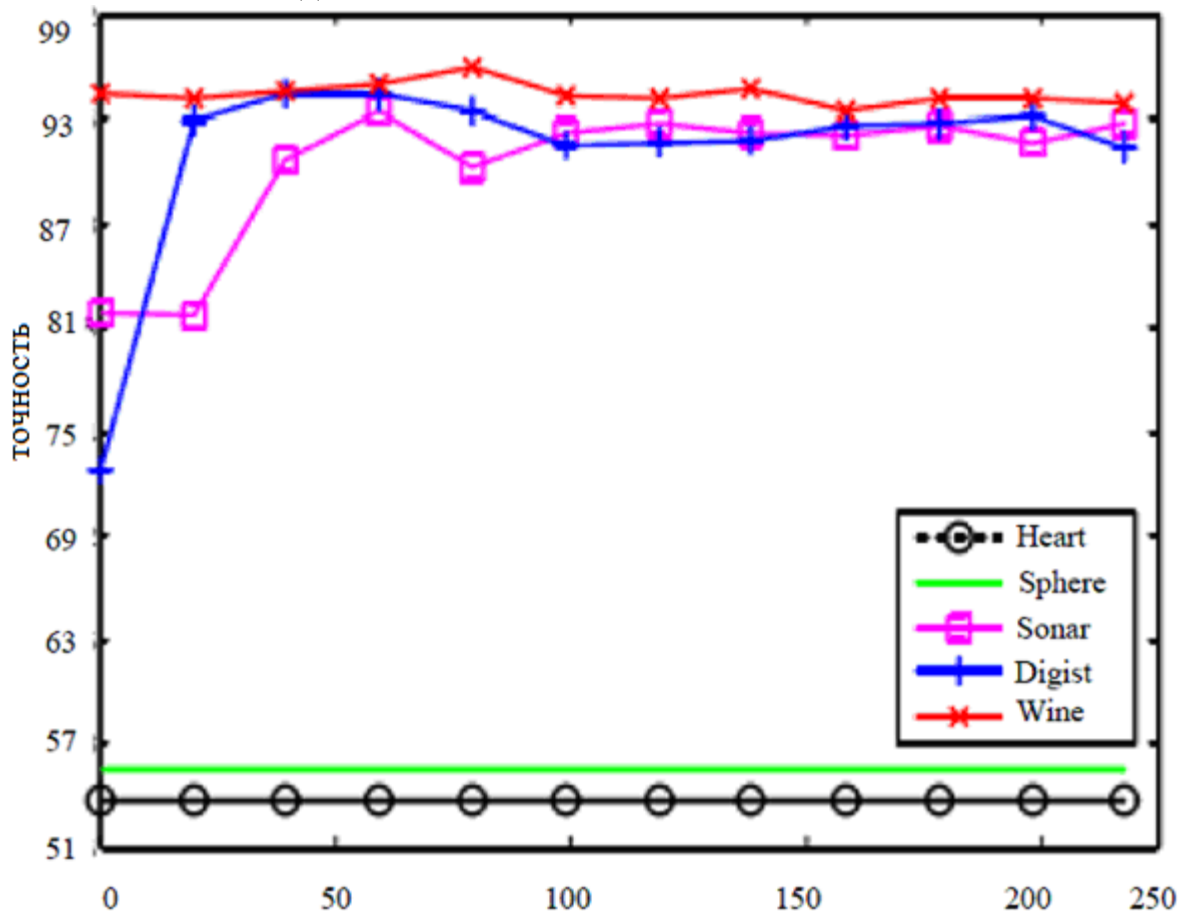
Из данных, представленных в табл. 2, следует, что производительность предложенного алгоритма превосходит показатели традиционного алгоритма выбора признаков. Несмотря на то, что значения данных ниже характерного значения информации об ограничениях, эффективность кластеризации данных значительно улучшается после проведения глобального или локального мониторинга. Полученные результаты свидетельствуют о том, что алгоритм выбора признаков данных, основанный на гранулярном слиянии, позволяет эффективно использовать теорию гранулярного синтеза и неконтролируемую информацию для селекции признаков, повышая скорость восстановления

данных и подтверждая эффективность алгоритма.

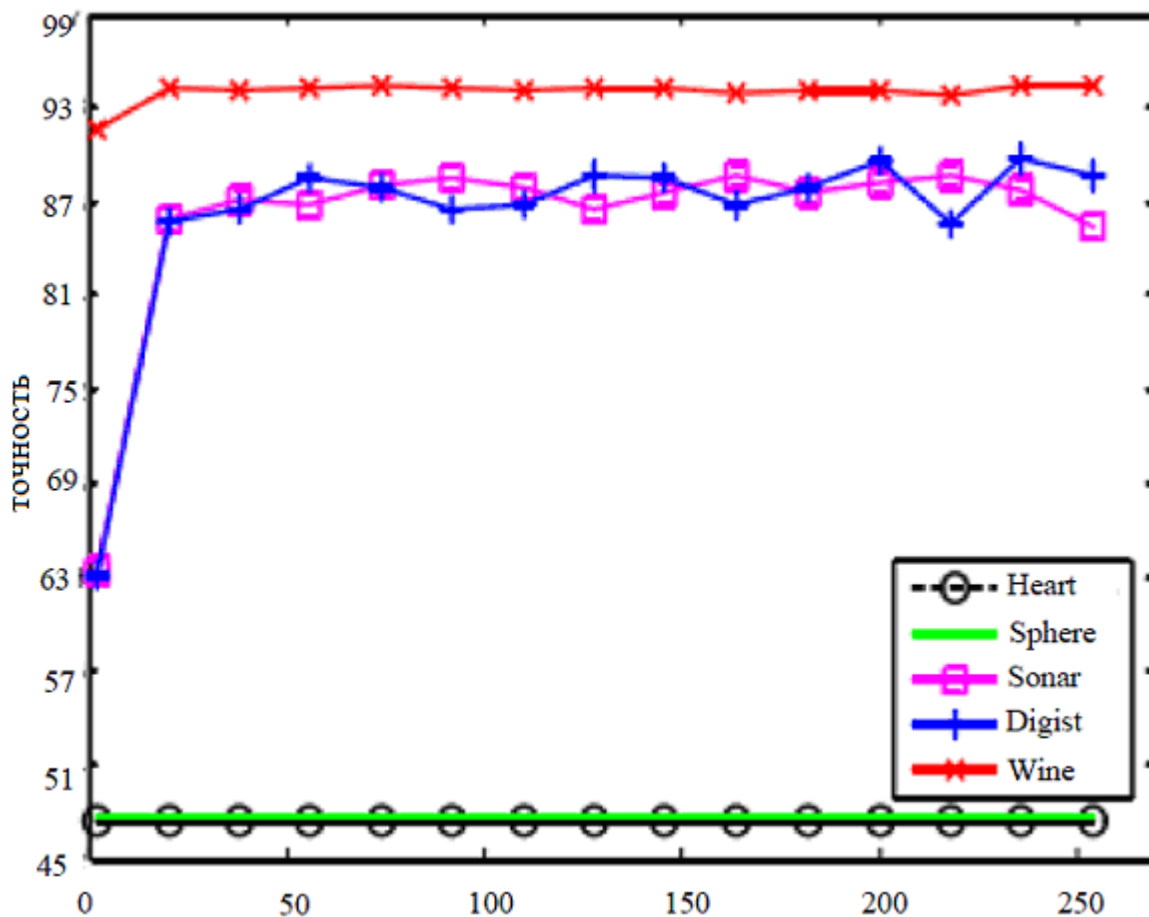
В наборе данных Wine скорость восстановления данных у обоих алгоритмов выбора объектов близка к оптимальной, что связано с потенциальной возможностью достижения наивысшей производительности кластеризации. Тем не менее, предложенный алгоритм демонстрирует определенные преимущества, достигая максимальной скорости восстановления данных на уровне 99,6%, что значительно повышает эффективность выбора признаков для больших наборов данных.

Сравнение точности выбора признаков для многомерных несбалансированных больших наборов данных с использованием двух алгоритмов показано на рис. 3.

На рис. 3а отражена точность выбора признаков из больших наборов данных, достигнутая традиционным методом; на рис. 3б представлены показатели точности выбора признаков для предлагаемого в данной работе алгоритма. Анализ результатов показывает, что независимо от количества пар ограниченных данных, используемых в эксперименте, точность предложенного метода выше по сравнению с традиционным подходом. При этом для больших наборов данных, характеризующихся глобальной дисперсией, различия в точности выбора признаков между двумя алгоритмами не являются статистически значимыми. Тем не менее, преимущества рассматриваемого алгоритма остаются очевидными.



а) Традиционный алгоритм



б) Предложенный алгоритм

Рис. 3. Сравнение точности выбора признака из большого набора данных

Кроме того, при использовании данных с попарными ограничениями для выбора объектов, наблюдается снижение точности предлагаемого алгоритма для данных Sphere, что может быть обусловлено линейными характеристиками данного набора данных. Невысокая точность анализа линейных данных приводит к снижению точности выбора объектов. В остальном, оба описанных алгоритма демонстрируют общие преимущества.

Таким образом, можно заключить, что точность и скорость воспроизведения предложенного алгоритма выше, чем у традиционного метода. Следовательно, разработанный алгоритм выбора признаков для многомерных неравновесных больших данных является эффективным и практически реализуемым.

Для дополнительной верификации эффективности описываемого алгоритма проведено сравнительное исследование времени, затрачиваемого на селекцию признаков многомерного несбалансированного большого набора данных для предложенного и традиционного алгоритмов. Результаты сравнительного анализа представлены на рис. 4.

