Научная статья
УДК 621.391
https://doi.org/10.31854/1813-324X-2024-10-3-24-34

Динамические туманные вычисления и бессерверная архитектура: на пути к зеленым ИКТ

Артем Николаевич Волков, artem.nv@sut.ru

Санкт-Петербургский государственный университет телекоммуникаций им. проф. М. А. Бонч-Бруевича, Санкт-Петербург, 193232, Российская Федерация

Аннотация

Актуальность. *В условиях роста парка оборудования центров обработки данных, развития сетей IMT-2020 и появлением услуг Телеприсутствия сетей IMT-2030 особо актуальным направлением современных исследований является поиск нетривиальных, нестандартных подходов и решений в области обеспечения вычислительными и сетевыми ресурсами. Данная статья освещает актуальные вопросы инфраструктурного направления сетей IMT-2030 – динамических туманных вычислений. Рассматривается вклад данной технологии для повышения эффективности используемых ресурсов, приводятся актуальные сценарии сетей IMT-2030. В частности, исследуется задача поиска группы устройств в туманных вычислениях для последующей миграции типовых контейнеров платформы FaaS.*

Постановка задачи: *исследование вопросов совместного использования бессерверной архитектуры и динамических туманных вычислений для эффективного распределения нагрузки услуг Телеприсутствия.*

Цель работы: *исследование и разработка эффективного метода распределения группы микросервисов в динамических туманных вычислениях.*

Используемые методы: *исследуемые алгоритмы относятся к типу метаэвристических алгоритмов для решения задач многокритериальной оптимизации. Для апробации метода был разработан сегмент лабораторной сети, который послужил генератором реальных данных работы тестируемых платформ в условиях роста нагрузки. На базе серии экспериментов были собраны данные для последующего моделирования предложенного метода, который, в свою очередь, был реализован на языке программирования Python.*

Анализ ***результатов*** *показал эффективность предложенного метода в рамках поставленной задачи, что, в конечном итоге, позволяет значительно быстрее принимать решение о миграции.*

Новизна: *разработаны модель и метод для бессерверной архитектуры с миграцией групп микросервисов на группы устройств туманных вычислений в условиях их подвижности, и использован метаэвристический алгоритм стаи серых волков с целью определения группы устройств для последующей миграции типовых микросервисов.*

Практическая значимость: *разработанная модель и метод могут быть использованы при реализации туманных вычислений в условиях подвижности устройств, в том числе с целью достижения требований перспективных услуг Телеприсутствия.*

Ключевые слова: *IMT-2030, туманные вычисления, услуги Телеприсутствия, бессерверная архитектура, метаэвристические алгоритмы*

Источник финансирования: *статья подготовлена в рамках мегагранта Минобрнауки по соглашению № 075-15-2022-1137*.

Ссылка для цитирования: Волков А.Н. Динамические туманные вычисления и бессерверная архитектура: на пути к зеленым ИКТ. 2024. Т. 10. № 3. С. 24–34. DOI:10.31854/1813-324X-2024-10-3-24-34. EDN:QOELMJ

Original research
https://doi.org/10.31854/1813-324X-2024-10-3-24-34

Dynamic Fog Computing Towards Green ICT

Artem N. Volkov, artem.nv@sut.ru

The Bonch-Bruevich Saint Petersburg State University of Telecommunications,
St. Petersburg, 193232, Russian Federation

Annotation

Relevance: *In the context of the growing fleet of data center equipment, the development of IMT-2020 networks and the imminent emergence of Telepresence services of IMT-2030 networks, a particularly relevant area of modern research is the search for non-trivial, non-standard approaches and solutions in the field of provision of computing and network resources. This article covers current issues in the infrastructure direction of IMT-2030 networks - dynamic fog computing. The contribution of this technology to improve the efficiency of used resources is considered, and current scenarios for IMT-2030 networks are presented. In particular, we study the problem of searching for a group of devices in the computing fog for subsequent migration of typical FaaS platform containers.*

Problem statement: *Research on the joint use of serverless architecture and dynamic fog computing for efficient load distribution of telepresence services.*

Goal of the work: *Research and development of an effective method for distributing a group of microservices in dynamic fog computing.*

Methods: *the algorithms under study belong to the type of metaheuristic algorithms for solving multicriteria optimization problems. To test the method, a laboratory network segment was developed, which served as a generator of real data on the operation of the tested platforms under conditions of increasing load. Based on a series of experiments, data was collected that formed the basis for subsequent modeling of the proposed method, which in turn was implemented in the Python programming language.*

Result*: Analysis of the results showed the effectiveness of the proposed method within the framework of the task, which ultimately makes it possible to make a decision on migration many times faster.*

Novelty: *A model and method for serverless architecture have been developed for migrating groups of microservices to groups of fog computing devices, under conditions of their mobility, and a meta-heuristic algorithm of a pack of gray wolves has been used to determine a group of devices for subsequent migration of typical microservices.*

Practical significance*: The developed model and method can be used in the implementation of fog Computing, in conditions of device mobility, including in order to achieve the requirements of promising Telepresence services.*

Keywords: *IMT-2030, fog computing, telepresence services, serverless architecture, metaheuristic algorithms*

Funding: *The article was prepared within the framework of a megagrant from the Ministry of Education and Science under agreement No. 075-15-2022-1137*.

