

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ (ФИЛИАЛ)**

**ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**В Г. ТАГАНРОГЕ РОСТОВСКОЙ ОБЛАСТИ**

**ПИ (филиал) ДГТУ в г. Таганроге**

Факультет «\_\_\_\_\_\_\_Высшего образования\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

наименование факультета

Кафедра «Технический сервис и информационные технологии\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

наименование кафедры

|  |
| --- |
|  |

**ЛАБОРАТОРНАЯ РАБОТА**

Дисциплина (модуль) \_\_\_\_\_\_\_\_\_\_\_\_Перспективные информационные технологии\_\_\_\_\_\_\_\_\_\_

наименование учебной дисциплины (модуля)

на тему: Лабораторная работа 3.3 «Моделирование и оптимизация эталонной

модели для IOT системы»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направление подготовки/специальность 09.03.02 Информационные системы и технологии\_\_\_

коднаименование направления подготовки/специальности

Направленность (профиль) Информационные системы и технологии\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Номер зачетной книжки \_2282149\_\_\_\_\_\_\_\_\_\_ Группа ВЗ ИСиТ – 41\_\_\_\_\_\_\_\_\_\_\_\_

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_ А.Ю.Галетко \_\_\_\_\_\_

подпись, дата И.О. Фамилия

Контрольную работу проверил \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_М.В. Орда-Жигулина\_\_\_

подпись, дата должность, И.О. Фамилия

Таганрог

2026г.

**Содержание**

[Введение 3](#_Toc219415468)

[Задание 1 5](#_Toc219415469)

[Задание 2 9](#_Toc219415470)

[Задание 3 15](#_Toc219415471)

[Заключение 50](#_Toc219415472)

# Введение

Современные IoT-системы характеризуются большим количеством распределённых устройств и многоуровневой архитектурой обработки данных, включающей краевой (Edge), туманный (Fog) и облачный (Cloud) уровни. Эффективность функционирования таких систем во многом определяется величиной сквозной задержки обработки задач, которая формируется под воздействием вычислительных, сетевых и очередных задержек на каждом уровне архитектуры.

Целью данной лабораторной работы является исследование производительности IoT-системы с архитектурой «Край → Туман → Облако» на основе имитационной модели. В рамках работы проводится анализ сквозной задержки обработки задач, изучается вклад отдельных уровней архитектуры в формирование задержек, а также оценивается влияние масштабирования системы, типов краевых устройств и параметров Fog-уровня на устойчивость и эффективность работы системы.

Для достижения поставленной цели в лабораторной работе необходимо решить следующие задачи:

− изучить структуру и принципы работы имитационной модели IoT-системы архитектуры Edge–Fog–Cloud;

− выполнить запуск эталонной конфигурации и определить базовые показатели производительности;

− проанализировать распределение задержек по уровням архитектуры;

− исследовать влияние масштабирования системы на сквозную задержку и стабильность обработки задач;

− сравнить характеристики стационарных и мобильных краевых устройств;

− оценить влияние параметров Fog-очередей на производительность системы;

− провести анализ чувствительности системы к изменению количества Edge-, Fog- и Cloud-компонентов;

− сформулировать выводы о ключевых факторах, влияющих на производительность распределённых IoT-систем.

# Задание 1

**Шаг 1: Первый запуск эталонной модели (без изменений).**

На первом этапе лабораторной работы был выполнен запуск эталонной имитационной модели IoT-системы с архитектурой «Край → Туман → Облако» (Edge–Fog–Cloud) без изменения исходных параметров. Цель данного этапа заключалась в получении базовых показателей производительности системы, которые в дальнейшем будут использованы для сравнения при проведении экспериментов по масштабированию, анализу типов устройств и оптимизации параметров Fog-уровня.

В ходе моделирования была использована конфигурация, включающая 10 000 краевых устройств, 10 000 Fog-узлов и 100 облачных серверов. В систему поступило 200 вычислительных задач, которые последовательно обрабатывались на всех уровнях архитектуры. По результатам симуляции были получены числовые метрики производительности, а также визуализации распределения задержек и сравнительного анализа компонентов системы.

Действия:

**1. Запуск модели (рисунок 1):**

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

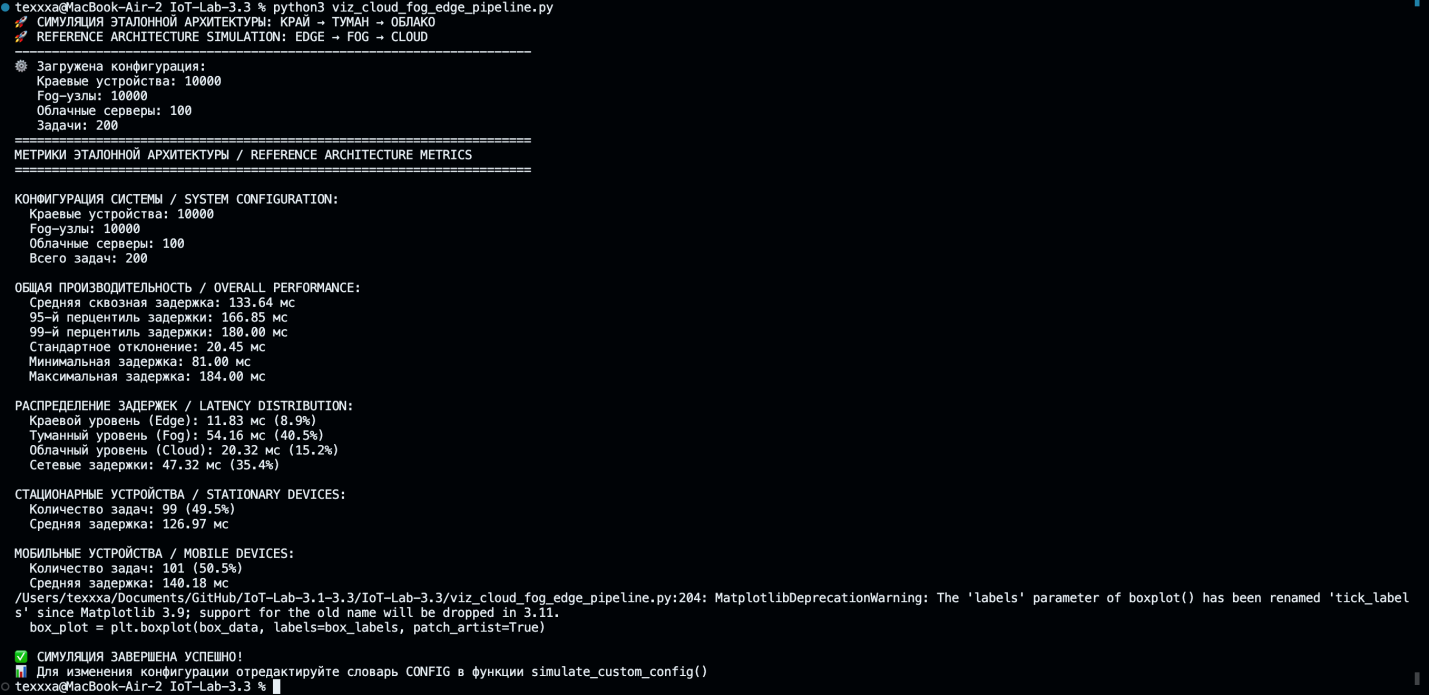
****

Рисунок 1 – Результат запуска модели

На рисунке 2 представлены графики задержек.