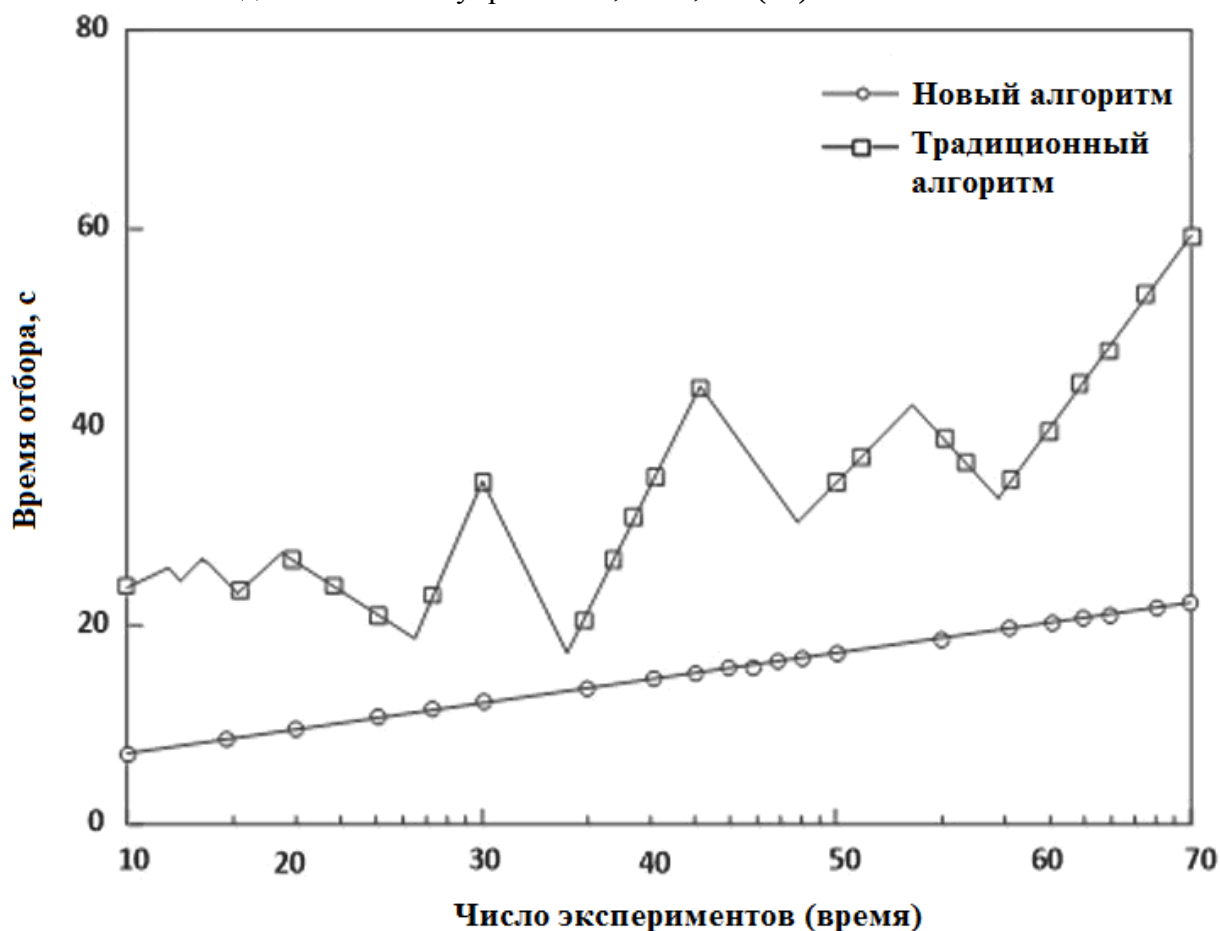


Рис. 4. Сравнение и анализ эффективного времени отбора

Согласно данным, представленным на рис. 4, с увеличением числа экспериментов эффективное время выбора признаков для многомерных несбалансированных наборов данных как для предложенного, так и для традиционных алгоритмов постепенно увеличивается. Однако временные затраты, характерные для алгоритма, разработанного в данной работе, стабильно находятся в пределах 20 секунд, в то время как эффективное время выбора признаков для традиционного алгоритма характеризуется нестабильностью и достигает 60 секунд. Полученные данные указывают на существенно меньшие временные затраты, обеспечиваемые предлагаемым алгоритмом при обработке многомерных несбалансированных больших наборов данных.

4. Заключение

В рамках данного исследования разработан алгоритм выбора признаков для больших наборов данных высокой размерности и несбалансированности, основанный на теории гранулярного слияния. Исходный большой массив данных преобразуется в подмножества данных малого масштаба с использованием функций регуляризации данных. На данной основе вычисляется функция выбора признаков для фрагментов данных. На заключительном этапе определяются весовые коэффициенты для каждого подмножества признаков, что обеспечивает реализацию классификации признаков для многомерных несбалансированных больших массивов данных. Эффективность и практическая применимость предложенного алгоритма подтверждена результатами экспериментальных исследований.

Список использованных источников

1. Zhe, J., Jianjun, H.: Prediction of lysine glutarylation sites by maximum relevance minimum redundancy feature selection. *Anal. Biochem.* 550, 1–7 (2018)
2. Xiaofeng, N.: Research on locally discrete text data mining in high-dimensional data sets. *Mod. Electron. Technol.* 40(19), 138–141 (2017)
3. Dan, Z., Chunming, W.: Feature selection based on improved quantum evolutionary algorithm. *Comput. Eng. Appl.* 54(1), 146–152 (2018)
4. Hongxiang, D., Qiuyu, Z., Moyi, Z.: FCBF feature selection algorithm based on normalized mutual information. *J. Huazhong Univ. Sci. Technol. (Nat. Sci. Ed.)* 45(1), 52–56 (2017)
5. Xin, Z., Haitao, W., Xuehong, C.: Feature selection algorithm based on random forest in Hadoop environment. *Comput. Technol. Dev.* 28(255(07)), 94–98+104 (2018)
6. Zhao, X., Zhang, L.: High-dimensional unbalanced data set classification algorithm based on SVM. *J. Nanjing Univ. (Nat. Sci.)* 54(2) (2018)
7. Liping, Y., Yunfei, L.: Zhu World Bank: anomaly detection algorithm based on high dimensional data stream. *Comput. Eng.* 44(1), 51–55 (2018)
8. Liu, S., Liu, D., Srivastava, G., et al.: Overview and methods of correlation filter algorithms in object tracking. *Complex Intell. Syst.* (2020). <https://doi.org/10.1007/s40747-020-00161-4>
9. Liu, S., Bai, W., Liu, G., et al.: Parallel fractal compression method for big video data. *Complexity* 2018, 2016976 (2018). <http://doi.org/10.1155/2018/2016976>
10. Hongjun, Z.: Research on visualization algorithm of high-dimensional data in multidimensional data sets. *Microelectron. Comput.* 34(5), 110–113 (2017)
11. Shuai, L., Weiling, B., Nianyin, Z., et al.: A fast fractal based compression for MRI images. *IEEE Access* 7, 62412–62420 (2019)
12. Gaber, M.M., Philip, S.Y.U.: Data stream mining in fog computing environment with feature selection using ensemble of swarm search algorithms. *New Gener. Comput.* 25(1), 95–115 (2018)
13. Li, J., Liu, H.: Challenges of feature selection for big data analytics. *IEEE Intell. Syst.* 32(2), 9–15 (2017)

Кравец О.Я.

**ПРОЕКТИРОВАНИЕ АЛГОРИТМА КЛАСТЕРИЗАЦИИ БОЛЬШИХ
НАБОРОВ ДАННЫХ О КОНФИДЕНЦИАЛЬНОСТИ ГЕТЕРОГЕННОЙ
СЕТИ, ФУНКЦИОНИРУЮЩЕГО НА БАЗЕ ОБЛАЧНЫХ
ВЫЧИСЛЕНИЙ**

Воронежский государственный технический университет

Введение

Благодаря стремительному развитию и повсеместному внедрению глобальных информационных технологий, а также экспоненциальному росту объемов генерируемых данных, мировое сообщество вступило в эпоху больших данных (Big Data). Этот переход характеризуется не только колоссальным увеличением объема информации, но и качественным изменением ее структуры, скорости поступления и способов обработки, что совокупно описывается концепцией "4V" (Volume, Velocity, Variety, Veracity). В этих условиях стратегия информационной безопасности любого технологически развитого государства должна учитывать беспрецедентную сложность и высокую динамику (своевременность) действий по обеспечению защищенности

крупномасштабных, гетерогенных и распределенных сетевых инфраструктур. Традиционные монолитные системы защиты, основанные на статических правилах и сигнатурах, демонстрируют свою несостоятельность при обработке потоков данных, объемы которых измеряются петабайтами, а скорость обновления требует мгновенной реакции.

Для решения проблемы недостаточной точности, высокой вычислительной стоимости и значительной доли случайности (недетерминированности) результатов, получаемых при использовании одиночных (единичных) алгоритмов кластеризации в условиях больших данных, был предложен инновационный алгоритм кластеризации, базирующийся на парадигме облачных вычислений (Cloud Computing). Данный алгоритм специально разработан для анализа больших наборов данных, содержащих сведения о конфиденциальности гетерогенных сетей (heterogeneous network privacy data). Архитектурно алгоритм использует распределенные вычислительные мощности облачной инфраструктуры для решения задач сбора, предварительной обработки и извлечения характерных признаков из массивных, слабоструктурированных наборов данных. На последующем этапе, для выполнения непосредственного интеллектуального анализа (Data Mining) и реализации процесса кластеризации, применяется метод вычисления подобия (similarity measure), который позволяет группировать объекты на основе количественной оценки их близости в многомерном пространстве признаков. Верификация предложенного алгоритма проводилась на эталонных наборах данных UCI (University of California, Irvine Machine Learning Repository), которые являются де-факто стандартом для бенчмаркинга в задачах машинного обучения. Эмпирические результаты показали, что эффективность (скорость сходимости и вычислительная масштабируемость) и точность (ассигасу, precision) кластеризации, достигнутые предлагаемым методом на основе облачных вычислений, значительно превосходят показатели существующих аналогов. Это убедительно свидетельствует о практической значимости разработанной стратегии и корректности выбранного направления оптимизации алгоритмической базы.

1. Состояние проблемы

Кластерный анализ (Cluster Analysis) как раздел интеллектуального анализа данных и машинного обучения изучается на протяжении многих десятилетий, за которые сформировалась обширная и систематизированная методологическая база [1]. В своей основе, кластеризация представляет собой задачу обучения без учителя (unsupervised learning), целью которой является разбиение множества физических или абстрактных объектов на группы (кластеры) на основе вычисления степени их сходства. Формально, задача сводится к тому, чтобы максимизировать внутрикластерное сходство (гомогенность) и минимизировать межкластерное сходство (гетерогенность) [2]. Однако ключевой проблемой применения единичного (одиночного) алгоритма кластеризации является нестабильность получаемых результатов и высокая степень случайности, обусловленная, в частности, чувствительностью к начальной инициализации параметров (например, выбору центроидов в k-means) и к локальным оптимумам целевой функции. Современные исследо-

вания в этой области, как правило, направлены на консолидацию результатов множества прогонов кластеризации или на ансамблевые методы (ensemble clustering), что позволяет нивелировать недостатки отдельных алгоритмов и повысить робастность итогового разбиения.

В последние годы особое внимание исследователей приковано к проблемам кластеризации частных (приватных) наборов данных в контексте гетерогенных сетей [3]. Это обусловлено необходимостью обеспечения информационной безопасности и конфиденциальности при анализе распределенных данных, циркулирующих в сетях с различной архитектурой и протоколами. Несмотря на растущий интерес, ряд фундаментальных вопросов остается нерешенным. В частности, до сих пор открыта проблема генерации оптимального подмножества данных для кластеризации, а также выбора наиболее эффективной стратегии объединения результатов. Особую сложность представляет собой разработка алгоритмов консенсусной кластеризации (consensus clustering) для больших наборов данных, которые содержат атрибуты категориального типа (classification attributes), поскольку стандартные метрики расстояния (например, евклидово) здесь неприменимы напрямую и требуют специальных методов кодирования и сопоставления. Следовательно, существует острая необходимость в проведении фундаментальных и прикладных исследований, направленных на разработку методов генерации участников кластера (cluster members) и их последующего интеллектуального анализа для достижения наилучших, наиболее устойчивых и интерпретируемых результатов кластеризации.

В данной статье предлагается исследование и разработка алгоритма кластеризации больших наборов данных, основанного на облачных вычислениях, специально предназначенного для анализа данных о конфиденциальности гетерогенных сетей. В работе представлены как теоретическое обоснование, так и практические методы и стратегии объединения (консенсуса) больших наборов данных. Методологический подход включает несколько последовательных этапов. На первом этапе, атрибуты каждого большого набора данных разделяются в соответствии с их значениями (дискретизация и сегментация), после чего осуществляется процесс сбора и извлечения характерных признаков, что позволяет получить исходные элементы кластера (базовые кластеры). На последующих этапах, путем итеративной корректировки параметров и применения методов интеллектуального анализа данных, достигаются оптимальные результаты агрегации (объединения) базовых кластеров в итоговую структуру. Для подтверждения теоретической значимости и практической применимости разработанного алгоритма была проведена экспериментальная апробация на реальных наборах данных. Результаты экспериментов демонстрируют, что предложенный алгоритм кластеризации на основе облачных вычислений позволяет существенно улучшить качество кластеризации данных, повысить точность и достоверность получаемых результатов, а также обеспечить требуемый уровень масштабируемости, что в совокупности подтверждает его высокую эффективность.