For citation: Volkov A.N. Dynamic Fog Computing Towards Green ICT. *Proceedings of Telecommunication Universities.* 2024;10(3):24–34. (in Russ.) DOI:10.31854/1813-324X-2024-10-3-24-34. EDN:QOELMJ

Введение

Общее стремление к цифровизации процессов в государстве, бизнесе и обществе одновременно приносит как положительные аспекты – повышение эффективности и прозрачности систем, так и новые вызовы и трудноразрешимые задачи. Принятие и внедрение концепции Интернета Вещей (ИВ), масштабы которой 10 лет назад были недооценены, и последующие базирующиеся на ней технологические направления привели к фундаментальным изменениям не только в области конкретных технологий и решений, но и во взглядах на услуги и саму сеть. Так, развитие платформ ИВ, позволяющих обрабатывать гигабайты данных и получать новые знания с использованием инструментов искусственного интеллекта (ИИ), привело к бурному развитию центров обработки данных (ЦОДы). Последующее развитие ИВ породило различные концепции услуг, например таких, как: Тактильный Интернет, Интернет Навыков и другие. Приведенные примеры относятся к сверхнадежным сетям с ультрамалыми задержками

(URLLC, *аббр. от англ*. Ultra-Reliable Low Latency Communications), которые являются одним из фундаментальных направлений сетей IMT-2020. Стоит отметить, что принятая в конце 2023 г. концепция сетей IMT-2030 естественным образом стала преемницей IMT-2020, в тоже время увеличив требования и представив такие сценарии, как TIRO (*аббр. от англ*. Tactile Internet for Remote Operations) и HTC (*аббр. от англ*. Holographic Type Communications). Сети IMT-2030 определяют следующие «технологические измерения»:

– Massive Communication (*пер. с англ.* – массовые коммуникации), что, в свою очередь, является расширением massive Machine Type Communication (mMTC);

– Immersive Communication (*пер. с англ.* – иммерсивные коммуникации), является расширенным enhanced Mobile Broadband (eMBB); при этом важно отметить, что новый сценарий включает в себя и поднаправления mMTC и URLLC;

– Hyper Reliable & Low-Latency Communication (HRLLC, *пер. с англ.* – гипернадежные сети связи с ультрамалой задержкой), представляют развитие направления URLLC-сетей IMT-2020;

– Artificial Intelligence in Communication (AI in Communication, *пер. с англ.* – искусственный интеллект в связи) или автономные сети связи; на сегодня можно обнаружить достаточно большой пласт разработанных Рекомендаций МСЭ-Т (Международный союз электросвязи, сектор стандартизации телекоммуникаций) в 13 ИК (Исследовательской комиссии), где определены фундаментальные решения для последующей имплементации технологий ИИ в сети связи;

– Ubiquitous Connectivity (*пер. с англ.* – повсеместная связь): данное направление, в первую очередь, раскрывает концепцию МСЭ сетей 2030 STIN (*от англ.* Space-Terrestrial Integrated Network – космически-наземная интегрированная сеть) и является предложением по решению ряда целей устойчивого развития ООН, в том числе для сокращения цифрового разрыва с удаленными районами и поселения;

– Integrated Sensing and Communication (*пер. с англ.* – интегрированное зондирование и связь): формирует целый пласт задач в области позиционирования в сети IMT-2030 и тесно связано с другими сценариями ее применения.

В результате бурного развития ИВ и других сценариев сетей IMT-2020 несколько лет назад был сформирован тренд на декомпозицию систем вычислений. Граничные вычисления позволили, благодаря своей архитектуре, приблизить реализацию ряда услуг, а также снизить нагрузку на ядро сети. В то же время автономные сети призваны разрешить вопросы интеллектуального распределения ресурсов как сетевых, так и вычислительных, которые мягко интегрированы в системы управления сетью. В итоге глобальный тренд на абстрагирование программного обеспечения сетевых и вычислительных сущностей от аппаратной части сформировал технические возможности реализации программного слайсинга ресурсов. Пройдя вышеприведенные этапы развития сети и вычислительных систем, стала более ясна цель – максимальная автономизация инфраструктуры с рациональным использованием вычислительных и сетевых ресурсов на базе алгоритмов машинного интеллекта.

Если обратиться к отчетам статистического агентства Straits Research [1], можно заметить продолжающийся рост объема рынка оборудования для ЦОДов. Ожидается, что к 2031 г. он достигнет 164,36 млрд долларов США, а среднегодовой темп роста (рисунок 1) составит 13,2 % (CAGR, *аббр. от англ.* Compound Annual Growth Rate – совокупный среднегодовой темп роста) в течение прогнозируемого периода (2023–2031 гг.).

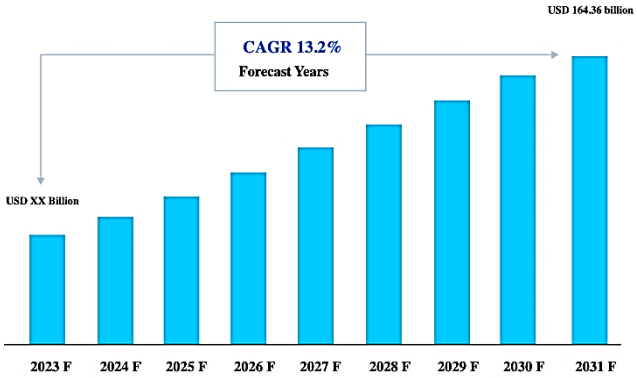


Рис. 1. Рост рынка оборудования ЦОД [1]

Fig. 1. Data Center Equipment Market Growing [1]

С учетом планируемых услуг Телеприсутствия сетей IMT-2030 цифры роста могут быть изменены в бóльшую сторону, и факт объема строящихся ЦОДов заставляет задуматься об эффективности используемого сетевого и вычислительного оборудования, а также увеличении потребления энергоресурсов для обеспечения работы данного парка оборудования.

В работе [2] приводится проблема создания зеленых инфокоммуникационных технологий (Green ICT). Исследуются как организационные, так и информационно-технологические способы перехода к Green ICT, то есть на рациональное энергопотребление. К примеру, приводится сравнительный анализ потребности энергии для судна-контейнеровоза «Emma-Maersk», одного из крупных ЦОДов европейской части России, где сравнительно наглядно становится понятно, насколько ЦОДы потребляют электроресурсов. Существует немало работ в области Green ICT [3–6], а также – немало решений в данной области,