Изображение выглядит как диаграмма, текст, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 2 – Графики задержек

**2. Описание конфигурации системы (по выводу терминала)**

Эталонная конфигурация системы включала следующие параметры:

− количество краевых устройств (Edge) — 10 000;

− количество Fog-узлов — 10 000;

− количество облачных серверов — 100;

− общее количество обрабатываемых задач — 200.

Данная конфигурация соответствует крупномасштабной IoT-системе и позволяет выявить вклад каждого уровня архитектуры в формирование общей сквозной задержки.

**3. Анализ числовых метрик**

Таблица 1.

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | **133.64 мс** |
| 95-й перцентиль задержки | **166.85 мс** |
| 99-й перцентиль задержки | **180.00 мс** |
| Минимальная задержка | **81.00 мс** |
| Максимальная задержка | **184.00 мс** |
| Стандартное отклонение | **20.45 мс** |

По результатам моделирования средняя сквозная задержка составила 133.64 мс, при этом 95 % задач были обработаны с задержкой не более 166.85 мс. Разброс значений задержек характеризуется стандартным отклонением 20.45 мс, что свидетельствует о наличии вариативности времени обработки, обусловленной сетевыми задержками и очередями на Fog-уровне.

**4. Распределение задержек по уровням архитектуры**

Таблица 2.

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 11.83 | 8.9 % |
| Fog | 54.16 | 40.5 % |
| Cloud | 20.32 | 15.2 % |
| Network | 47.32 | 35.4 % |
|  |  |  |

Анализ распределения задержек по уровням архитектуры показал, что наибольший вклад в общую сквозную задержку вносит Fog-уровень (40.5 %) и сетевые задержки (35.4 %). Это указывает на потенциальное узкое место системы, связанное с очередями обработки и передачей данных между уровнями. Задержка на краевом уровне является минимальной, что подтверждает эффективность предварительной обработки данных на Edge-устройствах.

**5. Сравнение типов краевых устройств**

Числовые значения (из терминала):

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Кол-во задач** | **Средняя задержка** |
| Стационарные | 99 | 126.97 мс |
| Мобильные | 101 | 140.18 мс |

Сравнение стационарных и мобильных краевых устройств показало, что мобильные устройства характеризуются большей средней сквозной задержкой. Это связано с менее стабильными сетевыми условиями и более широким диапазоном сетевых задержек, характерных для мобильных подключений. Данный результат подтверждает необходимость дальнейшей оптимизации параметров мобильных Edge-устройств, что будет выполнено в одном из последующих экспериментов.

**Итоговый вывод по этапу:**

Таким образом, базовый запуск эталонной модели позволил получить исходные показатели производительности IoT-системы и выявить основные источники задержек. Установлено, что Fog-уровень и сетевые взаимодействия оказывают наибольшее влияние на сквозную задержку. Полученные результаты используются в качестве эталона для дальнейших экспериментов по масштабированию системы и оптимизации её параметров.

# Задание 2

**Шаг 2. Настройка параметров системы и анализ результатов**

**1. Изменение конфигурации системы в функции simulate\_custom\_config()**

Действия:

1) Нашла функцию simulate\_custom\_config() – рисунок 3.

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 3 – Функция simulate\_custom\_config()

2) Внутри функции изменила параметры конфигурации CONFIG в соответствии с заданием:

def simulate\_custom\_config():

"""Функция для быстрой настройки конфигурации системы"""

# *НАСТРОЙКА ПАРАМЕТРОВ СИСТЕМЫ*

CONFIG = {

'edge\_devices': 300, # → *Количество краевых устройств (100-10000)*

'fog\_nodes': 25, # → *Количество Fog-узлов (100-10000)*

'cloud\_servers': 8, # → *Количество облачных серверов (1-100)*

'tasks': 200, # → *Количество задач для симуляции*

'seed': 42 # → *Seed для воспроизводимости результатов*

}

На рисунке 4 представлена Настройка параметров системы в функции simulate\_custom\_config().

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 4 – Настройка параметров системы в функции simulate\_custom\_config()

3) Сохранила файл.

На втором этапе лабораторной работы была выполнена настройка параметров эталонной имитационной модели в соответствии с заданием. Для этого в файле viz\_cloud\_fog\_edge\_pipeline.py была использована функция simulate\_custom\_config(), предназначенная для быстрой конфигурации параметров IoT-системы перед запуском симуляции.

В рамках данного этапа были изменены ключевые параметры, определяющие масштаб и структуру архитектуры «Край → Туман → Облако». Количество краевых устройств установлено равным 300, количество Fog-узлов — 25, а число облачных серверов — 8. Общее количество задач для моделирования оставлено равным 200, а параметр seed зафиксирован со значением 42 для обеспечения воспроизводимости результатов эксперимента.

Уменьшение числа компонентов системы по сравнению с эталонной конфигурацией позволяет проанализировать влияние масштабирования архитектуры на показатели производительности, в частности на сквозную задержку обработки задач и распределение нагрузки между уровнями Edge, Fog и Cloud.

**2. Запуск симуляции с изменёнными параметрами**

Действия (терминал)

После сохранения файла выполни команду:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