2. Разработка алгоритма кластеризации больших массивов данных

В качестве входного набора данных для разрабатываемого алгоритма кластеризации на основе облачных вычислений используется распределенная матрица большого набора данных (Distributed Matrix). Архитектура алгоритма предполагает параллельную обработку данных на вычислительных узлах кластера. В исходной матрице для каждого столбца (соответствующего определенному признаку или атрибуту) попарно вычисляются коэффициенты признаков (feature coefficients), которые представляют собой количественную меру связи или схожести между отдельными записями (точками данных) в пространстве признаков. На следующем этапе производится сравнительный анализ матричных характеристик каждого набора больших данных в пределах заданного порогового значения (threshold). Ключевым критерием классификации точек данных является сравнение количества точек в окрестности анализируемой точки (плотность) с заданным пороговым значением. Формально, точка данных считается основной (ядерной) точкой (core point), если ее коэффициент признака по отношению к любой другой точке превышает пороговое значение матричного объекта, а количество объектов в ее окрестности (точек с плотностью выше пороговой) больше или равно заданному параметру. Все точки, которые находятся в пределах порогового расстояния от основной точки и имеют сопоставимую плотность, относятся к одному и тому же кластеру (типу). Точки, не удовлетворяющие критериям плотности и связанности с основными точками, классифицируются как шумовые (noise points) и исключаются из дальнейшего анализа кластерной структуры.

Интеллектуальный анализ данных (Data Mining) в предложенной схеме выполняется с использованием всех общих характеристик, извлеченных из набора больших данных. Технология облачных вычислений применяется для распределенного сбора и извлечения характеристик общих данных, что позволяет эффективно анализировать возможности кластеризации наборов больших данных и выполнять расчеты кластеризации в режиме реального времени. Общая структурная схема алгоритма кластеризации на основе облачных вычислений, отражающая взаимосвязь этапов сбора данных, извлечения признаков, вычисления плотности и формирования кластеров, представлена на рис. 1.

2.1. Сбор и извлечение функций большого набора данных

Пусть имеется множество из n точек данных $X = \{x_1; x_2; x_3; \dots; x_n\}$, каждая из которых характеризуется набором из m атрибутов. Для каждого i -го атрибута определено множество из k_i различных дискретных значений, а сам атрибут обладает нормированным весовым коэффициентом w_i , отражающим его значимость в процессе кластеризации. В работе применяется наиболее простой и интерпретируемый метод генерации элементов кластера [4], основанный на процедуре разделения значений атрибутов. Функциональное выражение правила разделения атрибутов R_i для i -го набора данных имеет вид:

$$R_i = [C_{i,1}; C_{i,2}; C_{i,3}; \dots; C_{i,j}], 1 \leq i \leq m \quad (1)$$

где $C_{i,j}$ - j -я характеристика данных результата сегментации, $\sum_{i=1}^m w_i = 1$.

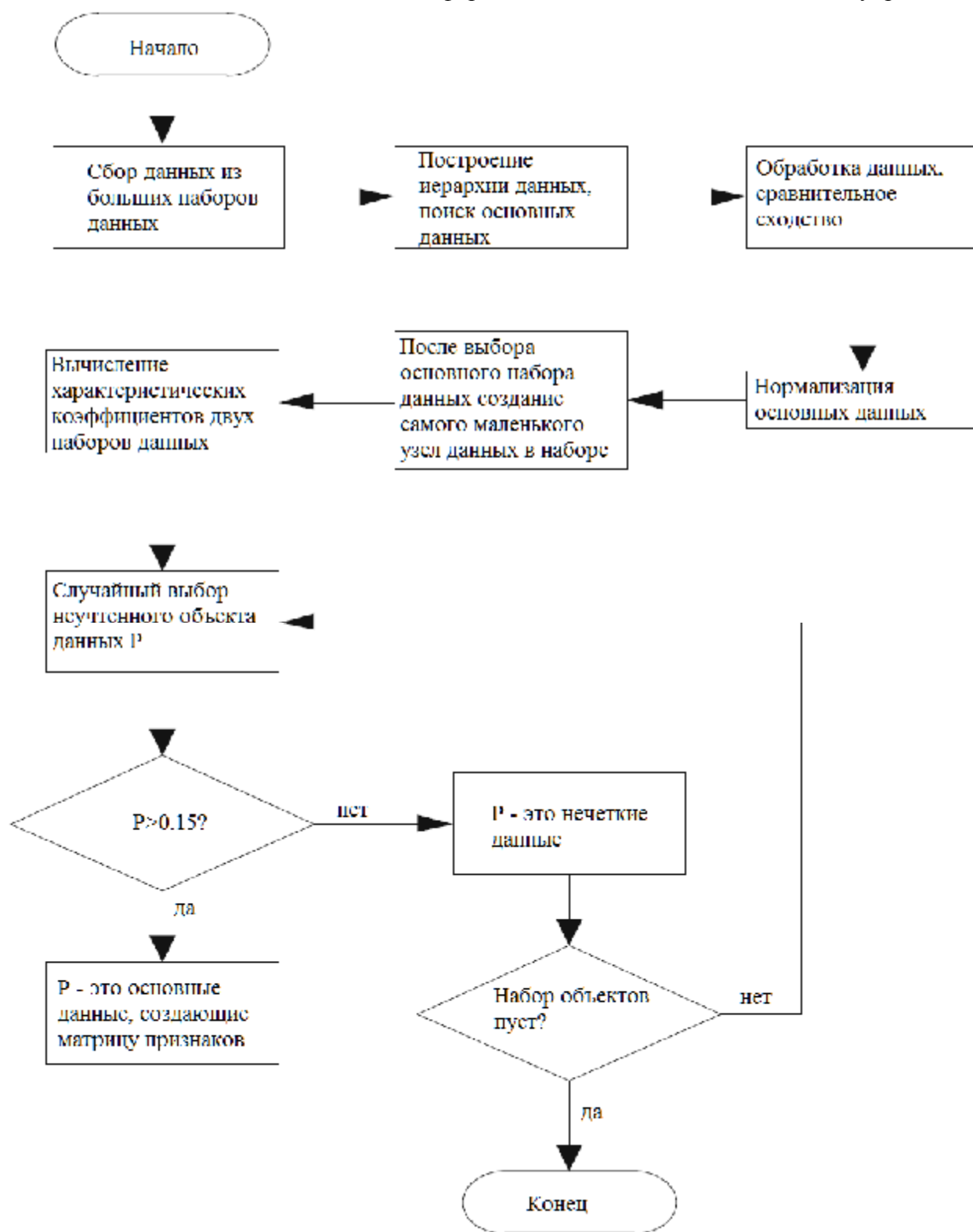


Рис. 1. Структурная схема алгоритма кластеризации на основе облачных вычислений

Основываясь на методологических подходах, изложенных в литературе [5], данные по каждому атрибуту разделяются на элементы кластера. Применение унифицированного метода для разделения различных подмножеств данных позволяет выявить характерные взаимосвязи между элементами кластера. В результате данной процедуры формируется полный набор результатов кластеризации $R = \{R_1; R_2; R_3; \dots; R_m\}$, состоящий из m элементов класте-

ра, где каждый элемент R_i характеризуется матрицей признаков размерности k_i . Такой подход обеспечивает унификацию представления разнородных данных и создает основу для их последующего сравнения.

Метод сбора признаков, основанный на определении сходства [6], представляет собой процесс агрегации метаданных в большом наборе данных, репрезентирующем конфиденциальность гетерогенных сетей. Посредством построения матрицы признаков определяется метод комбинированного разбиения на кластеры для нескольких больших наборов данных, при этом сходство между любыми двумя точками данных используется для количественного описания и определения признаков кластеризации. На начальном этапе технология облачных вычислений применяется для распределенного извлечения функций из больших массивов конфиденциальных данных в гетерогенных сетях с последующей их классификацией в соответствии с характеристиками [7]. На следующем этапе вводятся содержательные характеристики метаданных, при этом сами метаданные рассматриваются как векторное пространство, порождаемое набором ортогональных условий конфиденциальности в гетерогенной сети. Если t_i рассматривается как терм, $w_i(d)$ рассматривается как вес t_i в метаданных d , и каждое метадачное d может рассматриваться как нормализованный вектор признаков $V(d)=(t_{1i}, w_i(d)); (t_{2i}, w_i(d)); \dots, (t_{ni}, w_i(d))$. В общем, все данные, появляющиеся в d , рассматриваются как $t_i, w_i(d)$ определяется как функция частоты $tf_i(d)$, с которой t_i принадлежат d , т.е. $w_i(d)=\theta(tf_i(d))$. Функция частоты извлекается для получения характеристической функции большого набора данных:

$$J = \begin{cases} 1, & tf_i(R_i d) \geq 1 \\ 0, & tf_i(R_i d) < 1 \end{cases} \quad (2)$$

Извлечение функции частоты позволяет получить характеристическую функцию большого набора данных, которая отражает его ключевые свойства и закономерности:

Данная функция большого набора данных подвергается дальнейшей обработке с использованием аналогичных методов нормализации и масштабирования, что необходимо для подготовки к последующему процессу интеллектуального анализа данных и корректного вычисления мер сходства в многомерном пространстве признаков.

2.2. Интеллектуальный анализ больших массивов данных на основе облачных вычислений

Прежде всего, используя технологию облачных вычислений, случайным образом извлекаем элементы метаданных и преобразуем частный набор данных в структурированные данные, которые могут описывать содержимое метаданных. Затем используем кластерный анализ набора больших данных, чтобы сформировать структурированное дерево метаданных и определить новую концепцию набора больших данных в соответствии со структурой и получить соответствующую логическую взаимосвязь. Процесс интеллектуального анализа больших данных на основе облачных вычислений показан на рис. 2.

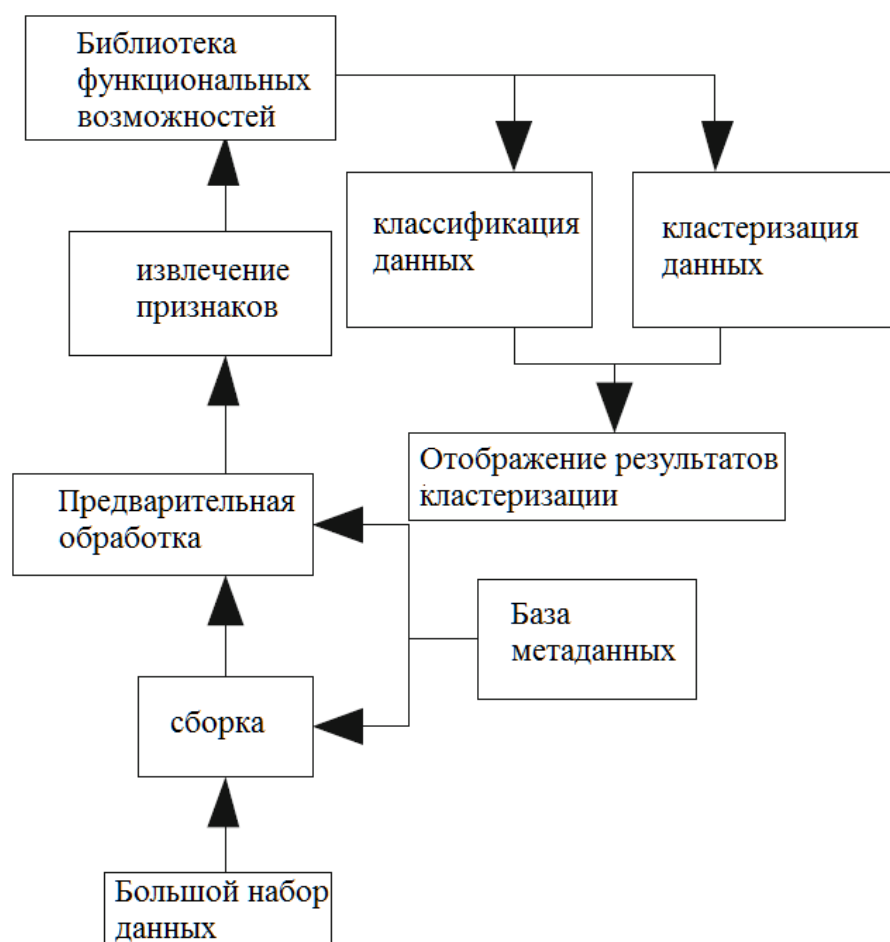


Рис. 2. Схема процесса интеллектуального анализа больших массивов данных

Поскольку объем данных в частном большом наборе данных в гетерогенной сети очень велик, размерность, используемая для представления вектора признаков метаданных, также очень велика и может достигать десятков тысяч измерений. Следовательно, необходимо извлечь сетевой терм с более высоким весом в качестве элемента признака из метаданных, чтобы достичь цели уменьшения размерности вектора признаков. Затем выполняется процесс интеллектуального анализа больших наборов данных с использованием кластеризации признаков. Процесс кластерного анализа больших наборов данных заключается в следующем:

1) Выбор нескольких наиболее репрезентативных объектов данных из исходных объектов.