которые внедряются различными компаниями по всему миру. Стоит также отметить то, что данная повестка активно исследуется и прорабатывается на уровне международных рекомендаций МСЭ-Т Исследовательской комиссией № 5. В частности, можно найти серию Рекомендаций МСЭ-Т L.1300-L.1399: Энергоэффективность, умная энергетика и экологически чистые центры обработки данных, а также немало технических отчетов и спецификаций. МСЭ-Т дает следующее определение: «Зеленый» или устойчивый ЦОД – это хранилище для управления и распространения данных, в котором механические, осветительные, электрические и компьютерные системы спроектированы с учетом максимальной энергоэффективности и минимального воздействия на окружающую среду. В работе [7] дано следующее определение Green ICT как совокупности способов, призванных уменьшать вредное воздействие на человека и окружающую среду, эффективно использовать ресурсы, которые предоставляет природа и одновременно повышать производительность систем в расчете на единицу потребляемых физических ресурсов. Существует также принцип Р. Ландауэра, который позволяет установить связь между объемом данных и энергозатратами, при этом независимо от физики и технологии вычислительного процесса в случае потери одного бита данных в ходе вычисления как минимум выделяется энергия E (Дж). Как бы это «парадоксально» не звучало, принцип Ландауэра означает, что компьютер потребляет тем меньше энергии, чем меньше вычислительных операций он выполняет. Если же рассматривать принципы обработки данных на базе ЦОД, то необходимо учесть добавочные затраты энергоресурсов, которые необходимы для постоянной работы систем охлаждения и других вспомогательных систем, а также систем питания сетей связи, которые обеспечивают доступ к ЦОД.

Таким образом, с точки зрения принципа Ландауэра, а также учитывая вышеприведенные исследования в области трендов рынка и развития технологий услуг Телеприсутствия и сетей IMT-2030, в настоящее время существует потребность в поиске более эффективных методов построения инфраструктуры. В частности, ожидается, что декомпозиция ЦОДов в меньшие центры, а также внедрение алгоритмов ИИ и более инновационных материалов, а также использование возобновляемых источников энергии позволит снизить общий эффект на окружающую среду. При этом декомпозиция вычислительной архитектуры и систем «эпохи» до туманных вычислений (Fog), в том числе динамических, позволит исключить некоторые составляющие в общей формуле потребления энергии на удельный размер вычислительной задачи. Здесь использование устройств пользователя и устройств ИВ не требует дополнительных систем охлаждений, резервирования и построения технологичных зданий с надстроенными инженерными системами для его функционирования. В таком случае концепция Fog не только позволяет обеспечить требования к качеству обслуживания для ряда услуг при условии микросервисных архитектур, но и снизить общую суммарную потребность в энергоресурсах. Де-факто Fog являются инструментом бережливого производства в области инфокоммуникационных технологий и систем связи.

Динамические туманные вычисления и бессерверная архитектура услуг

Бессерверная архитектура (serverless) и контейнеры могут быть интегрированы для создания высокомасштабируемых и эффективных платформ высоконагруженных услуг. Контейнеры могут использоваться для упаковки и развертывания функций serverless, что позволяет разработчикам услуг использовать преимущества обеих технологий. К преимуществам совместного использования технологий можно отнести следующие:

– упрощенное развертывание (контейнеры упрощают развертывание функций serverless, поскольку они уже содержат все зависимости программного обеспечения в виде библиотек и фреймворков);

– портативность (контейнер позволяет быстро переносить функции serverless между различными облачными платформами и локальными средами);

– безопасность (контейнеры обеспечивают изоляцию и безопасность между serverless-функциями, что в итоге снижает риск уязвимостей).

Для анализа работы сети Fog необходимо использовать модель, которая позволит описать функционирование сети в условиях изменяющейся архитектуры ввиду мобильности Fog-устройств как рамках кластера, так и в рамках *туманностей* (Nebula) [8]. При этом данные кластера и/или туманности обладают характеристикой гетерогенности как самих Fog-устройств, так и собственно структур. Таким образом, с учетом вышеприведенных допущений, в качестве математической модели может быть использована модель точечного процесса, учитывая гетерогенность структур и Fog-устройств. При этом, в данных условиях процессы Неймана – Скотта, относящиеся к точечным процессам с кластеризацией (то есть с объединениями устройств), будут предпочтительнее для построения модели. Так как процессы без кластеризации применяются для моделирования однородных систем, где объекты образуют одно единственное поле, размещенное на плоскости или в пространстве (такая модель может быть применима в частных случаях при использовании

однородных вычислительных ферм). Более подробные исследования в области применения точечных процессов для распределенных Fog представлены в следующей статье [9]. Стоит отметить, что Fog обладают отчасти противоречивыми характеристиками. Например, устройства динамических Fog могут образовывать mesh-сеть, при этом обладающую характеристикой самоорганизации с возможностью горизонтального масштабирования, что больше напоминает самоорганизующиеся сенсорные сети. В то же время, Fog являются вычислительным кластером, на базе которого могут быть развернуты платформы, услуги и др. Соответственно, для разрешения задач может быть перенят опыт исследований и разработок в вышеуказанных областях знаний. Так, в работах [10, 11] приводятся исследования применения аппарата точечных процессов, в частности процессов Томаса для формирования математических моделей в сверхплотных сенсорных сетях ИВ, что говорит об актуальности использования данной математической базы для перспективных сетей и услуг.

В данной работе необходимо решить задачу поиска группы устройств в вычислительной *туманности*, которая может быть представлена в виде группы или одного кластера Fog. При этом делается допущение о том, что все устройства, находясь в условиях мобильности относительно базовой станции или сетевого координатора, стремятся сохранять единый вектор перемещения при минимально возможных отклонениях между собой. На практике такой сценарий может быть обнаружен в быстрых поездах типа Сапсан, самолетах и других похожих условиях (общественный транспорт, проспекты и т. п).

Таким образом, опираясь на сценарий, модель сети Fog при поиске группы Fog-устройств для дальнейшей живой миграции 4-х типовых контейнеров платформы FaaS может быть представлена в виде рисунка 2. Каждое из Fog-устройств – набор параметров, характеризующих устройство с точки зрения элемента вычислительного кластера: CPU (количество и частота ядер процессора), GPU (производительность графического процессора), Network (тип интерфейса и тип технологии связи, в том числе возможная скорость подключения), и многие другие. В текущей модели, для развертывания FaaS-unit, с 4-мя типовыми контейнерами ($MS_i, \forall\, i \in N$) и оценки предлагается использовать временны́е характеристики реализации задачи. Вводя типизацию контейнеров MS_i, возможно перейти к оценке характеристик самого Fog-устройства, а также собственно контейнера в пространстве времени, а именно: времени реализации задачи типовым контейнером T_c (мс) и задержки передачи соответствующего объема информации T_r (мс), который является конечным. Типовой контейнер решает одну задачу: например, сортировка данных методом пузырька, где определен соответствующий формат и объем принимаемых данных, а также формат и объем передаваемых результатов.