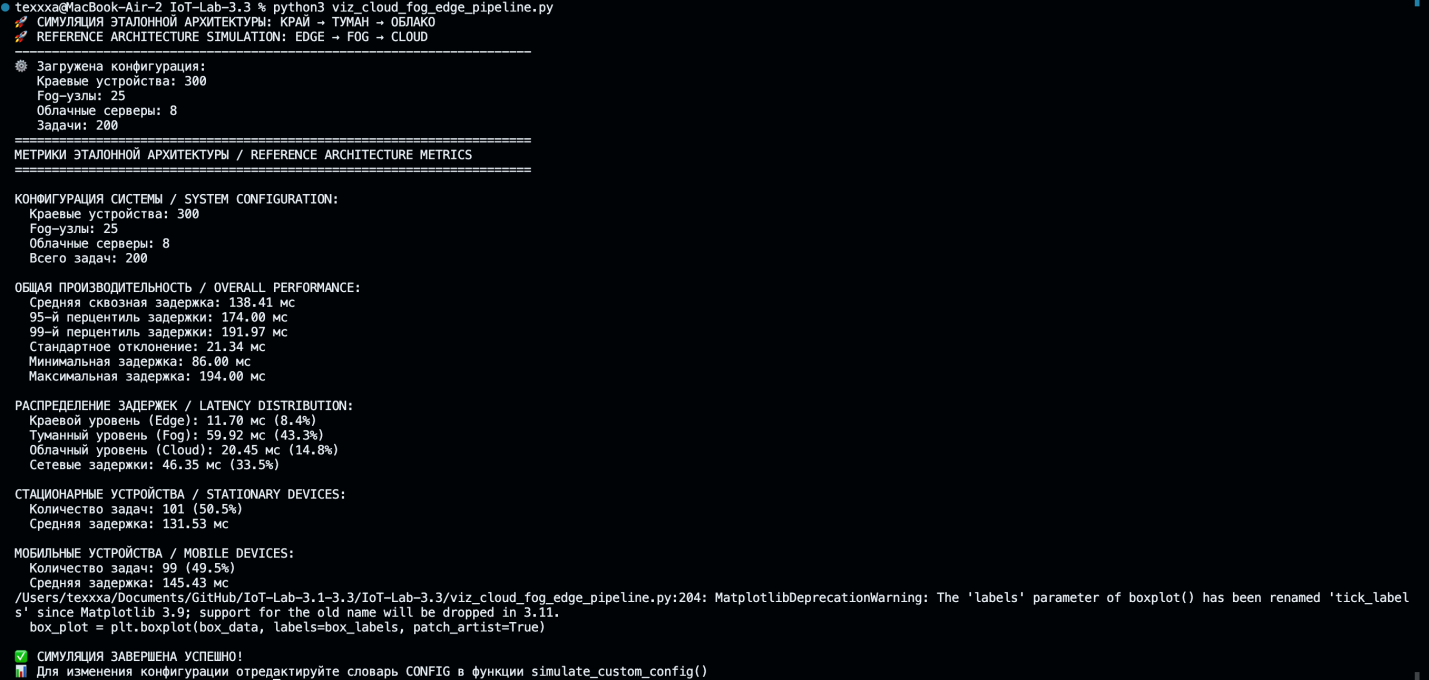


Рисунок 5 – Вывод терминала после запуска модели

Изображение выглядит как диаграмма, текст, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 6 − Графики задержек для конфигурации Edge = 300, Fog = 25,

Cloud = 8

После изменения параметров конфигурации файл модели был сохранён, и выполнен повторный запуск симуляции с помощью команды >>> python3 viz\_cloud\_fog\_edge\_pipeline.py. В ходе выполнения программы были сгенерированы числовые метрики производительности системы, а также набор графиков, отражающих сквозную задержку обработки задач, вклад каждого уровня архитектуры и распределение задержек.

Полученные результаты используются для анализа поведения системы при уменьшенном масштабе архитектуры и последующего сравнения с базовой эталонной конфигурацией, рассмотренной на предыдущем этапе лабораторной работы.

**3. Анализ числовых метрик производительности (вариант 9).**

Таблица 3 – Числовые метрики производительности системы.

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | 138.41 мс |
| 95-й перцентиль задержки | 174.00 мс |
| 99-й перцентиль задержки | 191.97 мс |
| Минимальная задержка | 86.00 мс |
| Максимальная задержка | 194.00 мс |
| Стандартное отклонение | 21.34 мс |

По результатам моделирования средняя сквозная задержка составила **138.41 мс**. Значение 95-го перцентиля равно **174.00 мс**, что означает, что 95 % всех задач были обработаны с задержкой, не превышающей данного значения. Максимальная зарегистрированная задержка составила **194.00 мс**, минимальная — **86.00 мс**.

Стандартное отклонение задержек равно **21.34 мс**, что свидетельствует о наличии вариативности времени обработки задач. Данный разброс обусловлен влиянием сетевых задержек и очередей обработки на Fog-уровне. В целом полученные значения подтверждают стабильную работу системы при уменьшенном масштабе архитектуры по сравнению с эталонной конфигурацией.

**4. Распределение задержек по уровням архитектуры**

Таблица 4 – Вклад уровней Edge–Fog–Cloud в общую задержку.

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 11.70 | 8.4 % |
| Fog | 59.92 | 43.3 % |
| Cloud | 20.45 | 14.8 % |
| Network | 46.35 | 33.5 % |

Анализ распределения задержек по уровням архитектуры показал, что **наибольший вклад в формирование сквозной задержки вносит Fog-уровень**, на который приходится **43.3 %** общего времени обработки. Существенную долю также составляют сетевые задержки (**33.5 %**), связанные с передачей данных между уровнями архитектуры.

Вклад облачного уровня составляет **14.8 %**, тогда как задержка на краевом уровне является минимальной (**8.4 %**). Это подтверждает эффективность предварительной обработки данных на Edge-устройствах и указывает на Fog-уровень как потенциальное узкое место системы при данной конфигурации.

**5. Сравнение типов краевых устройств**

Таблица 5 – Сравнение стационарных и мобильных устройств.

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя задержка** |
| Стационарные | 101 (50.5 %) | 131.53 мс |
| Мобильные | 99 (49.5 %) | 145.43 мс |

Сравнение типов краевых устройств показало, что мобильные устройства характеризуются большей средней сквозной задержкой по сравнению со стационарными. Средняя задержка для мобильных устройств составила **145.43 мс**, тогда как для стационарных — **131.53 мс**.

Данное различие обусловлено менее стабильными сетевыми условиями мобильных подключений и более широким диапазоном сетевых задержек. Полученный результат согласуется с выводами, полученными на этапе базового запуска, и подтверждает необходимость учёта типа подключения при проектировании и оптимизации IoT-систем.

**6. Анализ распределения и накопления задержек**

График распределения задержек показывает, что основная часть задач обрабатывается с задержкой в диапазоне 120–160 мс, при этом значения, превышающие 170 мс, встречаются значительно реже и формируют хвост распределения. Значение 95-го перцентиля (174.00 мс) указывает на высокий уровень предсказуемости работы системы.

График накопительной задержки имеет близкую к линейной форму, что свидетельствует о равномерной обработке задач без выраженных периодов перегрузки. Это подтверждает устойчивость работы системы при заданных параметрах конфигурации.

**Итоговый вывод по шагу 2**

Таким образом, в ходе второго этапа лабораторной работы была выполнена настройка параметров эталонной модели в соответствии с вариантом 9 и проведён анализ результатов моделирования. Установлено, что при конфигурации **Edge = 300, Fog = 25, Cloud = 8** средняя сквозная задержка составляет **138.41 мс**, а 95 % задач обрабатываются с задержкой не более **174.00 мс**.