2) В соответствии с принципом метода подобия [8] выбор набор данных о наиболее значимых объектах.

3) Преобразование исходных функций в меньшее количество новых функций посредством отображения или преобразования в технологии облачных вычислений [9, 10].

4) Используя метод оценочной функции [11], каждый объект в наборе объектов независимо оценивается и ему присваивается оценочный балл, и заранее определенное количество наилучших объектов выбирается в качестве подмножеств объектов из большого набора данных.

Пусть имеется выборочный набор $X=\{X_1; X_2; X_3; \dots; X_n\}$, подлежащий классификации, и n - это количество элементов в выборке, а c - количество целевых кластеров. Тогда существует следующая матрица интеллектуального анализа данных для n элементов, соответствующих классу c :

$$m_c = \begin{matrix} \begin{matrix} \hat{e} & \hat{e} & \dots & \hat{e} \\ \hat{e} & \dots & \dots & \hat{e} \\ \hat{e} & \dots & \dots & \hat{e} \end{matrix} \\ \begin{matrix} \mu_{c1}, \mu_{c2}, \dots, \mu_{cn} \end{matrix} \end{matrix} \quad n \in J, c \in J \quad (3)$$

где μ_{cn} ($1 \leq i \leq c$; $1 \leq j \leq n$) представляет функцию матрицы интеллектуального анализа данных n -го элемента для данных c -типа и соответствует

$\min J(X, m) = \sum_{i=1}^c \sum_{j=1}^n m_{cn} d_{cn}^2$, тогда задача кластеризации этого многомерного набора данных преобразуется в простую задачу нахождения минимального значения параметра целевой функции.

Процесс интеллектуального анализа данных для кластеризации больших массивов данных на основе облачных вычислений основан на исходном интеллектуальном анализе больших массивов данных, добавлении технологии облачных вычислений, добавлении ограничений к целевой функции для обеспечения выполнения кластерного поиска, удовлетворяющего условию, и упрощении процесса кластерного поиска информации мониторинга.

Вышеизложенное предполагает наличие большого набора данных без маркировки $X=\{X_1; X_2; X_3; \dots; X_n\}$, $X_n \in R_n$. Разделим его на K классов $C_1; C_2; C_3; \dots; C_k$, и среднее значение ни для одного класса не равно $M_1; M_2; M_3; \dots; M_k$. Предположим, что количество выборок в K равно N_k , тогда

$$m_k = \frac{1}{N_k} \sum_{i=0}^k X_i, k = 1 \dots K.$$

В соответствии с Евклидовым расстоянием и внутриклассовыми квадратами ошибок и критериев, целевая функция для кластеризации больших массивов данных на основе облака определяется как - это $J = \sum_{i=1}^{N_k} |X_i - m_k|^2$.

При инициализации алгоритма случайным образом выбирается центр каждого класса, поэтому выбор начального центра определяет качество результатов кластеризации. После внедрения технологии облачных вычислений большой набор данных, сформированный из небольшого числа помеченных выборок, содержит все K кластеров, и каждый класс содержит по крайней мере одну выборку для реализации процесса интеллектуального анализа больших массивов данных на основе облачных вычислений.

При инициализации алгоритма случайным образом выбирается центр каждого класса, поэтому выбор начального центра определяет качество результатов кластеризации. После внедрения технологии облачных вычислений большой набор данных, сформированный из небольшого числа помеченных выборок, содержит все K кластеров, и каждый класс содержит по крайней мере одну выборку для реализации процесса интеллектуального анализа больших массивов данных на основе облачных вычислений.

Важно подчеркнуть, что в контексте гетерогенных сетей, где данные отличаются высокой степенью разнородности как по своей природе, так и по происхождению, применение облачных вычислений в процессе интеллектуального анализа играет ключевую роль в обеспечении масштабируемости и отказоустойчивости вычислительных операций. Технология облачных вычислений позволяет распределять вычислительные ресурсы динамически, что особенно важно при работе с большими массивами данных, объем которых может изменяться непредсказуемо. В этом контексте процесс извлечения элементов метаданных из частного набора данных следует рассматривать не просто как механическую процедуру, а как целенаправленный процесс семантической интерпретации, в ходе которого неструктурированные или слабо структурированные данные преобразуются в форму, пригодную для последующей машинной обработки и анализа. Это преобразование подразумевает не только техническую нормализацию, но и концептуальное моделирование, которое позволяет связать отдельные элементы данных с их смысловым содержанием в рамках гетерогенной информационной инфраструктуры.

Следует также отметить, что уменьшение размерности вектора признаков является критическим этапом, поскольку высокая размерность данных может привести к так называемому "проклятию размерности", которое существенно снижает эффективность алгоритмов кластеризации и увеличивает вычислительную сложность. Выбор сетевых термов с более высоким весом в качестве элементов признаков основан на предположении, что именно эти термины наиболее полно и точно отражают семантическое содержание метаданных и, следовательно, позволяют сформировать репрезентативное подмножество признаков, которое сохраняет основные информационные характеристики исходного набора данных. Этот процесс аналогичен процедуре выбора признаков (feature selection) в классическом машинном обучении, однако в контексте облачных вычислений он приобретает дополнительные аспекты, связанные с необходимостью согласования распределенных операций и обеспечения целостности данных при выполнении параллельных вычислений. В этом отношении использование оценочной функции [11] для независимой оценки каждого объекта в наборе объектов является методологической основой для формализации процесса выбора оптимальных подмножеств объектов, которые будут использоваться в качестве репрезентативных элементов для последующей кластеризации.

2.3. Реализация алгоритма кластеризации больших массивов данных

Алгоритм кластеризации большого набора данных о конфиденциальности гетерогенной сети на основе облачных вычислений реализуется следующим образом:

При выполнении алгоритма кластеризации требуются два параметра ϵ и

μ , где $\mu = \begin{pmatrix} \epsilon_1, \epsilon_2, \dots, \epsilon_n \\ \epsilon_1, \epsilon_2, \dots, \epsilon_n \end{pmatrix}$, $n \in J, c \in J$; ϵ представляет собой пространственное измерение гетерогенной сети, вплоть до размерности [12-14].

Найдем количество основных точек данных, проверив параметр ε для поступающей точки данных в текущий момент времени. Если ε любой точки данных P содержит по крайней мере μ точек данных, создаем матрицу данных с точкой данных P в качестве основной точки. Затем, с помощью поиска по ширине, точки данных, которые могут быть непосредственно сгруппированы из этих основных точек данных, агрегируются, и вся полученная плотность из точки данных P присваивается одному классу.

Если P является основной точкой данных, точки данных кластера, начинающиеся с точки P , помечаются как текущий класс, и следующий шаг выполняется от центра матрицы. Если P не является точкой основных данных, то при кластеризации следующая точка данных будет обрабатываться по порядку до тех пор, пока не будет найдена полная точка основных данных кластера. Затем выбираем необработанную основную точку данных, чтобы начать расширение, и запускаем следующий процесс кластеризации последовательно, пока все точки данных не будут помечены [15, 16].

Для точек данных, которые не добавляются в матрицу кластеризации, они являются шумом и временно сохраняются в недопустимой области. Если количество данных в недопустимой области превышает максимальный диапазон заданного порогового значения, алгоритм группирует данные во временной области хранения и удаляет уже сгруппированные точки данных из области временного хранения. Динамические данные квадратичной кластеризации записываются как Q , а процесс вычисления кластеризации для Q выглядит следующим образом:

1) Большой набор данных, содержащий объекты со смешанными атрибутами, обрабатывается с использованием различных методов расчета расстояний, а новые точечные объекты данных вычисляются с помощью (2).

2) Выполняется оперативное обслуживание характеристик больших наборов данных и выполняйте интеллектуальную обработку после обслуживания.

3) Выполняется алгоритм кластеризации, и если есть данные, которые не кластеризованы, они помещаются во временное хранилище.

4) Матрица признаков данных снова обрабатывается, и алгоритм кластеризации выполняется до тех пор, пока не будут найдены основные точки данных.

Технология облачных вычислений используется для управления процессом реализации кластеризации больших массивов данных [17, 18], что решает проблему невысокого качества кластеризации по единственному алгоритму. Сначала вводится точка данных x^1 $X=\{d_1; d_2; \dots; d_n\}$ в памяти, d_1 представляет точку данных в памяти. Реализация большого набора данных заменяется тройкой, которая эквивалентна центральной точке с весом для участия в кластеризации, количество точек данных является весом, а результатом кластеризации является набор меток, $labels=U^k$. Выводим K наборов непересекающихся матриц больших данных $\{X_i\}_{i=1}^k$ из x , а целевая функция определена как $J \hat{A} J = ik$.

Таким образом, получается локальный оптимальный процесс кластеризации большого набора данных.

В представленной реализации алгоритма необходимо акцентировать внимание на том, что параметры ε и μ являются критическими для определения плотности распределения данных в пространстве гетерогенной сети. Параметр ε , представляющий пространственное измерение, фактически задает радиус окрестности, в пределах которого оценивается плотность точек данных, в то время как параметр μ определяет минимальное количество точек, необходимое для того, чтобы рассматривать данную область как кластер. Такая бинарная параметризация основана на классической парадигме алгоритмов плотностной кластеризации, таких как DBSCAN, однако в контексте облачных вычислений она приобретает дополнительные аспекты, связанные с распределенным характером обработки данных. Важно отметить, что выбор оптимального значения ε существенно влияет на качество кластеризации: слишком малое значение может привести к фрагментации кластеров и большому количеству точек шума, тогда как слишком большое значение может привести к слиянию различных кластеров и потере тонкой структуры данных. Для решения этой проблемы в гетерогенных сетях часто используется адаптивный подход, при котором значение ε определяется на основе статистических характеристик распределения данных, таких как расстояние до k -го ближайшего соседа.

Механизм идентификации основных точек данных, реализуемый путем проверки параметра ε для каждой поступающей точки, основан на принципе достижимости плотности. В этом контексте точка данных P считается основной, если в ее ε -окрестности содержится не менее μ точек, включая саму точку P . Такая точка является ядром потенциального кластера, и процесс расширения кластера осуществляется посредством поиска по ширине, что обеспечивает систематическое и полное покрытие всех точек, которые являются плотно-достижимыми из данной основной точки. Применение поиска по ширине имеет важное преимущество перед поиском в глубину: он гарантирует нахождение кратчайших путей в графе связности точек данных, что позволяет минимизировать количество итераций и повысить эффективность процесса кластеризации, что особенно важно при обработке больших массивов данных в распределенной среде облачных вычислений. После того как все точки, достижимые из основной точки P , будут агрегированы в один кластер, алгоритм переходит к следующей необработанной основной точке, что обеспечивает полное покрытие всего набора данных. Этот итеративный процесс продолжается до тех пор, пока все точки данных не будут помечены либо как принадлежащие к определенному кластеру, либо как шумовые.

Важно отметить, что обработка точек шума, которые не попадают ни в один из кластеров, представляет собой отдельную задачу, решение которой имеет важное значение для обеспечения полноты и корректности результатов кластеризации. В описанной реализации точки шума временно сохраняются в недопустимой области, что позволяет изолировать их от процесса кластеризации и не допустить их негативного влияния на формирование кластеров.