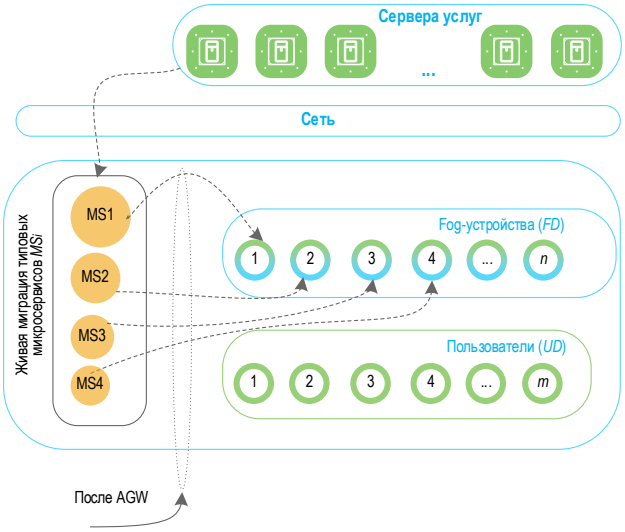


Рис. 2. Модель сети туманных вычислений при поиске группы Fog-устройств

Fig. 2. Fog Computing Network Model for Group of Fog-Devices Searching

Таким образом, необходимо определить параметры, описывающие каждое из исследуемых Fog-устройств. При этом некоторые параметры могут оцениваться на уровне сравнения с предельными значениями: например, количество выделенной логической ОЗУ, необходимой для работы готовящегося к миграции микросервиса или группы микросервисов. В рамках данной задачи были определены параметры, описывающие Fog-узел с точки зрения затраченного времени реализации типовой задачи.

В общем виде целевая функция оценки устройства может быть представлена следующим образом:

$$F = \sum_{i=1}^{m} k_i P_i, \qquad (1)$$

где $0 \le k_i \le 1$ – коэффициенты, при этом $\sum k_i = 1$; P_i – параметры оценки, при m – количество параметров для $\forall\, i > 0$, $i \in N$, а основная задача сформулирована в виде выражения:

$$\{F_1, F_2, F_3, F_4\} = \arg\min_{T_r, T_c, S}\{F\}, \qquad (2)$$

где $\{F_1, F_2, F_3, F_4\}$ – ряд решений $F(T_r, T_c, S)$ в порядке неубывания их значений, то есть: $F_1 \le F_2 \le \le F_3 \le F_4$. T_c – время выполнения типовой задачи в типовом контейнере MS_i. Параметр T_r характеризует связь узла с основным шлюзом-брокером в соответствующей зоне Fog, через который проходят все транзакции между устройствами и, соот-

ветственно, контейнерами; может быть представлен как задержка при передаче типового запроса с соответствующим объемом данных до шлюза-брокера. Параметр S описывает степень стабильности кластера Fog, в частности рассматривается стабильность соответствующего устройства в кластере Fog. При этом вероятность того, что расстояние между элементом кластера Fog и шлюзом-брокером превысит величину R, может быть оценено как [9]:

$$p(d > R) = 1 - \mathrm{Dis}(R, \delta), \quad (3)$$

где $\mathrm{Dis}(R, \delta)$ – функция распределения вероятности расстояния между между устройством и координатором. В зависимости от условий, она может быть описана Гамма-распределением или его частными формами, например, распределением Пирсона или распределением Райса. R – расстояние между устройством и координатором. При этом вероятность того, что расстояние между координатором и хотя бы одним из элементов кластера превышает величину R определяется как [9]:

$$p(> R) = 1 - \prod_{i=1}^{k} \mathrm{Dis}(R, \delta_i), \quad (4)$$

где k – число устройств кластера Fog; δ_i – параметр распределения вероятности расстояния.

Таким образом, в работе [9] было сформулировано предложение о максимальной стабильности Fog-кластера, где под стабильностью понимается стремление сохранить структуру, то есть связи между координатором и элементами кластера:

$$\min_R \left\{ 1 - \prod_{i=1}^{k} \mathrm{Dis}(R, \delta_i) \right\}. \quad (5)$$

Стабильность кластера Fog может быть представлена как оценка, описанная в виде функции минимизации вероятности изменения географического состояния устройства относительно так называемого центра масс кластера или шлюза-брокера в кластере Fog, через который проходят все транзакции. Данная задача может быть решена с использованием точечных процессов Неймана – Скотта [9].

Соответственно, стабильность самого устройства относительно его отношения к шлюзу-брокеру будет представляться в следующим виде:

$$\min_R \{1 - \mathrm{Dis}(R, \delta)\}, \quad (6)$$

а используемый параметр оценки стабильности S в основной целевой-функции может быть представлен следующим образом:

$$S = \max \left\{ \min_R \{1 - \mathrm{Dis}(R, \delta)\} \right\}. \quad (7)$$

В данной работе предлагается рассмотреть частный случай, как ранее было упомянуто, в качестве исследуемых параметров взять оценку Fog-узла с точки зрения T_c и T_r, а также параметра оценки стабильности устройства в кластере, где $0 < S \leq 1$.

Таким образом, целевая функция, исследуемая в данной работе, может быть представлена в следующем виде:

$$F = \sum_{i=1}^{m} k_i P_i = k_1 T_r + k_2 T_c + k_3 S. \quad (8)$$

Данный метод может быть использован при выполнении условия типизации контейнеров для развертывания serverless. В общем случае, метод может быть сложно реализуем с точки зрения необходимости предварительного тестирования среднего времени выполнения функции соответствующим нетиповым контейнером общей услуги.