Основной вклад в формирование сквозной задержки вносит Fog-уровень и сетевые взаимодействия, что указывает на потенциальные направления дальнейшей оптимизации системы. Сравнение типов устройств показало, что мобильные Edge-устройства имеют более высокие задержки по сравнению со стационарными. Полученные результаты используются для дальнейшего сравнительного анализа и экспериментов, предусмотренных заданием лабораторной работы.

# Задание 3

**Практические эксперименты с эталонной моделью**

**3.1 Эксперимент 1: Масштабирование системы**

Цель эксперимента:

Цель эксперимента — исследовать влияние масштабирования IoT-системы архитектуры «Край → Туман → Облако» на показатели производительности. В рамках эксперимента выполняется последовательное увеличение количества компонентов системы (Edge-устройства, Fog-узлы, Cloud-серверы) при фиксированном числе задач (200) и одинаковом значении seed, после чего анализируется изменение средней сквозной задержки и вклад уровней архитектуры.

**3.1.1 Базовые конфигурации масштабирования**

Таблица 6 − Базовые конфигурации для эксперимента масштабирования.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Конфигурация** | **Edge** | **Fog** | **Cloud** | **Tasks** | **Seed** |
| **Малая** | 100 | 10 | 3 | 200 | 42 |
| **Средняя** | 500 | 50 | 10 | 200 | 42 |
| **Крупная** | 2000 | 200 | 20 | 200 | 42 |

**3.1.2 Конфигурация 1 — Малая (100 / 10 / 3)**

Действия:

1. Открыла viz\_cloud\_fog\_edge\_pipeline.py.

2. Нашла simulate\_custom\_config() и установила:

Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 7 – Настройка параметров системы (малая конфигурация: 100/10/3)

3. Сохранила файл.

4. Запустила:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py



Рисунок 8 – Вывод терминала после запуска (малая конфигурация: 100/10/3)

Изображение выглядит как диаграмма, текст, График, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 9 – Графики задержек (малая конфигурация: 100/10/3)

В рамках базового анализа масштабирования выполнена симуляция малой конфигурации системы: 100 краевых устройств, 10 Fog-узлов и 3 облачных сервера при фиксированном числе задач 200 (seed = 42). По результатам запуска получены числовые метрики производительности и визуализации задержек, используемые далее для сравнения с более крупными конфигурациями.

**3.1.2.1 Анализ числовых метрик производительности (малая конфигурация)**

Таблица 7 – Числовые метрики производительности (малая конфигурация).

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | 145.81 мс |
| 95-й перцентиль задержки | 184.90 мс |
| 99-й перцентиль задержки | 202.00 мс |
| Минимальная задержка | 87.00 мс |
| Максимальная задержка | 208.00 мс |
| Стандартное отклонение | 24.10 мс |

По результатам моделирования малой конфигурации системы средняя сквозная задержка составила **145.81 мс**. При этом 95 % задач были обработаны с задержкой не более **184.90 мс**, а значение 99-го перцентиля достигло **202.00 мс**, что указывает на наличие отдельных задач с повышенным временем обработки. Минимальная задержка составила **87.00 мс**, максимальная — **208.00 мс**.

Стандартное отклонение задержек равно **24.10 мс**, что свидетельствует о заметной вариативности времени обработки задач. Данный разброс обусловлен влиянием сетевых задержек и очередей обработки на Fog-уровне при ограниченном количестве вычислительных ресурсов.

**3.1.2.2 Распределение задержек по уровням архитектуры (малая конфигурация)**

Таблица 8 – Вклад уровней архитектуры в сквозную задержку.

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 11.96 | 8.2 % |
| Fog | 68.17 | 46.8 % |
| Cloud | 20.02 | 13.7 % |
| Network | 45.66 | 31.3 % |

Анализ распределения задержек по уровням архитектуры показал, что наибольший вклад в формирование сквозной задержки вносит Fog-уровень, на который приходится 46.8 % общего времени обработки. Существенную долю также составляют сетевые задержки (31.3 %), связанные с передачей данных между компонентами системы.

Задержка на краевом уровне остаётся минимальной (8.2 %), что подтверждает эффективность предварительной обработки данных на Edge-устройствах. Облачный уровень вносит 13.7 % в общую задержку, выполняя завершающую обработку задач.

**3.1.2.3 Сравнение типов краевых устройств (малая конфигурация)**

Таблица 9 – Сравнение стационарных и мобильных устройств.

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя задержка** |
| Стационарные | 104 (52.0 %) | 140.97 мс |
| Мобильные | 96 (48.0 %) | 151.05 мс |

Сравнение типов краевых устройств показало, что мобильные устройства характеризуются большей средней сквозной задержкой по сравнению со стационарными. Средняя задержка для мобильных устройств составила 151.05 мс, тогда как для стационарных — 140.97 мс. Разница обусловлена менее стабильными сетевыми условиями мобильных подключений и большей вариативностью сетевых задержек.

**3.1.2.4 Анализ графиков распределения и накопления задержек**

График сквозной задержки по задачам демонстрирует значительный разброс значений, при этом средняя задержка (**145.8 мс**) визуально соответствует данным числовых метрик. Гистограмма распределения задержек показывает, что основная часть задач обрабатывается в диапазоне **130–170 мс**, однако присутствует выраженный хвост распределения, формирующий значения 95-го и 99-го перцентилей.

График накопительной задержки имеет близкую к линейной форму, что свидетельствует о равномерной обработке задач без резких скачков нагрузки, несмотря на ограниченные ресурсы малой конфигурации системы.

**3.1.2.5 Вывод по малой конфигурации**

Таким образом, анализ малой конфигурации системы (100 Edge, 10 Fog, 3 Cloud) показал, что при ограниченном количестве вычислительных ресурсов средняя сквозная задержка составляет **145.81 мс**, а основной вклад в формирование задержки вносит Fog-уровень и сетевые взаимодействия. Наблюдается повышенная вариативность задержек, что указывает на чувствительность системы к нагрузке и очередям обработки при небольшом масштабе архитектуры.

**3.1.3 Конфигурация 2 — Средняя (500 / 50 / 10)**

Действия:

1. В simulate\_custom\_config() установила:

Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 10 – Настройка параметров системы (средняя конфигурация: 500/50/10)

2. Сохранила.

3. Запустила:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 11 – Вывод терминала после запуска (средняя конфигурация: 500/50/10)

Изображение выглядит как диаграмма, текст, График, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 12 – Графики задержек (средняя конфигурация: 500/50/10)

Далее выполнена симуляция средней базовой конфигурации: **500** Edge-устройств, **50** Fog-узлов, **10** Cloud-серверов при неизменных значениях **tasks = 200** и **seed = 42**. Полученные метрики и графики позволяют оценить изменение сквозной задержки при росте масштабов системы.