Однако, если количество таких точек превышает максимальный диапазон заданного порогового значения, алгоритм инициирует процесс повторной кластеризации данных из временной области хранения. Этот механизм является важным элементом адаптивности алгоритма, поскольку он позволяет учитывать возможность того, что некоторые точки данных, первоначально классифицированные как шум, на самом деле могут образовывать отдельные кластеры малого размера, которые не были обнаружены на начальных итерациях из-за недостаточной плотности. Таким образом, алгоритм демонстрирует способность к самокоррекции и уточнению результатов кластеризации, что повышает его устойчивость к вариативности данных в гетерогенных сетях.

Процесс квадратичной кластеризации динамических данных Q , описанный в четырех шагах, представляет собой двухэтапный подход, который позволяет повысить качество кластеризации за счет последовательной обработки данных с использованием различных методов расчета расстояний. На первом этапе большой набор данных, содержащий объекты со смешанными атрибутами, обрабатывается с использованием различных методов расчета расстояний, что позволяет учесть специфику различных типов атрибутов. Например, для числовых атрибутов может использоваться евклидово расстояние, для категориальных атрибутов - расстояние Хэмминга или коэффициент Жаккара, а для порядковых атрибутов - ранговые метрики. Такой многоаспектный подход к вычислению расстояний является критически важным для корректной кластеризации данных в гетерогенных сетях, где объекты могут характеризоваться атрибутами различной природы. На втором этапе выполняется оперативное обслуживание характеристик больших наборов данных, что включает в себя обновление матрицы признаков и пересчет расстояний в соответствии с изменениями в данных. Интеллектуальная обработка после обслуживания позволяет выявить динамические изменения в структуре данных и соответствующим образом скорректировать процесс кластеризации.

Внедрение технологии облачных вычислений в процесс кластеризации больших массивов данных обеспечивает ряд существенных преимуществ. Во-первых, облачная инфраструктура позволяет распределять вычислительные нагрузки между множеством серверов, что значительно сокращает время выполнения кластеризации для больших наборов данных. Во-вторых, облачные вычисления обеспечивают высокую доступность и отказоустойчивость, что гарантирует продолжение процесса кластеризации даже при сбоях отдельных компонентов инфраструктуры. В-третьих, масштабируемость облачных платформ позволяет динамически добавлять вычислительные ресурсы по мере увеличения объема данных, что особенно важно в контексте больших данных, объем которых может быстро увеличиваться. Представление большого набора данных в виде тройки, эквивалентной центральной точке с весом, является эффективным способом агрегации информации для кластеризации: количество точек данных служит весом, что позволяет алгоритму учитывать плотность распределения данных при формировании кластеров. Выходной результат кластеризации - набор меток $labels=U_k$ и K непересекающихся матриц больших данных $\{X_i\}$, $k=1$ - позволяет получить полную

картину структуры данных, что необходимо для последующего анализа и интерпретации результатов.

На данный момент завершена разработка алгоритма кластеризации большого набора данных, основанного на облачных данных, для обеспечения конфиденциальности гетерогенной сети.

4. Имитационный эксперимент

Для подтверждения теоретической обоснованности и практической применимости разработанного алгоритма кластеризации, функционирующего на базе облачных вычислений и ориентированного на обработку больших массивов конфиденциальных данных в гетерогенных сетях, была реализована серия имитационных экспериментов. Целью данных испытаний являлась верификация заявленных характеристик алгоритма, оценка его устойчивости к различным типам входных данных, а также сравнительный анализ эффективности с классическими подходами к кластеризации.

В качестве экспериментального объекта выбран эталонный набор данных UCI, репрезентирующий структуру конфиденциальной информации в гетерогенной сети. Данный выбор обусловлен наличием в наборе разнородных признаков, varying по своей природе и размерности, что позволяет комплексно оценить адаптивные свойства алгоритма. Над подготовленным массивом данных была произведена процедура расчета кластеризации с применением предложенного облачного алгоритма.

Для обеспечения объективности и воспроизводимости результатов эксперимента была разработана методика сравнительного анализа. В соответствии с ней, параллельно с разработанным алгоритмом, проводилась кластеризация того же набора данных традиционным методом, выступающим в роли базового эталона. Помимо визуального сопоставления полученных кластерных структур, был выполнен статистический анализ ключевых метрик производительности обоих алгоритмов. К числу таких метрик относятся частота ошибок кластеризации, отражающая долю неверно классифицированных объектов, и скорость поиска кластеров, характеризующая временную эффективность обработки данных. Итоговые количественные показатели, полученные в ходе серии запусков, систематизированы и представлены в табл. 1 и 2, а графическая интерпретация точности работы алгоритмов отображена на рис. 3.

Таблица 1

Результаты традиционного алгоритма кластеризации

Категория	Размерность сети	Поток	Точность	Матрица данных
Частота ошибок (%)	0.59	0.85	0.36	0.48
Скорость поиска кластеров (v/мс)	23.6	15.4	41.2	25.9

Таблица 2

Результаты алгоритма кластеризации на основе облачных вычислений

Категория	Размерность сети	Поток	Точность	Матрица данных
Частота ошибок (%)	0.12	0.05	0.11	0.03
Скорость поиска кластеров (v/мс)	68.3	72.5	82.6	75.6

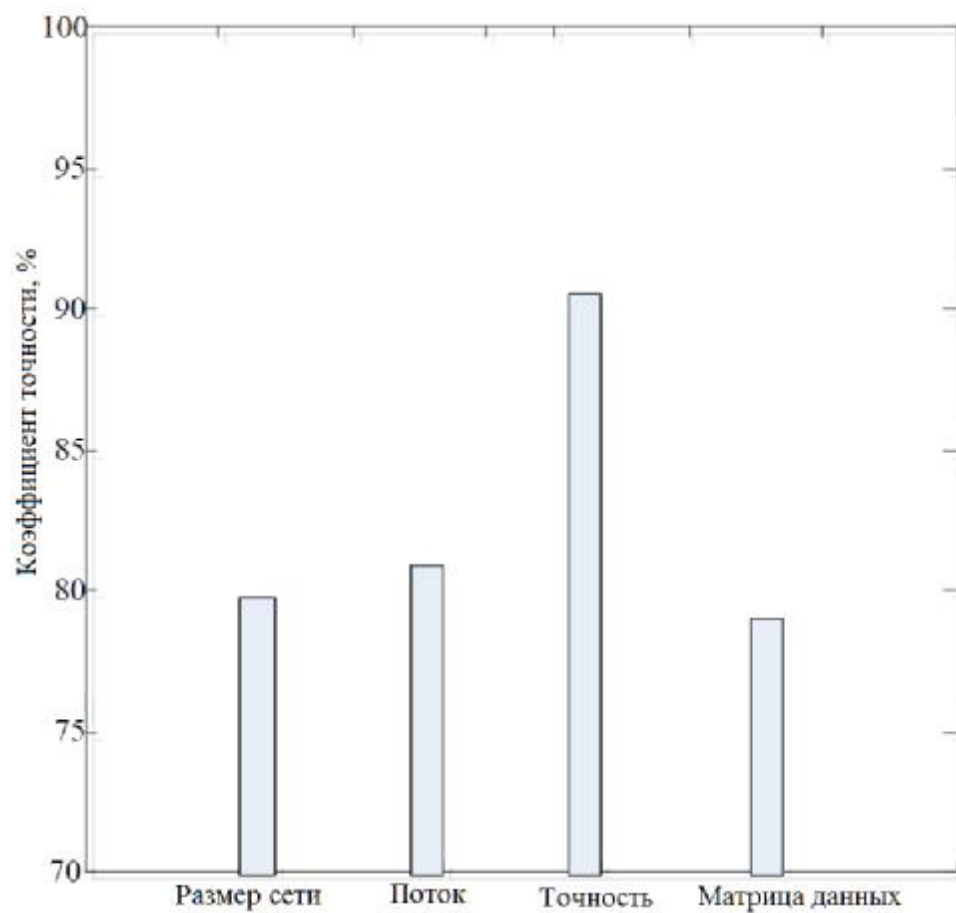
Детальный анализ данных, приведенных в табл. 1 и 2, позволяет сделать ряд существенных выводов. Прежде всего, обращает на себя внимание значительное превосходство разработанного алгоритма по критерию точности. Частота ошибок, допускаемых традиционным алгоритмом при обработке различных структур данных (размерность сети, поток, точность, матрица данных), варьируется в диапазоне от 0.36% до 0.85%. В то же время, для облачного алгоритма этот показатель не превышает 0.12%, демонстрируя снижение ошибки в несколько раз, вплоть до значения 0.03% для матричных данных. Это свидетельствует о высокой надежности и точности предложенного метода при работе с гетерогенными данными.

Не менее показательным является сравнение скоростных характеристик. Скорость поиска кластеров традиционного алгоритма составляет от 15.4 до 41.2 v/мс в зависимости от категории данных. Разработанный алгоритм демонстрирует стабильно высокую скорость (от 68.3 до 82.6 v/мс), что в среднем в 2.5–3 раза превышает показатели традиционной методики. Такая значительная разница в производительности объясняется, прежде всего, использованием распределенных вычислительных мощностей облачной инфраструктуры, позволяющих параллельно обрабатывать подмножества данных и минимизировать временные затраты на итерационные процессы поиска оптимальных кластерных центров.

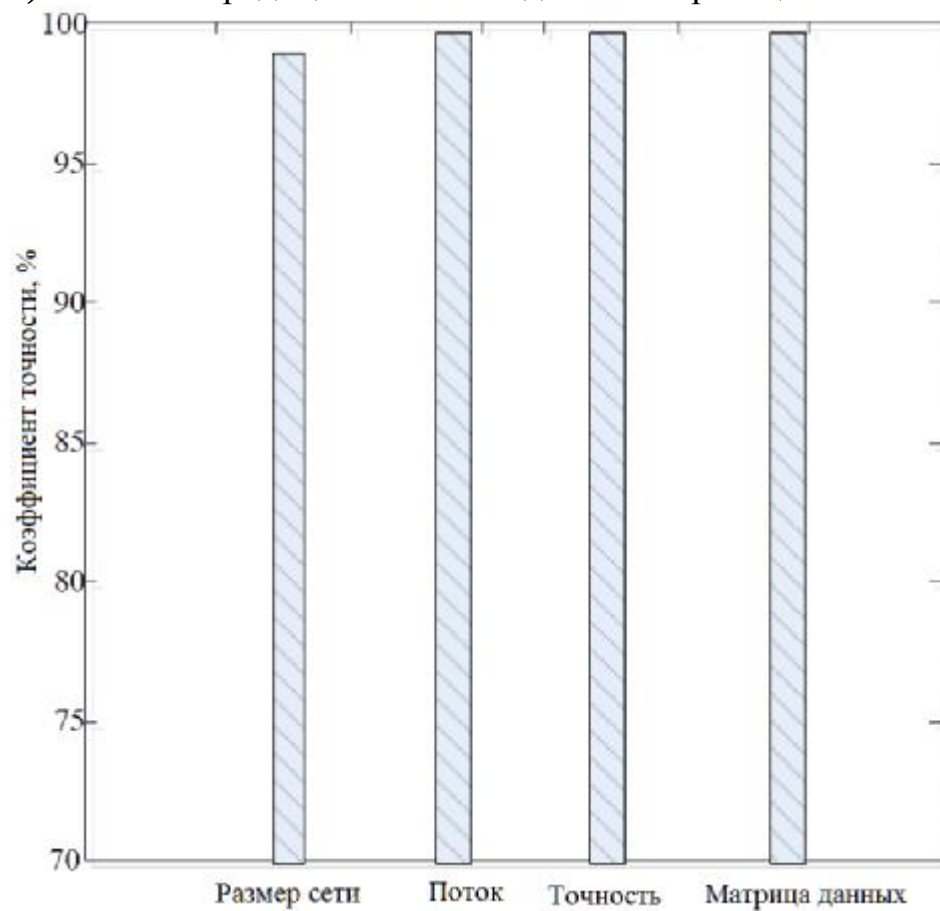
Таким образом, совокупность полученных данных убедительно доказывает, что алгоритм кластеризации больших данных о конфиденциальности гетерогенной сети, основанный на облачных вычислениях, обеспечивает существенно лучший эффект как по точности, так и по скорости обработки по сравнению с традиционным алгоритмом.