Для решения поставленной задачи был исследован класс метаэвристических алгоритмов, которые позволяют определить глобальный экстремум фитнес-функции. Существует немало метаэвристических алгоритмов. Рассмотрим основные: алгоритм стаи серых волков (GWO, *аббр. от англ.* Grew Wolf Optimizer), алгоритм оптимизации роя частиц (PSO, *аббр. от англ.* Particle Swarm Optimization), генетический алгоритм (GA, *аббр. от англ.* Genetic Algorithm) и алгоритм роя сальп (SSA, *аббр. от англ.* Salp Swarm Algorithm). Каждый из этих алгоритмов имеет свои уникальные особенности и преимущества. К их сходству можно отнести следующие критерии: все они являются популяционными (то есть, они работающими с популяцией кандидатных решений) и используют итеративный процесс поиска.

В целом GWO, PSO, GA и SSA являются мощными и эффективными алгоритмами, которые полезно использовать для решения широкого спектра оптимизационных задач. Выбор алгоритма зависит от конкретной задачи и требований к производительности. Так, например, генетический алгоритм является более ресурсозатратным, что также влечет за собой скорость/время сходимости.

В результате анализа особенностей данных алгоритмов, в частности при рассмотрении требований к скорости сходимости, простоте (в том числе при настройке параметров), предлагается использовать алгоритм GWO (метаэвристический алгоритм, который был вдохновлен социальным поведением серых волков). Серые волки – это социальные животные, которые живут в стае, возглавляемой альфа-самцом. Метаэвристический алгоритм GWO имитирует социальное поведение серых волков, чтобы решить оптимизационные задачи – поиск экстремума фитнес-функции. Алгоритм

инициализирует популяцию волков (возможных решений) и оценивает каждого волка в популяции $X_\alpha, X_\beta, X_\delta$. Затем алгоритм итеративно обновляет положения волков на основе их текущих позиций и позиций других волков в стае.

ТАБЛИЦА 1. Сравнительный анализ

TABLE 1. Comparing Analysis

	Метаэвристические алгоритмы			
	GWO	PSO	GA	SSA
Биологическая модель-источник	социальное поведение волков	поведении стаи птиц	процесс эволюции	поведение роя сальп (морские животные)
Структура популяции	волки	частицы	хромосомы	сальпы
Принцип обновления	волки обновляют свои положения на основе позиций альфа-, бета- и дельта-волков	частицы обновляют свои положения на основе своих собственных лучших положений и лучшего положения в популяции	хромосомы обновляются с помощью операций кроссовера и мутации	сальпы обновляют свои положения на основе положения передней и задней сальпы
Топологии	волки расположены в иерархической структуре	частицы обычно расположены в топологии кольца	хромосомы не имеют определенной топологии	сальпы расположены в одномерной топологии

Таким образом, на каждой итерации, в том числе финальной, алгоритм будет выдавать 4 результата, которые по возрастающей будут отражать соответствующие устройства, пригодные для последующего размещения типовых микросервисов услуг.

В GWO позиция каждого волка рассчитывается на основе следующих выражений (9–11):

$$D_\alpha = |C_1 X_a - X(i)|,\ D_\beta = |C_2 X_\beta - X(i)|,\ D_\delta = |C_3 X_\delta - X(i)|, \quad (9)$$

$$X_1 = X_a - A_1(D_\alpha),\ X_2 = X_\beta - A_2 * (D_\beta),\ X_3 = X_\delta - A_3 * (D_\delta), \quad (10)$$

$$X(x+1) = \frac{X_1 + X_2 + X_3}{3}, \quad (11)$$

где $[X_\alpha, X_\beta, X_\delta, X_\omega]$: альфа-волк ($X_\alpha$) – лучшее решение в популяции; бета-волк (X_β) – второе лучшее решение в популяции; дельта-волк (X_δ) – третье лучшее решение в популяции; омега-волки (X_ω) – остальные волки в популяции; X_i – позиция решения на соответствующей итерации i; $D_\alpha, D_\beta, D_\delta$ – вспомогательные векторы для расчета соответственно значений X_1, X_2, X_3.

На каждой итерации алгоритма обновляются коэффициенты A и C, согласно следующим выражениям (12–14):

$$A = 2a \cdot r_1 - a, \quad (12)$$

$$C = 2 r_2, \quad (13)$$

$$a = 2 - \left(i \cdot \frac{2}{I}\right). \quad (14)$$

где a – параметр, линейно уменьшающейся от 2 до 0 на каждой итерации согласно выражению (14); r_1, r_2 – равномерно распределенные случайные числа от 0 до 1; I – количество итераций.

Следственно параметрами, которые могут повлиять на эффективность алгоритма GWO, являются количество волков, количество итераций, а также параметры a и C. При этом, величина C на каждой итерации обеспечивает диверсификацию алгоритма GWO. Схематично работа алгоритма представлена на рисунке 3.

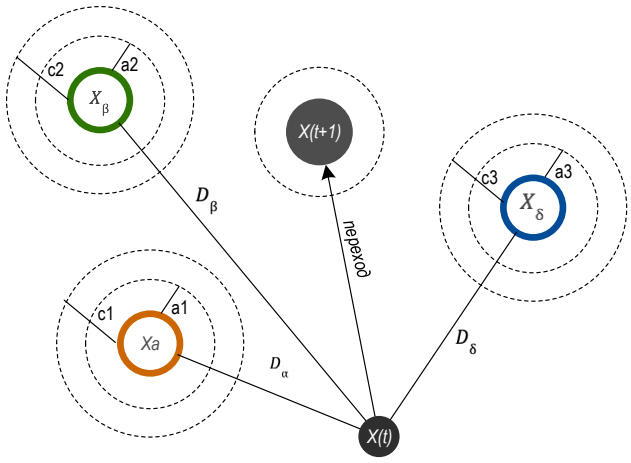


Рис. 3. Схема работы алгоритма

Fig. 3. Algorithm Scheme

Алгоритм GWO представлен в виде псевдокода:

0:**BEGIN**
1: **Инициализация популяции GWO** – $X_i(i:=1,2,\ldots,n)$;
2: **Инициализация GWO-параметров**: *a, A* и *C;*
3: **Расчет фитнес-функции** F_X для каждого агента;
4: */* комментарий:*
5: X_α *– лучший агент*
6: X_β*–- второй лучший агент*
7: X_δ *– третий лучший агент*
8: X_ω *– остальные агенты*
9: **/*
10: **Нахождение** 3-х лучших X_α, X_β, X_δ;
11: **While** достижение критерия **do**
12: **for** каждого агента **do**
13: Обновление позиции согласно с формулой (11);
14: **end for**
15: **обновить** *a, A* и *C;*
16: ***расчет*** F_X *фитнес-функции для каждого агента;*
17: **обновление** X_α, X_β, X_δ;
18: **end while**
19: **вернуть значение** X_α, *лучшее значение* $F(X_\alpha)$, *а также значение на последней итерации алгоритма* [X_β, X_δ, X_ω];
20:**END.**

GWO – это мощный и эффективный алгоритм, который можно использовать для решения широкого спектра оптимизационных задач, в том числе для поиска группы значений экстремумов функции, в том числе фитнес-функции, описывающей каждое из Fog-устройств в туманности.

