ПРИМЕНЕНИЕ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА В ЗАДАЧАХ РАСПРЕДЕЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ СЕТИ

Амирсаидов Улугбек Бабурович
Профессор кафедры «С и СПД» ТУИТ
Тел: +998 98 301 48 64
E-mail:amirsadov.ulugbek@gmail.com

Аннотация. В данной работе предлагается гибридный подход к оптимизации распределения вычислительных ресурсов в иерархической архитектуре, включающей уровни Mobile Edge Computing (MEC), Fog и Intermediate Support Server (ISS). Задача оптимизации формулируется как минимизация суммарной задержки при ограничении на общий вычислительный ресурс. Для ускорения вычислений предложено использовать нейронную сеть, обученную аппроксимировать оптимальные значения интенсивностей обслуживания. Результаты моделирования демонстрируют высокую точность предсказаний и существенное снижение вычислительной сложности по сравнению с классическими методами оптимизации.

Annotatsiya: Ushbu maqolada Mobile Edge Computing (MEC), Fog va Intermediate Support Server (ISS) qatlamlaridan iborat ierarxik arxitekturada hisoblash resurslarini taqsimlashni optimallashtirishning gibrid yondashuvi taklif etilgan. Optimallashtirish muammosi umumiy hisoblash resursini cheklab, umumiy kechikishni minimallashtirish sifatida shakllantirilgan. Hisoblashni tezlashtirish uchun optimal xizmat koʻrsatish intensivligini taxminiy aniqlashga oʻrgatilgan neyron tarmoqdan foydalanish taklif qilingan. Simulyatsiya natijalari klassik optimallashtirish usullariga nisbatan yuqori bashorat aniqligi va hisoblash murakkabligining sezilarli darajada kamayganligini koʻrsatadi.

Abstract: This paper proposes a hybrid approach to optimizing computing resource allocation in a hierarchical architecture comprising Mobile Edge Computing (MEC), Fog, and Intermediate Support Server (ISS) layers. The optimization problem is formulated as minimizing the total latency while constraining the total computing resource. To accelerate the computation, we propose using a neural network trained to approximate optimal service intensities. The simulation results demonstrate high prediction accuracy and a significant reduction in computational complexity compared to classical optimization methods.

Ключевые слова: Fog Computing, системы массового обслуживания, оптимизация, нейронные сети, 5G, задержка.

Kalit so'zlar: Tumanli hisoblash, navbat tizimlari, optimallashtirish, neyron tarmoqlari, 5G, kechikish.

Keywords: Fog Computing, queuing systems, optimization, neural networks, 5G, latency.

Введение

Стремительное развитие сетей пятого и шестого поколений (5G/6G), Интернета вещей (IoT) и массовых машинных коммуникаций (mMTC) приводит к экспоненциальному росту объема генерируемых данных и предъявляет жесткие требования к задержке, надежности и масштабируемости вычислительной инфраструктуры. В частности, сценарии URLLC требуют задержек на уровне порядка 1 мс и высокой надежности (до 99.999%), что невозможно обеспечить исключительно за счет централизованных облачных вычислений.

В ответ на эти вызовы активно развиваются иерархические вычислительные архитектуры, включающие:

- Mobile Edge Computing (MEC),
- Fog Computing,
- облачные и промежуточные серверы (ISS/Cloud).

Основная идея заключается в переносе вычислений ближе к источнику данных, что позволяет:

- снизить задержку,
- уменьшить нагрузку на магистральные сети,
- повысить контекстную осведомленность системы.

Однако такая распределенная архитектура приводит к новой фундаментальной проблеме - оптимальному распределению ограниченных вычислительных ресурсов между уровнями системы.

Методы распределение ресурсов

Современные вычислительные парадигмы edge и fog computing были впервые систематически описаны в работах Satyanarayanan и Mao et al., где подчёркивается необходимость обработки данных ближе к источнику

для снижения задержек и нагрузки на облако [Y.Mao, 2017]. Проблема распределения ресурсов в fog/edge системах рассматривается как NP-сложная задача. Обзор работ [J.Huang, 2023] показывает основные стратегии: heuristic, optimization-based и AI-based методы, а также Machine Learning и AI в fog computing. Использование ML/DL/RL стало ключевым направлением исследований. Reinforcement learning применяется для динамического task offloading в работах [S.Wang, 2022, F.Luo, 2024], где показано улучшение QoS и снижение задержки.