Графическая интерпретация результатов, представленная на рис. 3, визуализирует процентное соотношение корректно классифицированных объектов для обоих алгоритмов. Согласно приведенным данным, точность кластеризации, достигнутая разработанным алгоритмом, приближается к 99%. В то время как точность традиционного алгоритма кластеризации зафиксирована на уровне 91%. Разница в 8 процентных пунктов является статистически значимой и подтверждает вывод о том, что точность кластеризации разработанного алгоритма не просто выше, а качественно превосходит показатели традиционного подхода.

Алгоритм кластеризации на основе облачных вычислений в процессе обработки большого набора данных о конфиденциальности гетерогенной сети демонстрирует превосходство по всем ключевым параметрам. Независимо от того, идет ли речь о скорости кластеризации, точности обработки каждого типа структуры конфиденциальных данных, устойчивости к вариациям трафика или точности измерений, его показатели значительно лучше, чем результаты, полученные при использовании традиционного алгоритма. Наблюдается не только повышение точности кластеризации наборов частных данных в гетерогенных сетях, но и существенное возрастание стабильности всего вычислительного процесса.



а) Точность традиционных методов кластеризации



б) Точность предложенного метода кластеризации

Рис. 3. Анализ точности двух алгоритмов кластеризации

Заключение

В представленной работе проведен всесторонний анализ и выполнено проектирование алгоритма кластеризации больших наборов данных о конфиденциальности гетерогенной сети, функционирующего на базе облачных вычислений. Ключевым аспектом исследования стало использование преимуществ технологии облачных вычислений для организации эффективного процесса сбора и извлечения матричных характеристик из больших массивов разнородных данных. Разработанный подход органично сочетает в себе метод подобия для оценки близости объектов, методы интеллектуального анализа больших массивов данных, позволяющие выявлять скрытые закономерности, и непосредственно реализацию алгоритма кластеризации, адаптированную для распределенной облачной среды.

Результаты проведенных имитационных экспериментов наглядно демонстрируют высокую эффективность разработанного алгоритма. При выполнении расчета кластеризации большого набора данных о конфиденциальности в гетерогенной сети предложенный метод обеспечивает значительное повышение точности кластеризации по сравнению с традиционными подходами. Кроме того, применение облачной парадигмы позволяет эффективно уменьшить ошибку кластеризации, свести к минимуму временные затраты на выполнение вычислений и, как следствие, существенно повысить общую эффективность работы алгоритма. Разработанное решение представляет собой надежный инструмент для решения задач анализа конфиденциальных данных в сложных гетерогенных сетевых структурах, гарантируя высокую точность и производительность.

Список использованных источников

1. Xinchun, Y.C., et al.: Application of multi-relational data clustering algorithm in internetpublic opinion pre-warning on emergent. *Int. Engl. Educ. Res.* 1, 16–19 (2019)
 2. Dongmei, C.: Discussions on big data security and privacy protection based on cloud-computing. *Comput. Knowl. Technol.* 15(15), 101–103 (2018)
 3. Ma, R., Angryk, R.A., Riley, P., et al.: Coronal mass ejection data clustering and visualization of decision trees. *Astrophys. J. Suppl. Ser.* 236(1), 14–17 (2018)
 4. Mingbo, Pan: Research on privacy protection algorithms for network data in large dataenvironment. *Microelectron. Comput.* 34(7), 101–104 (2017)
 5. Wenzheng, Z., Zaiyun, W., Afang, L.: Relevant analysis of big data security and privacyprotection based on cloud computing. *Netw. Secur. Technol. Appl.* 15(4), 59–63 (2018)
 6. Hodi: Analysis of big data security and privacy protection in cloud computing. *Electron.World* 25(16), 98–101 (2017)
 7. Yang, H.: Privacy protection of large data security based on cloud computing. *Netw. Secur. Technol. Appl.* 25(11), 86–87 (2017)
 8. Arman, O., Rahmat, K., Mehrdad, T.H., et al.: Direct probabilistic load flow in radial distribution systems including wind farms: an approach based on data clustering. *Energies* 11(2), 310–311 (2018)
 9. Jinbo, X., Jun, R., Lei, C., et al.: Enhancing privacy and availability for data clustering inintelligent electrical service of IoT. *IEEE Internet Things J.* 6(2), 1530–1540 (2018)
 10. Yating, W.: Exploration of big data security privacy and protection based on cloud-computing. *J. Heihe Univ.* 15(6), 85–87 (2018)
 11. Liu, S., Li, Z., Zhang, Y., et al.: Introduction of key problems in long-distance learning
-

andtraining. Mob. Netw. Appl. 24(1), 1–4 (2019)

12. Shudong, H., Yazhou, R., Zenglin, X.: Robust multi-view data clustering with multi-view capped-norm K-means. Neurocomputing 311, 197–208 (2018)

13. Sun, G., Liu, S. (eds.): ADHIP 2017. LNICST, vol. 219. Springer, Cham (2018). <https://doi.org/10.1007/978-3-319-73317-3>.

14. Allab, K., Labiod, L., Nadif, M.: A semi-NMF-PCA unified framework for data clustering. IEEE Trans. Knowl. Data Eng. 29(1), 2–16 (2017)

15. Li, T., Pintado, F.D.L.P., Corchado, J.M., et al.: Multi-source homogeneous data clustering for multi-target detection from cluttered background with misdetection. Appl. Soft Comput. 60, 436–446 (2017)

16. Fu, W., Liu, S., Srivastava, G.: Optimization of big data scheduling in social networks. Entropy 21(9), 902 (2019)

17. Chang, X., Wang, Q., Liu, Y., et al.: Sparse regularization in fuzzy -means for high-dimensional data clustering. Cybern. IEEE Trans. 47(9), 2616–2627 (2017)

18. Liu, S., Lu, M., Li, H., et al.: Prediction of gene expression patterns with generalized linear regression model. Front. Genet. 10, 120 (2019)

Недикова Т.Н., Сергеев М.Ю.

ПРИМЕНЕНИЕ АППАРАТА НЕЧЕТКОЙ ЛОГИКИ В ЗАДАЧЕ ОЦЕНКИ КРЕДИТНОГО РИСКА

Воронежский государственный технический университет, г. Воронеж

Введение

Моделирование реальных процессов и явлений часто базируется на использовании информации, которая является неполной, неточной или неопределенной.

Качественные оценки, как правило, не укладываются в жесткие рамки двузначной логики.

Традиционные математические подходы, основанные на классической теории вероятностей или двузначной логике, не всегда позволяют адекватно описать ситуации, в которых неопределенность связана не со случайностью событий, а с размытостью самих понятий и нечеткостью границ между классами объектов.

Применение нечеткой логики позволяет описывать лингвистическую неопределенность, связанную с размытостью границ между понятиями.

Объектом данного исследования является разработка методики применения нечетких булевых переменных и логических операций над ними для исследования аналитических функций нечетких переменных с целью определения диапазона значений этих функций. Выполнение программной реализации для 3D-визуализации аналитических функций нечеткой логики на примере задачи оценки кредитного риска обеспечивает создание инструмента исследования поведения функций в области нечетких систем. Программное средство позволит быстро получать числовые и графические характеристики исследуемых функций.

Общие подходы к моделированию неопределенности

Нечеткая логика описывает лингвистическую неопределенность - неопределенность, связанную с размытостью границ между понятиями. В теории

нечеткой логики определяют, что высказывание с некоторой промежуточной степенью истинности принимает любое значение на отрезке $[0,1]$.

Предметом исследования выступают аналитические функции нечетких булевых переменных. Эти функции могут быть представлены формулой, содержащей только операции конъюнкции, дизъюнкции и отрицания, выполняемые над аргументами функции и их отрицаниями.

Требуется рассмотреть возможности упрощения аналитических функций, представленных в виде формул.

Необходимо изучить интервальный подход к оценке истинности нечетких функций, позволяющий определять диапазон значений функции, если известны интервалы принадлежности входных переменных.

Требуется рассмотреть лингвистические переменные «истина» и «ложь», их функции принадлежности и способы интерпретации числовых значений истинности в терминах естественного языка.

На последнем этапе необходимо разработать программную реализацию для 3D-визуализации аналитических функций нечеткой логики.

При исследовании могут быть использованы следующие методы:

- методы математической логики (для анализа булевых функций);
- методы теории множеств (для формализации нечетких понятий);
- методы нечеткой логики (для определения операций и анализа их свойств);
- методы компьютерной графики (для визуализации трехмерных поверхностей);
- методы веб-программирования (HTML, CSS, JavaScript, библиотека Three.js) для организации интерфейса взаимодействия с пользователями.

Постановка задачи оценки кредитного риска заемщика

В банковской сфере при принятии решения о выдаче кредита кредитный эксперт оценивает множество факторов, которые часто формулируются в нечетких, качественных терминах. К таким факторам относятся:

- кредитная история; оценивается как «отличная», «хорошая», «удовлетворительная», «плохая»;
- уровень дохода; «высокий», «средний», «низкий»;
- наличие и качество обеспечения (залога); «надежное», «достаточное», «недостаточное», «отсутствует».

Классическая бинарная логика не позволяет адекватно обработать такие качественные оценки, поскольку они не сводятся к простой дихотомии «истина/ложь». Нечеткая логика предоставляет математический аппарат для формализации и анализа подобных высказываний [1-2].

Для описания задачи оценки кредитоспособности заемщика вводят следующие лингвистические переменные и их числовые интерпретации (табл. 1). Каждая переменная принимает значения на отрезке $[0,1]$, где 0 соответствует наихудшему значению, а 1 - наилучшему.

Кредитный комитет принимает решение на основе следующего правила (экспертная логика).

Таблица 1

Лингвистические переменные

Переменная	Обозначение	Лингвистическое значение	Значения на отрезке
Кредитная история	k	Отличная	[0.8,1.0]
		Хорошая	[0.6,0.8]
		Удовлетворительная	[0.4,0.6]
		Плохая	[0.0,0.4]
Уровень дохода	d	Высокий	[0.7,1.0]
		Средний	[0.4,0.7]
		Низкий	[0.0,0.4]
Обеспечение кредита (залог)	c	Надежное	[0.8,1.0]
		Достаточное	[0.5,0.8]
		Недостаточное	[0.2,0.5]
		Отсутствует	[0.0,0.2]

Правило. Кредит одобряется, если кредитная история является хорошей или отличной, доход не является низким, а обеспечение не является недостаточным или отсутствующим.

На языке нечеткой логики это правило записывается в виде аналитической функции [1-2]:

$$R=(k \geq 0.6) \wedge (d \geq 0.4) \wedge (c \geq 0.5)$$

где $k \geq 0.6$ - кредитная история хорошая или отличная;

$d \geq 0.4$ - доход не низкий (средний или высокий);

$c \geq 0.5$ - обеспечение достаточное или надежное.

Однако на практике эти условия не являются жесткими порогами. Более реалистичная модель использует нечеткие операции:

$$R=(k \wedge d \wedge c) \vee (\text{особые случаи})$$

Для упрощения рассматривают базовую формулу:

$$R=\min (k, d, c)$$

Таким образом, итоговая оценка кредитоспособности определяется худшим из трех показателей. Это соответствует консервативной стратегии: слабое место заемщика определяет общий риск.