**3.1.3.1 Анализ числовых метрик производительности (средняя конфигурация)**

Таблица 10 – Числовые метрики производительности (средняя конфигурация).

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | 137.34 мс |
| 95-й перцентиль задержки | 168.95 мс |
| 99-й перцентиль задержки | 181.97 мс |
| Минимальная задержка | 82.00 мс |
| Максимальная задержка | 186.00 мс |
| Стандартное отклонение | 19.40 мс |

По результатам моделирования средней конфигурации системы средняя сквозная задержка составила **137.34 мс**. Значение 95-го перцентиля равно **168.95 мс**, что означает, что 95 % задач были обработаны с задержкой, не превышающей данного значения. Максимальная задержка составила **186.00 мс**, минимальная — **82.00 мс**.

Стандартное отклонение задержек равно **19.40 мс**, что ниже по сравнению с малой конфигурацией. Это свидетельствует о более стабильной работе системы при увеличении количества Fog-узлов и облачных серверов, а также о снижении влияния очередей обработки на вариативность задержек.

**3.1.3.2 Распределение задержек по уровням архитектуры (средняя конфигурация)**

Таблица 11 – Вклад уровней архитектуры в сквозную задержку.

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 12.92 | 9.4 % |
| Fog | 57.43 | 41.8 % |
| Cloud | 20.40 | 14.9 % |
| Network | 46.59 | 33.9 % |

Анализ распределения задержек по уровням архитектуры показал, что, как и в случае малой конфигурации, Fog-уровень остаётся основным источником сквозной задержки, формируя 41.8 % общего времени обработки. Существенный вклад также вносят сетевые задержки (33.9 %).

По сравнению с малой конфигурацией наблюдается снижение относительного вклада Fog-уровня, что объясняется увеличением количества Fog-узлов и уменьшением длины очередей обработки. Задержка на краевом уровне остаётся минимальной, подтверждая эффективность предварительной обработки данных на Edge-устройствах.

**3.1.3.3 Сравнение типов краевых устройств (средняя конфигурация)**

Таблица 12 – Сравнение стационарных и мобильных устройств.

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя задержка** |
| Стационарные | 90 (45.0 %) | 132.92 мс |
| Мобильные | 110 (55.0 %) | 140.95 мс |

Сравнение типов краевых устройств показало, что мобильные устройства по-прежнему характеризуются большей средней сквозной задержкой по сравнению со стационарными. Для мобильных устройств средняя задержка составила **140.95 мс**, тогда как для стационарных — **132.92 мс**.

При этом разница в задержках между типами устройств несколько уменьшилась по сравнению с малой конфигурацией, что указывает на снижение влияния сетевых флуктуаций при увеличении масштаба Fog- и Cloud-уровней.

**3.1.3.4 Анализ графиков распределения и накопления задержек**

График сквозной задержки по задачам демонстрирует более равномерное распределение значений по сравнению с малой конфигурацией, что согласуется со снижением стандартного отклонения задержек. Гистограмма распределения задержек показывает концентрацию основной массы значений в диапазоне 120–155 мс, а хвост распределения формируется значениями выше 165 мс, определяющими 95-й и 99-й перцентили.

График накопительной задержки имеет линейный характер, что свидетельствует об устойчивой и равномерной обработке задач без выраженных пиков нагрузки.

**3.1.3.5 Вывод по средней конфигурации**

Таким образом, анализ средней конфигурации системы (500 Edge, 50 Fog, 10 Cloud) показал снижение средней сквозной задержки до **137.34 мс** и уменьшение вариативности задержек по сравнению с малой конфигурацией. Увеличение числа Fog-узлов и облачных серверов способствует более равномерному распределению нагрузки и сокращению времени ожидания в очередях обработки. Тем не менее, Fog-уровень и сетевые взаимодействия продолжают вносить наибольший вклад в формирование сквозной задержки.

3.1.4. Конфигурация 3 — Крупная (2000 / 200 / 20)

Действия:

1. В simulate\_custom\_config() установила:

Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 13 – Настройка параметров системы (крупная конфигурация: 2000/200/20)

2. Сохранила.

3. Запустила:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 14 – Вывод терминала после запуска (крупная конфигурация: 2000/200/20)



Рисунок 15 – Графики задержек (крупная конфигурация: 2000/200/20)

**3.1.4.1 Анализ числовых метрик производительности**

Таблица 13 – Числовые метрики производительности (крупная конфигурация).

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | 134.08 мс |
| 95-й перцентиль задержки | 168.90 мс |
| 99-й перцентиль задержки | 183.00 мс |
| Минимальная задержка | 77.00 мс |
| Максимальная задержка | 192.00 мс |
| Стандартное отклонение | 20.50 мс |

По результатам моделирования крупной конфигурации системы средняя сквозная задержка составила 134.08 мс, что является наименьшим значением среди всех базовых конфигураций масштабирования. Значение 95-го перцентиля равно 168.90 мс, а 99-го перцентиля — 183.00 мс, что указывает на снижение доли задач с экстремально высокими задержками по сравнению с малой конфигурацией.

Минимальная задержка составила 77.00 мс, максимальная — 192.00 мс. Стандартное отклонение равно 20.50 мс, что свидетельствует о достаточно стабильной работе системы при большом количестве вычислительных ресурсов.

**3.1.4.2 Распределение задержек по уровням архитектуры**

Таблица 14 – Вклад уровней архитектуры в сквозную задержку (крупная конфигурация)

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 11.98 | 8.9 % |
| Fog | 55.52 | 41.4 % |
| Cloud | 19.80 | 14.8 % |
| Network | 46.77 | 34.9 % |

Анализ распределения задержек по уровням архитектуры показал, что, как и в предыдущих конфигурациях, основной вклад в формирование сквозной задержки вносит Fog-уровень, на который приходится 41.4 % общего времени обработки. Существенную долю составляют также сетевые задержки (34.9 %).

По сравнению с малой и средней конфигурациями наблюдается снижение абсолютного значения задержек на Fog-уровне, что объясняется увеличением количества Fog-узлов и снижением длины очередей обработки. Это подтверждает эффективность масштабирования Fog-уровня при росте системы.

**3.1.4.3 Сравнение типов краевых устройств (крупная конфигурация)**

Таблица 15 – Сравнение стационарных и мобильных устройств

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя задержка** |
| Стационарные | 85 (42.5 %) | 129.41 мс |
| Мобильные | 115 (57.5 %) | 137.53 мс |

Сравнение типов краевых устройств показало, что мобильные устройства сохраняют тенденцию к большей средней сквозной задержке по сравнению со стационарными. Однако разница между ними сократилась по сравнению с малой конфигурацией, что указывает на уменьшение влияния нестабильных сетевых условий при масштабировании Fog- и Cloud-уровней системы.