Результаты моделирования

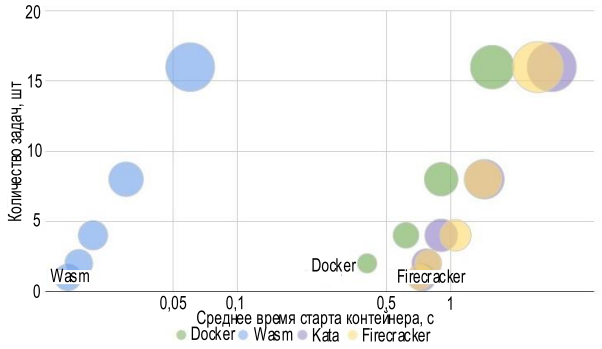
Для моделирования было проведено предварительное натурное исследование существующих платформ: Kata, Firecracker, Wasm, Docker. Тестирование проводилось на базе серверного оборудования лаборатории Meganetlab 6G кафедры сетей связи и передачи данных СПбГУТ. Для исследования были выбраны два метода распределения поступающей нагрузки:

Метод 1. Каждый новый запрос, приходящий на агент (брокер/шлюз), направлялся на типовой микросервис/контейнер, который обслуживал запросы в порядке очереди FIFO.

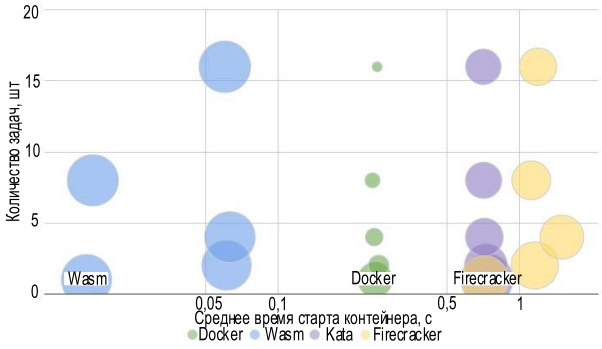
Метод 2. Каждый новый запрос, приходящий на агент, обслуживался вновь созданным клоном типового микросервиса, после чего данный клон уничтожался. Таким образом, система гибко масштабировалась под рост нагрузки.

Для этого рассматривался пример работы типового микросервиса, который был представлен в виде контейнера и решал типовую задачу – расчет числа π. На рисунке 4a в виде пузырьковой диаграммы отражены результаты тестирования первого метода на реальном стенде, где были развернуты платформы и реализованы типовые контейнеры. На рисунке 4b в виде пузырьковой диаграммы отражены результаты тестирования второго метода на реальном стенде, где при поступлении новой задачи происходил принудительный старт нового контейнера в кластере, и задача выполнялась параллельно.

На рисунке 4 размер пузырька является средним временем решения типовой задачи, вычисленным на базе пяти повторных экспериментов. При этом ось абсцисс представлена в логарифмическом масштабе на обоих рисунках. Представленные графики позволяют визуально оценить возможности существующих решений контейнеризации для сегментов Fog, учитывая различные подходы к распределению вычислений в микросервисной архитектуре. Согласно рисунку 4b, Docker при росте нагрузки и сравнительно равным (в меньшей степени изменяющимся) временем старта нового контейнера для каждый задачи уменьшает время реализации типовой задачи. При этом Docker находится на втором месте после Wasm, относительно времени старта типового контейнера.



a)



b)

Рис. 4. Диаграммы: по методу 1 (a); по методу 2 (b)

Fig. 4. Diagram for method 1 (a) and method 2 (b)

Для дальнейшего моделирования были взяты за основу данные, которые измерялись на стенде при работе Docker-контейнеров ввиду эффективности их работы при возрастающих нагрузках и большом количестве типовых микросервисов, что может быть свойственно Fog. Полученные в ходе эксперимента результаты легли в основу набора данных T_r и T_c, которые были составлены для 100 Fog-устройств, при этом структура этих данных была согласована с целевой функцией, представленной в выражении (8). Для визуализации ре-

зультата параметр устойчивости узла был установлен в виде константы $S = \text{const} = 1$, а коэффициенты k_1, k_2 и k_3 – равными 0,33. Итог моделирования Fog-устройств представлен на рисунке 6, где ось абсцисс – это значение $k_1 T_r$, ось ординат – представляет значение $k_2\ T_c$, а ось аппликат, соответственно, – значение целевой функции F. Далее на поле данных смоделированных Fog-устройств был применен алгоритм GWO, разработанный также на языке программирования Python. Результат поиска группы устройств с помощью данного алгоритма представлен на рисунке 7.