Операция минимума соответствует консервативной стратегии оценки: итоговая кредитоспособность определяется «самым слабым звеном» - худшим из трех показателей.

Если хотя бы один из параметров находится на низком уровне, общая оценка риска будет низкой. Данный подход часто используется на практике. Он отражает принцип: «кредит настолько надежен, насколько надежно его самое слабое место».

Возможные варианты интерпретации функции R приведены в табл. 2.

Можно привести примеры оценки возможных заемщиков. Характеристики заемщиков оценены экспертами в области выдачи кредитов.

Пример 1. Первый заемщик обладает следующими характеристиками:

- кредитная история хорошая и оценена $k=0.7$;
- уровень дохода средний и имеет значение $d=0.55$;
- обеспечение достаточное $c=0.65$.

Осуществляют вычисление функции

$$R = \min(0.7, 0.55, 0.65) = 0.55$$

Значение R попадает в интервал $[0.4, 0.6]$, что соответствует значению «условно надежный». Рекомендовано одобрить кредит с повышенной процентной ставкой или уменьшенной суммой.

Таблица 2

Итоговые оценки кредитоспособности

Значение R	Лингвистическая оценка	Рекомендуемое решение
$[0.8, 1.0]$	Очень надежный заемщик	Одобрение на максимальную сумму с льготной ставкой
$[0.6, 0.8]$	Надежный заемщик	Одобрение на стандартных условиях
$[0.4, 0.6]$	Условно надежный	Одобрение с повышенной ставкой или уменьшенной суммой
$[0.2, 0.4]$	С высоким риском	Требуется дополнительный залог или поручитель
$[0.0, 0.2]$	Ненадежный	Отказ в выдаче кредита

Пример 2. Второй заемщик обладает следующими характеристиками:

- кредитная история отличная и оценена $k=0.9$;
- уровень дохода высокий и имеет значение $d=0.95$;
- обеспечение надежное $c=0.9$.

Осуществляют вычисление функции

$$R = \min(0.9, 0.95, 0.9) = 0.9$$

Значение R попадает в интервал $[0.8, 1.0]$, что соответствует значению «очень надежный заемщик». Рекомендовано одобрить кредит на максимальную сумму с льготной ставкой.

Таким образом, оценка кредитного риска базируется на следующих действиях:

- использование нечетких булевых переменных для формализации задачи оценки кредитного риска;
- формирование лингвистических переменных, их лингвистических значений и числовых интерпретаций с привлечением экспертов из предметной области; при этом качественные экспертные оценки («хорошая кредитная история», «средний доход» и т.д.) получили количественное выражение в виде числовых интервалов;
- формирование правила принятия решения; например: использование операции минимума $R = \min(k, d, c)$ для реализации консервативной стратегии кредитования: итоговая оценка определяется худшим из трех показателей;
- оценка итогового значения функции R путем формирования лингвистических оценок, соответствующих им интервалов значений и рекомендуемых решений.

Программная реализация визуализатора аналитических функций нечеткой логики

Программная реализация осуществлена с использованием языка программирования Python [3, 4]. Данный язык оптимально подходит для задач математического моделирования и обработки данных.

Программа построена по модульному принципу и состоит из следующих

основных модулей.

Парсер нечетких формул - модуль, отвечающий за анализ и компиляцию введенной пользователем формулы во внутреннее представление, пригодное для вычислений.

Вычислительный модуль - модуль, который на основе скомпилированной формулы вычисляет значения функции для всех точек сетки $p, q \in [0,1]$.

Модуль визуализации - модуль, построенный на базе библиотеки Three.js, который создает трехмерную сцену, поверхность, освещение и управление камерой.

Модуль статистики - модуль, вычисляющий статистические характеристики функции (минимум, максимум, среднее, дисперсию).

Модуль экспорта - модуль, обеспечивающий сохранение скриншотов и экспорт данных в CSV-формат.

Пользовательский интерфейс - совокупность HTML-элементов и CSS-стилей, обеспечивающих взаимодействие пользователя с программой.

Архитектура приложения представлена на рис.1.

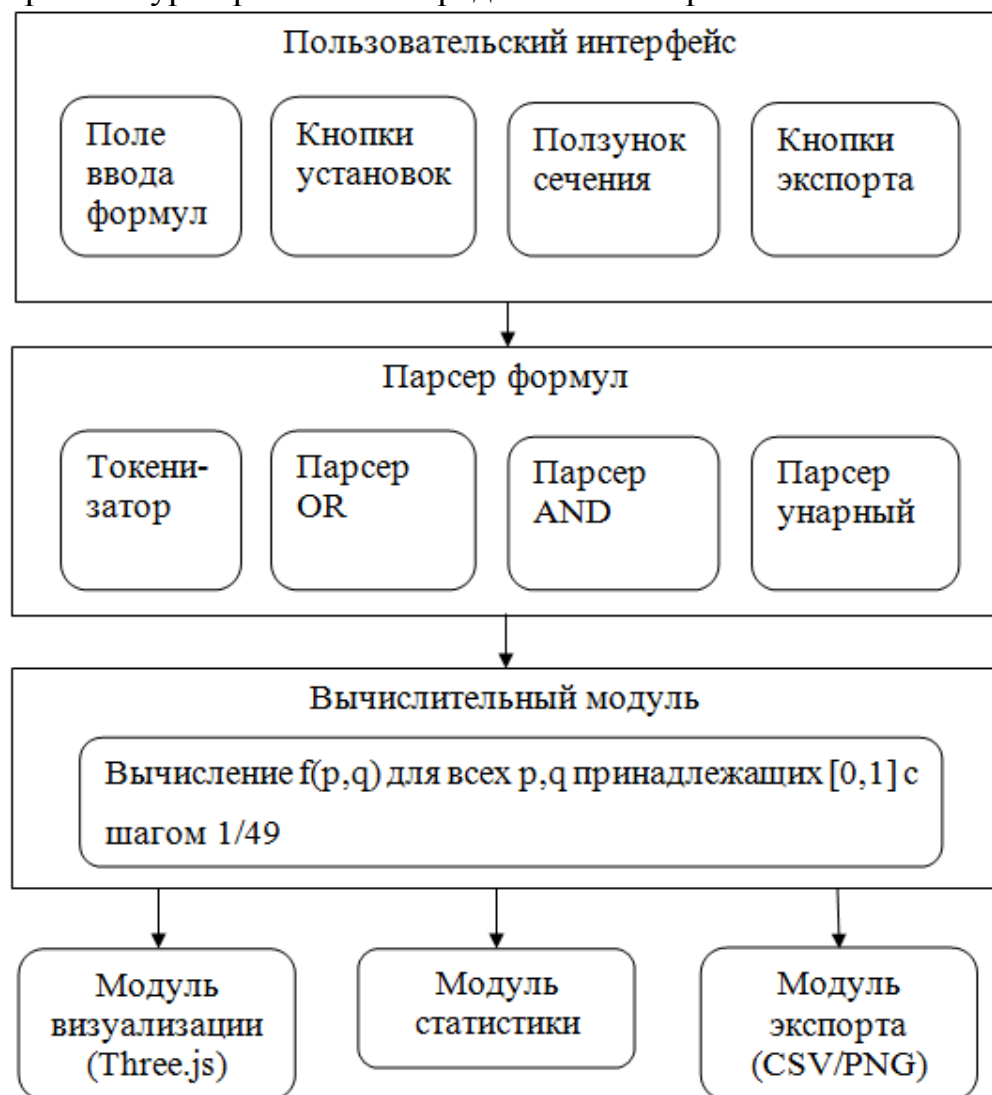


Рис. 1. Архитектура программного приложения

Парсер формул должен преобразовывать строку, введенную пользователем, в функцию, которую можно вычислить для любых значений p и q . В модуле поддерживаются следующие операции:

- & - конъюнкция (логическое И) - соответствует операции $\min(p, q)$;
- | - дизъюнкция (логическое ИЛИ) - соответствует операции $\max(p, q)$;
- ! - отрицание (логическое НЕ) - соответствует операции $1 - p$;
- p и q - нечеткие переменные;
- (и) - скобки для изменения приоритета операций.

Парсинг реализован с использованием метода рекурсивного спуска, который является классическим подходом для разбора арифметических и логических выражений. Алгоритм состоит из следующих этапов:

Этап 1. Токенизация.

Токенизатор разбивает входную строку на лексемы (токены) - минимальные значимые элементы выражения.

Этап 2. Рекурсивный спуск.

Парсер построен с учетом приоритета операций:

- наивысший приоритет - унарная операция ! (отрицание);
- средний приоритет - бинарная операция & (конъюнкция);
- низший приоритет - бинарная операция | (дизъюнкция).

Пользовательский интерфейс организован в виде панели управления, расположенной в нижней части экрана.

Для реализации интерфейса были использованы следующие технологии и библиотеки, приведенные в табл. 3 [5-7].

Таблица 3

Используемые технологии

Технология	Версия	Назначение
HTML5	-	Структура веб-страницы
CSS3	-	Стилизация интерфейса
JavaScript	ES2020	Логика приложения
Three.js	r128	3D-графика и визуализация
CSS2DRenderer	r128	Текстовые подписи в 3D-пространстве

Панель управления пользовательского интерфейса содержит следующие элементы:

- поле ввода формул - текстовое поле для ввода аналитической формулы;
- кнопки установок - быстрый выбор стандартных формул (AND, OR, XOR, импликация и др.);
- ползунок сечения - управление положением плоскости сечения по q ;
- селектор режима визуализации - выбор способа отображения поверхности;
- панель статистики - отображение числовых характеристик функции;
- кнопки экспорта - скриншот, экспорт CSV, сброс камеры, анимация.

Панель статистики отображает следующие характеристики текущей функции:

- текущая функция - название или усеченная формула;

- $\min(f(p,q))$ - минимальное значение функции на области определения;
- $\max(f(p,q))$ - максимальное значение функции;
- среднее - среднее арифметическое всех значений функции;
- дисперсия - мера разброса значений функции относительно среднего значения функции;
- сечение - диапазон значений на текущем сечении.

Результаты опытной апробации

Для проверки корректности работы разработанной программной реализации была выбрана формула $R = \min(p, q, r)$. В формуле p - кредитная история заемщика (значение от 0 до 1, 1 соответствует отличной истории), q - уровень дохода (также от 0 до 1, где 1 - высокий доход), r - качество обеспечения (от 0 до 1, где 1 - надежное обеспечение).

Для наглядности каждый тестовый случай сопровождался визуализацией: на трехмерной поверхности истинности фиксировался скриншот с желтым маркером, указывающим точное положение конкретного заемщика в пространстве параметров p, q, r .

Пример визуализации поверхности кредитоспособности приведен на рис. 2.