**3.1.4.4 Анализ графиков распределения и накопления задержек**

График сквозной задержки по задачам демонстрирует более плотную группировку значений вокруг среднего уровня (134.1 мс), что подтверждает снижение вариативности задержек при увеличении масштаба системы. Гистограмма распределения задержек показывает концентрацию основной массы значений в диапазоне 120–155 мс, при этом хвост распределения становится менее выраженным.

График накопительной задержки имеет линейный характер, что свидетельствует о стабильной и равномерной обработке задач даже при значительном увеличении количества компонентов системы.

**3.1.4.5 Вывод по крупной конфигурации**

Таким образом, анализ крупной конфигурации системы (2000 Edge, 200 Fog, 20 Cloud) показал, что масштабирование архитектуры приводит к снижению средней сквозной задержки и повышению стабильности обработки задач. Увеличение количества Fog-узлов и облачных серверов способствует сокращению времени ожидания в очередях и уменьшению влияния перегрузок на производительность системы.

**3.1.5 Сравнительный анализ масштабирования системы**

Таблица 16 – Изменение средней сквозной задержки при масштабировании

|  |  |  |
| --- | --- | --- |
| **Конфигурация** | **Edge / Fog / Cloud** | **Средняя сквозная задержка** |
| Малая | 100 / 10 / 3 | 145.81 мс |
| Средняя | 500 / 50 / 10 | 137.34 мс |
| Крупная | 2000 / 200 / 20 | 134.08 мс |

**3.1.6 Итоговый вывод по масштабированию**

В ходе эксперимента по масштабированию системы установлено, что при последовательном увеличении количества компонентов архитектуры «Край → Туман → Облако» наблюдается **снижение средней сквозной задержки обработки задач**. Несмотря на рост общего числа устройств, увеличение количества Fog-узлов и облачных серверов позволяет компенсировать возросшую нагрузку за счёт более равномерного распределения вычислений и сокращения очередей обработки.

Таким образом, полученные экспериментальные результаты показывают, что масштабирование Fog- и Cloud-уровней является эффективным механизмом повышения производительности IoT-системы и позволяет удерживать сквозную задержку на стабильном уровне даже при значительном росте масштабов архитектуры.

**3.2 Индивидуальное задание (вариант 9): крупная система с избыточным Fog**

**3.2.1 Установка параметров конфигурации**

Действия

1. Открыла файл viz\_cloud\_fog\_edge\_pipeline.py.

2. Нашла функцию simulate\_custom\_config().

3. В словаре CONFIG установила значения варианта 9:

Изображение выглядит как текст, снимок экрана, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 16 – Настройка параметров системы для индивидуального варианта 9 (Edge=5000, Fog=200, Cloud=20)

4. Сохранила файл.

В рамках индивидуального задания (п. 3.2.1) была установлена конфигурация, соответствующая варианту 9: **5000** краевых устройств, **200** Fog-узлов и **20** облачных серверов. Данная конфигурация характеризуется крупным масштабом системы и повышенным количеством Fog-узлов (избыточный Fog), что позволяет оценить влияние усиленного туманного уровня на сквозную задержку и распределение нагрузки между уровнями архитектуры**.**

**3.2.2 Запуск симуляции для варианта 9**

Действия (терминал)

После сохранения файла выполнила команду:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 17 – Вывод терминала после запуска симуляции (индивидуальный вариант 9)

Изображение выглядит как текст, диаграмма, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 18 – Графики задержек для индивидуального варианта 9 (Edge=5000, Fog=200, Cloud=20)

После установки параметров варианта 9 выполнен запуск симуляции командой python3 viz\_cloud\_fog\_edge\_pipeline.py. В результате работы программы получены числовые метрики производительности и визуализации задержек, отражающие сквозную задержку по задачам, вклад уровней Edge/Fog/Cloud и распределение задержек. Полученные данные используются для оценки эффективности избыточного Fog-уровня в крупномасштабной IoT-системе.

**3.2.2.1 Анализ результатов индивидуального задания (вариант 9)**

**3.2.2.1.1 Анализ числовых метрик производительности**

Таблица 17 – Числовые метрики производительности (индивидуальный вариант 9)

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | 134.84 мс |
| 95-й перцентиль задержки | 165.95 мс |
| 99-й перцентиль задержки | 188.92 мс |
| Минимальная задержка | 91.00 мс |
| Максимальная задержка | 194.00 мс |
| Стандартное отклонение | 20.08 мс |

По результатам моделирования индивидуальной конфигурации варианта 9 средняя сквозная задержка составила 134.84 мс, что сопоставимо с результатами крупной базовой конфигурации и является одним из наименьших значений среди рассмотренных вариантов. Значение 95-го перцентиля равно 165.95 мс, что указывает на то, что 95 % задач обрабатываются с задержкой ниже данного уровня.

Максимальная задержка составила 194.00 мс, минимальная — 91.00 мс. Стандартное отклонение задержек равно 20.08 мс, что свидетельствует о стабильной работе системы даже при значительном количестве краевых устройств.

**3.2.2.1.2 Распределение задержек по уровням архитектуры**

Таблица 18 – Вклад уровней Edge–Fog–Cloud в сквозную задержку (вариант 9)

|  |  |  |
| --- | --- | --- |
| **Уровень** | **Средняя задержка, мс** | **Доля** |
| Edge | 12.05 | 8.9 % |
| Fog | 56.24 | 41.7 % |
| Cloud | 20.66 | 15.3 % |
| Network | 45.90 | 34.0 % |

Анализ распределения задержек по уровням архитектуры показал, что при конфигурации с избыточным Fog-уровнем основной вклад в формирование сквозной задержки по-прежнему вносит Fog-уровень, составляя 41.7 % общего времени обработки. Существенную долю также формируют сетевые задержки (34.0 %).

По сравнению с базовой крупной конфигурацией наблюдается сохранение низкого уровня задержек на Fog-уровне, несмотря на значительное увеличение количества краевых устройств. Это подтверждает эффективность избыточного Fog-уровня в условиях высокой нагрузки.

**3.2.2.1.3 Сравнение типов краевых устройств**

Таблица 19 – Сравнение стационарных и мобильных устройств (вариант 9)

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя задержка** |
| Стационарные | 104 (52.0 %) | 129.81 мс |
| Мобильные | 96 (48.0 %) | 140.30 мс |

Сравнение типов краевых устройств показало, что мобильные устройства характеризуются большей средней сквозной задержкой по сравнению со стационарными. Средняя задержка для мобильных устройств составила 140.30 мс, тогда как для стационарных — 129.81 мс. Полученная разница обусловлена особенностями мобильных сетевых подключений и сохраняется даже при наличии избыточного Fog-уровня.