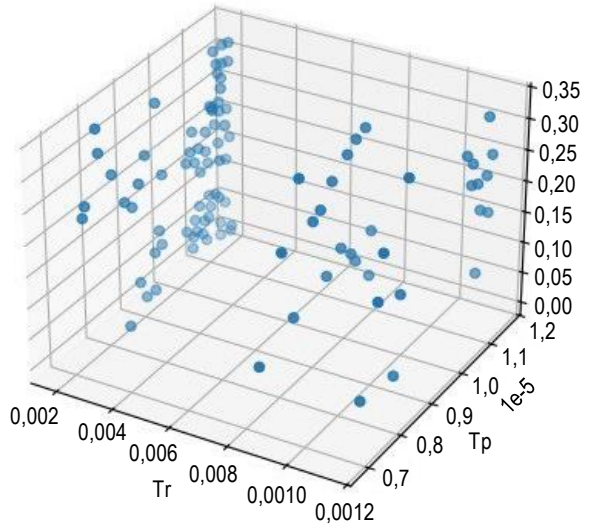


Рис. 6. Результаты моделирования Fog-устройств

Fig. 6. Results of Fog-Devices Modeling

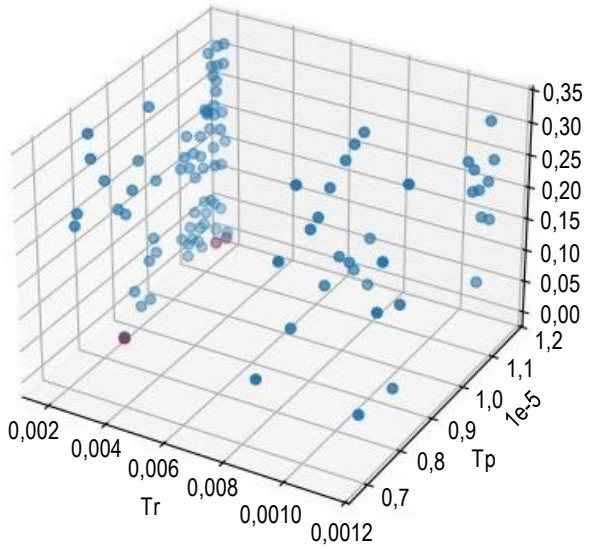


Рис. 7. Результат работы алгоритма GWO

Fig. 7. Result of GWO Algorithm Working

В ходе поиска были найдены основные три устройства, выделенные фиолетовым цветом (см. рисунок 7), а четвертое является первым устройство во множестве данных ω.

Оценка эффективности метода

Для оценки эффективности предложенного метода предлагается сравнить алгоритм GWO с алгоритмом PSO, который также является одним из эффективных с точки зрения скорости схождения и затрат вычислительных ресурсов. Ниже представлено описание данного алгоритма.

Каждая отдельная частица i состоит из трех векторов: ее положение в D-мерном пространстве поиска $\overline{x_i} = (x_{i1}, x_{i2}, \dots, x_{iD})$, лучшая найденная позиция $\overline{p_i} = (p_{i1}, p_{i2}, \dots, p_{iD})$, направленная скорость движения $\overline{v_i} = (v_{i1}, v_{i2}, \dots, v_{iD})$. При запуске алгоритма частицы равномерно, случайным образом инициализируются по всему пространству поиска, при этом скорость частиц также инициализируется случайным образом. Сформированные частицы перемещаются по пространству поиска с помощью довольно простого набора уравнений обновления векторов частицы. Алгоритм обновляет весь рой на каждом временно́м шаге, обновляя скорость и положение всякой частицы в каждом измерении по следующим правилам:

$$v_{id} = v_{id} + c\varepsilon_1(p_{id} - x_{id}) + c\varepsilon_2(p_{gd} - x_{id}), \quad (15)$$

$$v_{id} = x_{id} + v_{id}, \quad (16)$$

где c – константа со значением 2; 0, 1 и 2 – независимые случайные числа, уникально генерируемые при каждом обновлении для всякого отдельного измерения от $d = 1$, до D; p_{gd} – положение, найденное любой соседней частицей.

Процесс обновления кратко описан в алгоритме PSO, представленном в виде псевдокода:

1.for каждого шага *t* **do;**
2. **for** каждой частицы *i* в рое **do**
3. обновить позицию *xt*, используя выражения (15) и (16)
4. рассчитать фитнес-функцию для *xt f* (xt)
5. обновить *pi, pg*
6. **end for;**

7.end for.

Стоит отметить, что в алгоритме скорость частиц фиксируется на максимальном значении *vmax*. Без фиксации алгоритм склонен не сойтись, когда расчет значений (15) и (16) приводил бы к быстрому увеличению скорости и, следовательно, положения частиц, приближались бы к бесконечности. Параметр *vmax* не позволяет системе войти в данное состояние, ограничивая скорость всех частиц.

Для оценки эффективности алгоритмов была проведена серия экспериментов, где отслеживалось время схождения алгоритмов, то есть время поиска устройств для последующей миграции группы типовых контейнеров. Функция счета времени была реализована в разработанном программном коде самой модели на языке Python. Результаты сравнения представлены на рисунке 8.

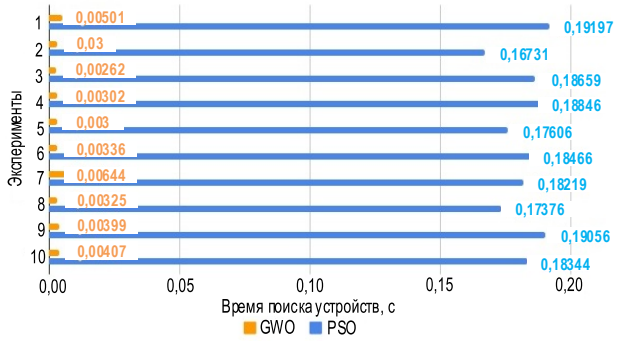


Рис. 8. Сравнение алгоритмов PSO и GWO

Fig. 8. PSO and GWO Algorithms Comparing

Среднее время в течение 10-ти экспериментов по алгоритму PSO составило 0,18249 с, в то время, как по алгоритму GWO – 0,00377 с, что меньше примерно в 48 раз. Также стоит учесть, что PSO в результате работы выдавал значение глобального экстремума целевой функции, то есть находил одно устройство, наиболее подходящее для миграции контейнера, в отличие от GWO, который позволял найти группу устройств $(\alpha, \beta, \delta, \omega)$, соответственно. Стоит отметить, что данное сравнение алгоритмов актуально в рамках исследуемой задачи. Как ранее было приведено в теоретической части, метаэвристические алгоритмы обладают своими особенностями ввиду собственной «проекции» с биологического мира, примерами которого они были вдохновлены. Соответственно, PSO может также достаточно эффективным решением в рамках других условий задачи.