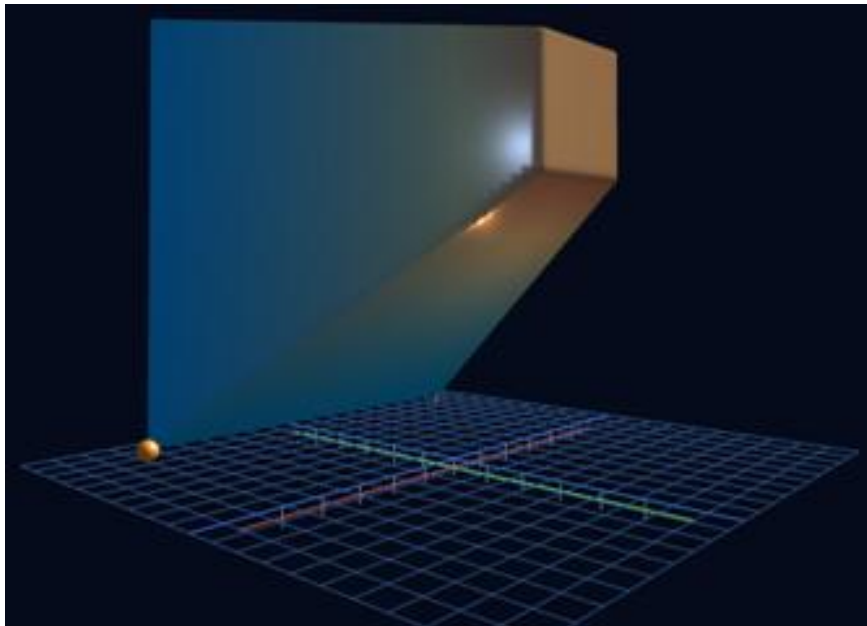


Рис. 2. Поверхность кредитоспособности $R = \min(p, q, 0.65)$

Поверхность кредитоспособности имеет характерную форму «усеченного угла». На участке, где значение минимума из p и q не превышает 0,65, итоговая оценка R равна этому минимуму, поэтому поверхность плавно поднимается от нуля до 0,65. Однако как только минимум из p и q становится больше 0,65, значение R фиксируется на уровне 0,65, образуя плоское «плато». Такая форма поверхности наглядно демонстрирует важный принцип консервативной стратегии кредитования: даже при отличных показателях кредитной истории и дохода итоговая оценка кредитоспособности ограничивается уровнем обеспечения. Цветовая кодировка поверхности помогает быстро интерпретировать результат: синий цвет ($R=0$) соответствует зоне отказа в выдаче кредита, зеленый ($R=0.5$) - зоне неопределенности, когда требуется

дополнительный анализ или повышенная ставка, а желтый и красный ($R=1$) - зоне надежных заемщиков, которым кредит может быть одобрен на максимально выгодных условиях.

Поверхность первого заемщика из примера приведена на рис. 3.

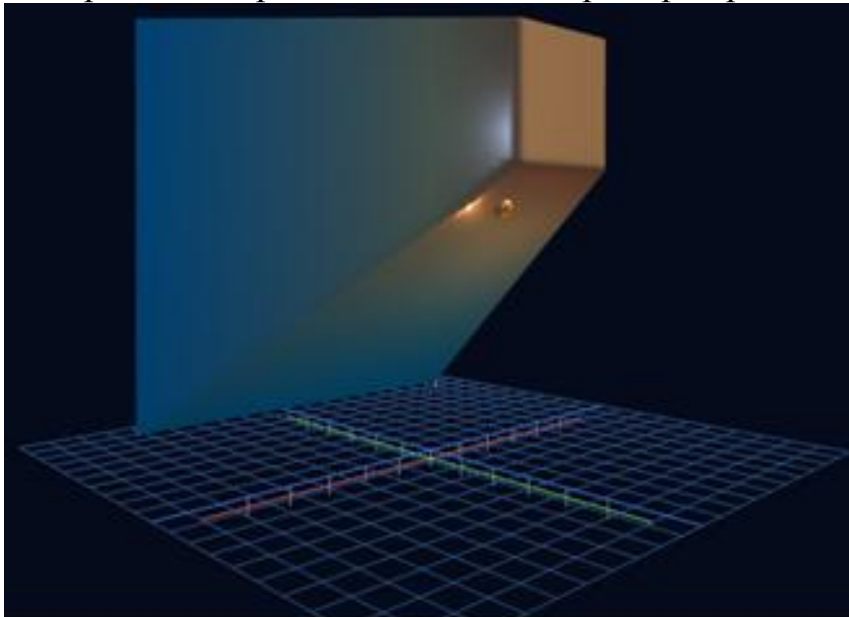


Рис. 3. Положение первого заемщика на поверхности при $r=0.65$

Для первого заемщика, характеризующегося хорошей кредитной историей ($p=0.7$), средним уровнем дохода ($q=0.55$) и достаточным обеспечением ($r=0.65$), итоговая оценка кредитоспособности составила $R=0.55$. На визуализации маркер, обозначающий положение данного заемщика, располагается в зелено-желтой зоне поверхности, что в соответствии с принятой цветовой кодировкой интерпретируется как «условно надежный заемщик». Желтая сфера маркера хорошо заметна на поверхности, что облегчает визуальный анализ и позволяет быстро определить итоговую оценку. Данный случай демонстрирует сбалансированную ситуацию, когда ни один из трех показателей не является критически низким, но и не достигает максимальных значений, в результате чего заемщик попадает в пограничную зону между надежностью и риском.

Поверхность второго заемщика из примера приведена на рис. 4.

Второй заемщик обладает отличной кредитной историей ($p=0.90$), высоким уровнем дохода ($q=0.95$) и надежным обеспечением ($r=0.90$). Его итоговая оценка кредитоспособности составила $R=0.90$. На визуализации маркер данного заемщика располагается в красной зоне на поверхности при фиксированном значении обеспечения $r=0.90$. В соответствии с принятой цветовой кодировкой интерпретируется как «очень надежный заемщик». Данный случай демонстрирует идеальную ситуацию для кредитора, когда все три ключевых параметра находятся на высоком уровне, и ни один из них не ограничивает итоговую оценку. Такому заемщику может быть рекомендовано одобрение кредита на максимальную сумму с льготной процентной ставкой.

Заключение

Разработан интервальный метод оценки истинности нечетких функций,

позволяющий определять диапазон значений функции, если известны интервалы принадлежности входных переменных. Получены условия попадания простейших аналитических функций в заданный промежуток, что имеет важное практическое значение для систем поддержки принятия решений.

Формализованы лингвистические переменные для кредитной истории, дохода и обеспечения. На основе экспертного правила сформулирована модель кредитоспособности $R = \min(k, d, c)$, отражающая консервативную стратегию кредитования.

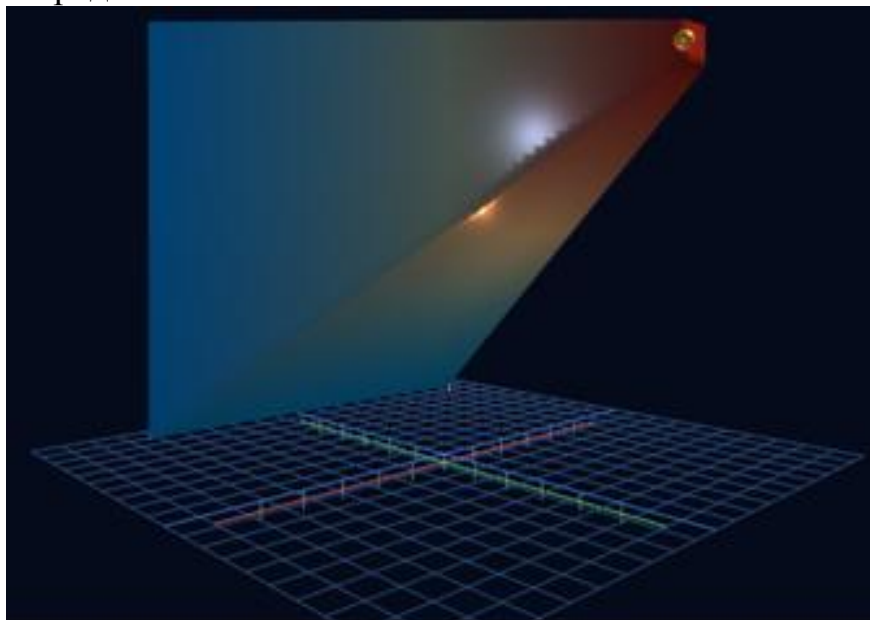


Рис. 4. Положение второго заемщика на поверхности при $r=0.90$

Разработанная программа обеспечивает ввод произвольных аналитических формул от двух или трех переменных с поддержкой операций конъюнкции, дизъюнкции, отрицания и скобок. Она вычисляет значения функции на всей области определения и визуализирует полученную поверхность в трехмерном пространстве с цветовой кодировкой от синего цвета (ложь) до красного цвета (истина).

Визуальный анализ поверхности кредитоспособности при различных фиксированных значениях обеспечения позволил наглядно наблюдать, как изменение каждого из параметров влияет на итоговую оценку. Желтый маркер, указывающий положение конкретного заемщика на поверхности, делает интерпретацию результатов интуитивно понятной.

Проведен анализ чувствительности модели, показавший, что при использовании консервативной стратегии все три фактора одинаково важны - ухудшение любого из них приводит к соответствующему снижению итоговой оценки. Экспериментально подтверждено, что при снижении обеспечения ниже определенного порога именно оно становится ограничивающим фактором.

Список использованных источников

1. Бахусова Е.В. Элементы теории нечетких множеств. - Тольятти: Издательство Тольяттинского государственного университета, 2013. - 116 с.
2. Бахусова Е.В. Нечёткая математика для программистов. - Тольятти: Филиал Рос-

сийского государственного социального университета в г. Тольятти, 2012. - 88 с.

3. Чернышев С.А. Основы программирования на Python. - М.: Издательство Юрайт, 2024. - 349 с.

4. Федоров Д.Ю. Программирование на языке высокого уровня Python. - М.: Издательство Юрайт, 2024. - 227 с.

5. Сергеев М.Ю., Сергеева Т.И., Гребенникова Н.И. Проектирование и разработка адаптивных кроссбраузерных веб-приложений// Интеллектуальные информационные системы: тр. Междунар. НПК. - Воронеж: ВГТУ, 2022. - С. 27-29

6. Сергеев М.Ю., Гребенникова Н.И. Разработка информационных клиент-серверных систем прикладного назначения с применением веб-технологий// Экономика и менеджмент систем управления, 2024. - № 3 (53). - С. 80-89.

7. Свиноухов Д.Д., Болдырев И.Р., Сергеев М.Ю. Разработка системы реального времени для взаимодействия с пользователем// Информационные технологии моделирования и управления. - 2024. - № 1 (135). - С. 74-80.

Издательство "Научная книга",
сообщает о требованиях, предъявляемых к статьям,
представляемым в научно-практический журнал
"Экономика и менеджмент систем управления"

Языки: русский; английский.

Основные направления:

организация производства;
стандартизация и управление качеством продукции;
системный анализ, управление и обработка информации;
управление в социальных и экономических системах;
экономика и управление народным хозяйством;
математические и инструментальные методы экономики.

Даты: научно-практический журнал "Экономика и менеджмент систем управления" издается не реже 4 выпусков в год.

Требования к материалам

1. Материалы предоставляются только по электронной почте emsu@bk.ru в единственном присоединенном к письму файле-архиве (WinRar, WinZip).

2. Материалы должны содержать инициалы и фамилии авторов, название (большими буквами), аннотацию (до 5 строк), ключевые слова (до 4 слов или словосочетаний) - все на русском и английском языках, а также полное название организации, представляющей статью.

3. Размер статьи должен находиться в пределах от 8 до 14 страниц стандартного машинописного текста (Word версии до 2003 включительно, при размере шрифта 14 pt, шрифт Times New Roman, страница A4, портретная ориентация, поля 25 мм всюду, одинарный межстрочный интервал). Список использованных источников обязателен.

4. Рисунки включаются в текст статьи, а также должны содержаться в отдельных графических файлах (форматы bmp, jpg, gif, tif, wmf). Размер шрифта в рисунках – не менее 12 pt.

В архиве с материалами в отдельном файле должны содержаться:

- сведения обо всех авторах (фамилия, имя, отчество, место работы и должность, ученая степень, звание, почтовый - с индексом - и электронный адрес);

- указание на количество заказываемых экземпляров (минимальное количество экземпляров, заказываемых авторами - не менее половины их количества с округлением в большую сторону);

- обязательство уплаты оргвзноса ориентировочно около 320 (380 - вне России) рублей (при оплате авторами) за одну страницу статьи в одном экземпляре журнала вместе со стоимостью пересылки в ценах марта 2022 г.

Цена одной страницы при безналичной оплате - 420 руб., не включая НДС.

Подписной индекс журнала «Экономика и менеджмент систем управления» в объединенном каталоге «Пресса России» - 43054
--