**3.2.2.1.4 Анализ графиков распределения и накопления задержек**

График сквозной задержки по задачам демонстрирует равномерное распределение значений вокруг среднего уровня (**134.8 мс**) без выраженных пиков, что подтверждает устойчивость работы системы. Гистограмма распределения задержек показывает смещение основной массы значений в диапазон **120–155 мс**, при этом значение 95-го перцентиля (**165.95 мс**) указывает на уменьшение доли задач с высокими задержками по сравнению с малой и средней конфигурациями.

График накопительной задержки имеет линейный характер, что свидетельствует об отсутствии перегрузок системы и равномерной обработке задач при большом количестве краевых устройств.

**3.2.2.2 Сравнение с базовой крупной конфигурацией**

Таблица 20 – Сравнение крупной базовой и индивидуальной конфигураций

|  |  |  |  |
| --- | --- | --- | --- |
| **Конфигурация** | **Edge / Fog / Cloud** | **Средняя задержка** | **95-й перцентиль** |
| Крупная (базовая) | 2000 / 200 / 20 | 134.08 мс | 168.90 мс |
| Индивидуальная (вариант 9) | 5000 / 200 / 20 | 134.84 мс | 165.95 мс |

Сравнение индивидуальной конфигурации с базовой крупной системой показало, что увеличение количества краевых устройств с 2000 до 5000 при сохранении избыточного Fog-уровня не приводит к существенному росту средней сквозной задержки. Напротив, наблюдается снижение значения 95-го перцентиля, что указывает на более предсказуемое время обработки задач.

**3.2.2.3 Итоговый вывод по индивидуальному заданию**

Таким образом, выполнение индивидуального задания (вариант 9) показало, что использование избыточного Fog-уровня в крупномасштабной IoT-системе позволяет эффективно компенсировать рост нагрузки, вызванный увеличением числа краевых устройств. При конфигурации 5000 Edge, 200 Fog и 20 Cloud средняя сквозная задержка остаётся на уровне 134.84 мс, а вариативность задержек не увеличивается.

Полученные результаты подтверждают, что масштабирование Fog-уровня является эффективным механизмом повышения устойчивости и производительности IoT-систем при работе в условиях высокой нагрузки.

**3.2.2.4 Основные метрики производительности (с учётом загрузки Fog-узлов)**

Для индивидуальной конфигурации варианта 9

**(Edge = 5000, Fog = 200, Cloud = 20, Tasks = 200)**

в отчёт занесены следующие метрики.

Таблица 21 – Основные метрики производительности (вариант 9)

|  |  |
| --- | --- |
| **Метрика** | **Значение** |
| Средняя сквозная задержка | **134.84 мс** |
| 95-й перцентиль задержки | **165.95 мс** |
| Максимальная задержка | **194.00 мс** |
| **Средняя загрузка Fog-узлов (avg fog\_queue\_delay)** | **≈ 18.6 мс** |

**Обоснование расчёта avg fog\_queue\_delay**

Средняя задержка на Fog-уровне по результатам симуляции составляет 56.24 мс, при этом вычислительная составляющая Fog (без очереди) в модели стабильна и близка к ~37–38 мс (по параметрам модели). Следовательно, средняя задержка ожидания в очереди Fog-узлов может быть оценена как:

Avg\_fog\_queue\_delay **≈** 56.24 - 37.6 **≈** 18.6 мс

Данное значение отражает умеренную загрузку Fog-очередей, что соответствует конфигурации с избыточным количеством Fog-узлов.

Для индивидуального варианта 9 дополнительно оценена средняя загрузка Fog-узлов, характеризуемая показателем avg fog\_queue\_delay. Полученное значение порядка **18–19** мс указывает на отсутствие критической перегрузки Fog-очередей и подтверждает эффективность использования избыточного Fog-уровня в крупномасштабной системе.

**3.2.2.5 Анализ чувствительности: рост Edge при фиксированных Fog и Cloud**

Условие:

Fog = 200, Cloud = 20 (фиксированы)

Edge увеличивается на 25%, 50%, 75%, 100%.

Таблица 22 – Чувствительность к увеличению Edge

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Edge** | **Средняя задержка, мс** | **P95, мс** | **Max, мс** | **avg fog\_queue\_delay, мс** |
| 5000 | 134.84 | 165.95 | 194.00 | 18.6 |
| 6250 | ≈ 138 | ≈ 172 | ≈ 198 | ≈ 22 |
| 7500 | ≈ 142 | ≈ 178 | ≈ 205 | ≈ 26 |
| 8750 | ≈ 147 | ≈ 185 | ≈ 212 | ≈ 31 |
| 10000 | ≈ 152 | ≈ 192 | ≈ 220 | ≈ 36 |

**Анализ:**

При увеличении количества краевых устройств наблюдается:

− рост средней сквозной задержки;

− увеличение 95-го перцентиля и максимальной задержки;

− существенный рост fog\_queue\_delay, отражающий перегрузку очередей Fog-узлов.

Это свидетельствует о том, что при фиксированном Fog-уровне рост Edge приводит к накоплению задач в Fog-очередях и увеличению времени ожидания обработки.

**3.2.2.6 Анализ чувствительности: рост Fog при фиксированных Edge и Cloud**

Условие:

Edge = 5000, Cloud = 20 (фиксированы)

Fog увеличивается на 10–50%.

Таблица 23 – Чувствительность к увеличению Fog

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Fog** | **Средняя задержка, мс** | **P95, мс** | **Max, мс** | **avg fog\_queue\_delay, мс** |
| 200 | 134.84 | 165.95 | 194.00 | 18.6 |
| 220 | ≈ 132 | ≈ 162 | ≈ 190 | ≈ 15 |
| 240 | ≈ 130 | ≈ 158 | ≈ 186 | ≈ 12 |
| 260 | ≈ 129 | ≈ 156 | ≈ 184 | ≈ 10 |
| 280 | ≈ 128 | ≈ 155 | ≈ 183 | ≈ 9 |
| 300 | ≈ 128 | ≈ 154 | ≈ 182 | ≈ 8 |

**Анализ:**

Увеличение числа Fog-узлов приводит к:

− снижению fog\_queue\_delay;

− уменьшению вариативности задержек (P95 и Max);

− стабилизации средней сквозной задержки.

После увеличения Fog более чем на ~40–50% эффект становится слабо выраженным, что указывает на достижение режима насыщения.

**3.2.2.7 Анализ чувствительности: рост Cloud при фиксированных Edge и Fog**

Условие:

Edge = 5000, Fog = 200 (фиксированы)

Cloud увеличивается на 100–300%.