Выводы

Исследования и разработки в области распределенных вычислений занимают немалую часть работ в современных и перспективных сетях и услугах. Одной из ожидаемых инфраструктурных технологий являются Fog, в частности, динамические. Данная технология позволит снизить нагрузку на ядро сети, замыкая пользовательский трафик, а также – приблизиться к достижению цели снижения энергозатрат вычислительной инфраструктуры в условиях бурного роста ЦОДов. В качестве практических исследований, в статье приводится разработанная модель и метод поиска группы Fog-устройств для последующей миграции типовых контейнеров одной из платформ FaaS. В качестве математической базы предлагается для решения сложной оптимизационной задачи использовать метаэвристические алгоритмы. В работе приводится практическое сравнение работы двух распространенных алгоритмов: GWO и PSO. В результате моделирования на основе экспериментальных данных эффективное решение поставленной задачи было достигнуто на основе алгоритма GWO.

Список источников

1. Market Overview // Straits research. URL: https://straitsresearch.com/report/data-center-equipment-market (дата обращения 31.05.2024)
2. Колбанёв М.О., Палкин И.И., Пойманова Е.Д., Татарникова Т.М. Пути создания зеленых информационных технологий // Гидрометеорология и экология. 2021. № 62. С. 127–138. DOI:10.33933/2074-2762-2021-62-127-138. EDN:OEJEMQ
3. Manner J. Black software – the energy unsustainability of software systems in the 21st century // Oxford Open Energy. 2023. Vol. 2. DOI:10.1093/ooenergy/oiac011
4. Alloghani M.A. Architecting Green Artificial Intelligence Products: Recommendations for Sustainable AI Software Development and Evaluation // Artificial Intelligence and Sustainability. Signals and Communication. Cham: Springer, 2024. PP. 65–86. DOI:10.1007/978-3-031-45214-7_4
5. Schwartz R., Dodge J., Smith N.A., Etzioni O. Green AI // Communications of the ACM. 2020. Vol. 63. Iss. 12. PP. 54–63. DOI:10.1145/3381831
6. Li Y., Zhu Z., Guan Y., Kang Y. Research on the structural features and influence mechanism of the green ICT transnational cooperation network // Economic Analysis and Policy. 2022. Vol. 75. PP. 734–749. DOI:10.1016/j.eap.2022.07.003
7. Кричевский Г.Е. Экология и «Зеленые технологии». Как сдержать превращение биосферы в техносферу? // НБИКС – Наука. Технологии. 2019. Т. 3. № 8. С. 22–26.
8. Волков А.Н. Туманность в перспективных сетях связи для услуг телеприсутствия // Электросвязь. 2024. № 4. С. 50–56.
9. Волков А.Н. Стабильность кластера в динамических туманных вычислениях // Электросвязь. 2024. № 6. С. 8–16.
10. Марочкина А.В. Моделирование и кластеризация трехмерной сети интернета вещей с применением метода оценки фрактальной размерности // Электросвязь. 2023. № 6. С. 60–66. DOI:10.34832/ELSV.2023.43.6.008. EDN:ZBNQKI
11. Марочкина А.В. Выбор головных узлов кластеров в трехмерных сетях Интернета вещей высокой плотности // Электросвязь. 2023. № 7. С. 26–32. DOI:10.34832/ELSV.2023.44.7.004. EDN:MKMNQZ

References

1. *Straits research*. Market Overview. URL: https://straitsresearch.com/report/data-center-equipment-market (дата обращения 31.05.2024)
2. Kolbanev M.O., Palkin I.I., Poymanova E.D., Tatarnikova T.M. The Challenges of the Digital Economy. *Hydrometeorology and Ecology*. 2021;62:127–138. (in Russ.) DOI:10.33933/2074-2762-2021-62-127-138. EDN:OEJEMQ

3. Manner J. Black software – the energy unsustainability of software systems in the 21st century. *Oxford Open Energy*. 2023;2. DOI:10.1093/ooenergy/oiac011
4. Alloghani M.A. Architecting Green Artificial Intelligence Products: Recommendations for Sustainable AI Software Development and Evaluation. *In: Artificial Intelligence and Sustainability. Signals and Communication*. Cham: Springer; 2024. p.65–86. DOI:10.1007/978-3-031-45214-7_4
5. Schwartz R., Dodge J., Smith N.A., Etzioni O. Green AI. *Communications of the ACM*. 2020;63(12):54–63. DOI:10.1145/3381831
6. Li Y., Zhu Z., Guan Y., Kang Y. Research on the structural features and influence mechanism of the green ICT transnational cooperation network. *Economic Analysis and Policy*. 2022;75:734–749. DOI:10.1016/j.eap.2022.07.003
7. Krichevsky G.E. Ecology and Green Technologies. How to contain the transformation of the biosphere into the technosphere? *NBIKS – Nauka. Tekhnologii*. 2019;3(8):22–26. (in Russ.)
8. Volkov A.N. Nebula in promising communication networks for telepresence services. *Electrosvyaz*. 2024;4:50–56. (in Russ.)
9. Volkov A.N. Cluster stability in dynamic fog computing. *Elektrosvyaz*. 2024;6:8–16. (in Russ.)
10. Marochkina A.V. Modeling and clustering of a three-dimensional Internet of things network using the fractal dimension estimation method. *Elektrosvyaz*. 2023;6:60–66. (in Russ.) DOI:10.34832/ELSV.2023.43.6.008. EDN:ZBNQKI
11. Marochkina, A.V. Selection of cluster head nodes in high-density three-dimensional Internet of Things networks. *Electrosvyaz*. 2023;7:26–32. (in Russ.) DOI:10.34832/ELSV.2023.44.7.004. EDN:MKMNQZ

Статья поступила в редакцию 02.06.2024; одобрена после рецензирования 25.06.2024; принята к публикации 28.06.2024.
The article was submitted 02.06.2024; approved after reviewing 25.06.2024; accepted for publication 28.06.2024.

Информация об авторе:

ВОЛКОВ Артем Николаевич
кандидат технических наук, доцент кафедры сетей связи и передачи данных Санкт-Петербургского государственного университета телекоммуникаций им. проф. М.А. Бонч-Бруевича
https://orcid.org/0009-0002-4296-1822

Автор сообщает об отсутствии конфликтов интересов.
The author declares no conflicts of interests.