В данной работе предлагается гибридный подход, сочетающий: аналитическую модель задержки (Erlang-C), численную оптимизацию (SLSQP) и нейросетевую аппроксимацию оптимального решения.

Математическая модель

Система состоит из трёх уровней обработки запросов: **MEC**, **Fog** и **ICC**, каждый с количеством каналов обслуживания и интенсивностью обслуживания . Входящий поток на каждом уровне задаётся интенсивностями .

Процесса построения модели и предсказания состоит их следующих этапов:

1. **Генерация данных**: создаются случайные входные нагрузки .
2. **Оптимизация**: каждая комбинация проходит через модель системы массового обслуживания типа M/M/K, чтобы найти минимальное среднее время задержки запросов.
3. **Фильтрация устойчивости**: только устойчивые решения добавляются в обучающий набор.
4. **Нормализация**: данные масштабируются до нулевого среднего и единичного стандартного отклонения для ускорения обучения.
5. **Обучение сети**: нейронная сеть обучается предсказывать по входным .

6. **Предсказание новых значений**: для новых нагрузок сеть быстро выдает прогноз оптимальных ресурсов.

Общая задержка по всей системе определяется как сумма задержек на всех уровнях:

Цель оптимизации — найти значения , минимизирующие суммарное среднее время задержки при условии устойчивости системы

при

Пусть суммарная пропускная способность всех уровней обработки фиксирована и равна :

Это условие вводится как линейное ограничение в задаче оптимизации.

Для минимизации суммарного времени задержки используется метод **SLSQP (Sequential Least Squares Quadratic Programming).** Решение задачи возвращает оптимальные значения , минимизирующие суммарное среднее время задержки, при соблюдении ограничений устойчивости и суммарной пропускной способности.

Архитектура нейронной сети

Для предсказания оптимальных параметров обслуживания на основе входных интенсивностей была использована полностью связная (fully connected) нейронная сеть.

Сеть состоит из трёх слоёв:

1. Входной слой: Этот слой передаёт данные в первый скрытый слой размерностью 12.

2. Скрытые слои: Два скрытых слоя по 12 нейронов каждый с функцией активации ReLU, позволяющая моделировать нелинейные зависимости между входными интенсивностями и оптимальными .

3. Выходной слой предсказывает оптимальные значения :

.

В результате создаётся нейронная сеть, способная принимать на вход значения интенсивностей и выдавать прогноз оптимального распределения ресурсов () с минимальной ошибкой.

В численных экспериментах для сравнения используются следующие методы распределение ресурсов: равномерное распределение (Equal Allocation), пропорциональное распределение (Load-Proportional) и обучение с подкреплением (Reinforcement Learning, RL).

Результаты экспериментов приведены таблице.

Таблица

Сравнение методов распределение ресурсов

Метод распределение ресурсов	Задержка	Отклонение от оптимума (%)	Время (мс)	Обучение
SLSQP (оптимум)	0.0125	0	45.2	нет
Нейросеть (предложенный)	0.0129	3.2	0.08	быстрое
RL	0.0146	16.8	0.12	длительное
Пропорциональный	0.0158	26	0.03	нет
Равномерный	0.0196	56	0.01	нет

Нейронная сеть обучается воспроизводить оптимальное распределение ресурсов, что позволяет заменить дорогостоящую оптимизацию быстрым предсказанием.

Заключение

Предложенный нейросетевой метод превосходит RL по точности и скорости обучения, сохраняя при этом возможность работы в реальном времени. RL требует существенно большего времени обучения по сравнению с нейросетевой аппроксимацией.

Несмотря на преимущества предложенного подхода, RL может быть более эффективен в следующих сценариях: динамически изменяющаяся среда, неполная или неизвестная модель системы и необходимость долгосрочной оптимизации.

Список использованной литературы

1. Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief (2017) “A survey on mobile edge computing: The communication perspective,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, DOI: 10.1109/COMST.2017.2745201.
2. J. Huang, X. Wang, Y. Zhang (2023). “Resource allocation in fog computing: A survey,” *Sensors*, vol. 23, no. 2, pp. 1–35, 2023. DOI: 10.3390/s23020621.
3. S. Wang, J. Zhang (2022). “Reinforcement learning for fog resource allocation,” *arXiv preprint arXiv:2203.04567*.
4. F. Luo, M. Zhang (2024). “Deep reinforcement learning for edge computing,” *IEEE Communications Surveys & Tutorials*, vol. 26, no. 1, pp. 560–590, DOI: 10.1109/COMST.2023.3298876.