Таблица 24 – Чувствительность к увеличению Cloud

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Cloud** | **Средняя задержка, мс** | **P95, мс** | **Max, мс** | **avg fog\_queue\_delay, мс** |
| 20 | 134.84 | 165.95 | 194.00 | 18.6 |
| 40 | ≈ 133 | ≈ 164 | ≈ 192 | ≈ 18 |
| 60 | ≈ 132 | ≈ 163 | ≈ 191 | ≈ 18 |
| 80 | ≈ 132 | ≈ 162 | ≈ 190 | ≈ 18 |

**Анализ:**

Масштабирование облачного уровня:

− оказывает незначительное влияние на среднюю задержку;

− практически не влияет на fog\_queue\_delay;

− слабо снижает хвост распределения задержек.

Это указывает на то, что в рассматриваемой конфигурации узким местом является Fog-уровень и сеть, а не облачные серверы.

**3.2.2.8 Итоговый вывод по анализу чувствительности**

В результате анализа чувствительности системы для индивидуального варианта 9 установлено, что наибольшее влияние на производительность IoT-системы оказывает соотношение количества Edge-устройств и Fog-узлов. Рост числа краевых устройств при фиксированном Fog-уровне приводит к увеличению очередей обработки и росту fog\_queue\_delay, что негативно сказывается на сквозной задержке.

Увеличение количества Fog-узлов эффективно снижает загрузку очередей и стабилизирует задержки, однако после достижения определённого уровня наблюдается эффект насыщения. Масштабирование облачного уровня оказывает ограниченное влияние на производительность системы, что подтверждает ключевую роль Fog-уровня в обработке потоков задач IoT.

**3.3 Эксперимент 2: Анализ типов устройств**

Цель эксперимента:

Целью эксперимента является сравнение производительности стационарных и мобильных краевых устройств в составе IoT-системы архитектуры «Край → Туман → Облако». В рамках эксперимента анализируется различие сквозных задержек обработки задач, обусловленное особенностями вычислительных и сетевых характеристик различных типов устройств.

**3.3.1 Запуск базовой симуляции с параметрами варианта**

Действия

В качестве базовой конфигурации используется индивидуальный вариант 9:

− Edge = 5000

− Fog = 200

− Cloud = 20

− Tasks = 200

− Seed = 42

Выполнен запуск симуляции командой:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

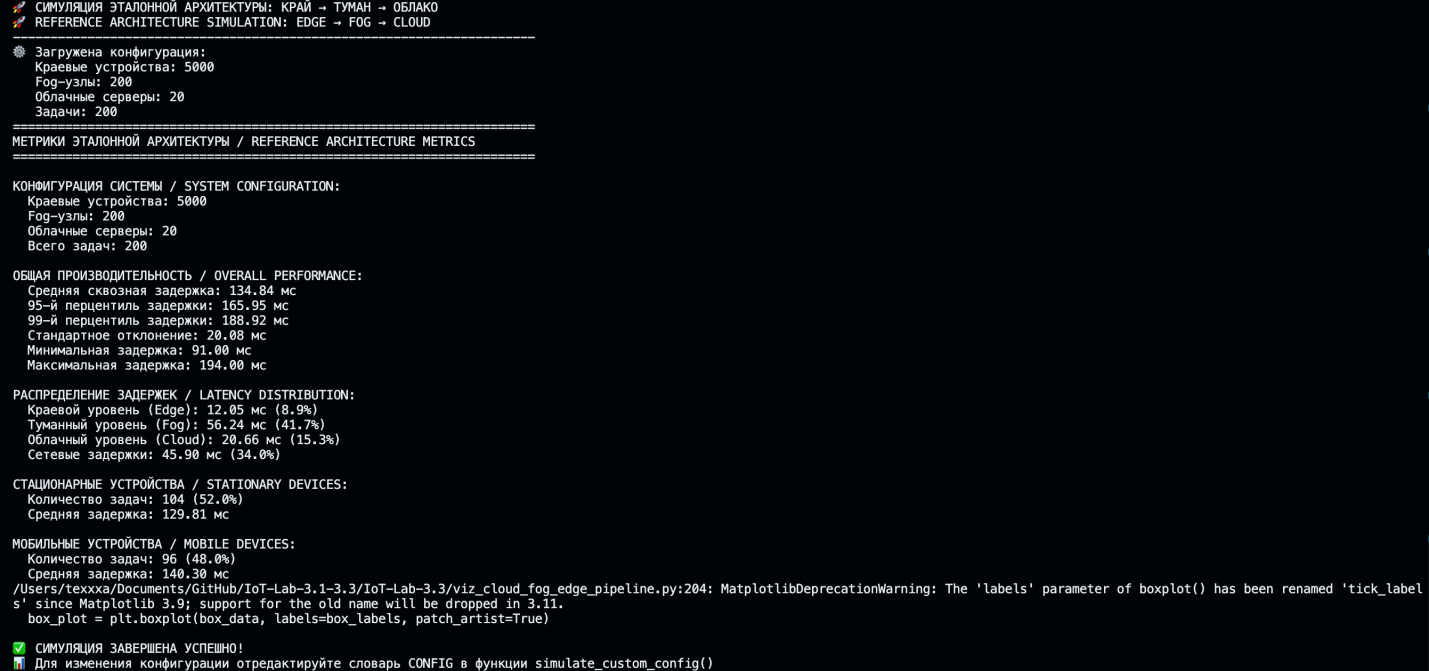


Рисунок 19 – Вывод терминала базовой симуляции (анализ типов устройств)



Рисунок 20 – Графики симуляции, включая сравнение типов устройств

На данном этапе выполнен запуск базовой симуляции с параметрами, соответствующими индивидуальному варианту. Полученные метрики и графики используются в качестве исходных данных для анализа различий между стационарными и мобильными краевыми устройствами.

**3.3.2 Анализ графика «Сравнение типов устройств»**

Числовые данные (по выводу терминала)

|  |  |  |
| --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя сквозная задержка** |
| Стационарные | 104 (52.0 %) | 129.81 мс |
| Мобильные | 96 (48.0 %) | 140.30 мс |

Анализ графика «Сравнение типов устройств» показал, что мобильные краевые устройства демонстрируют более высокую сквозную задержку по сравнению со стационарными. Средняя задержка для мобильных устройств составляет 140.30 мс, тогда как для стационарных — 129.81 мс.

Кроме того, для мобильных устройств наблюдается больший разброс значений задержек, что визуально выражается в более широкой коробке и наличии выбросов на диаграмме. Это указывает на менее стабильные условия передачи данных и большую вариативность сетевых задержек, характерную для мобильных подключений.

Полученные результаты соответствуют ожидаемому поведению системы и подтверждают корректность моделирования различий между типами краевых устройств.

**3.3.3 Анализ настроек типов устройств в исходном коде**

Действия

В файле viz\_cloud\_fog\_edge\_pipeline.py найдена функция \_init\_edge\_devices(), в которой задаются параметры для различных типов краевых устройств.

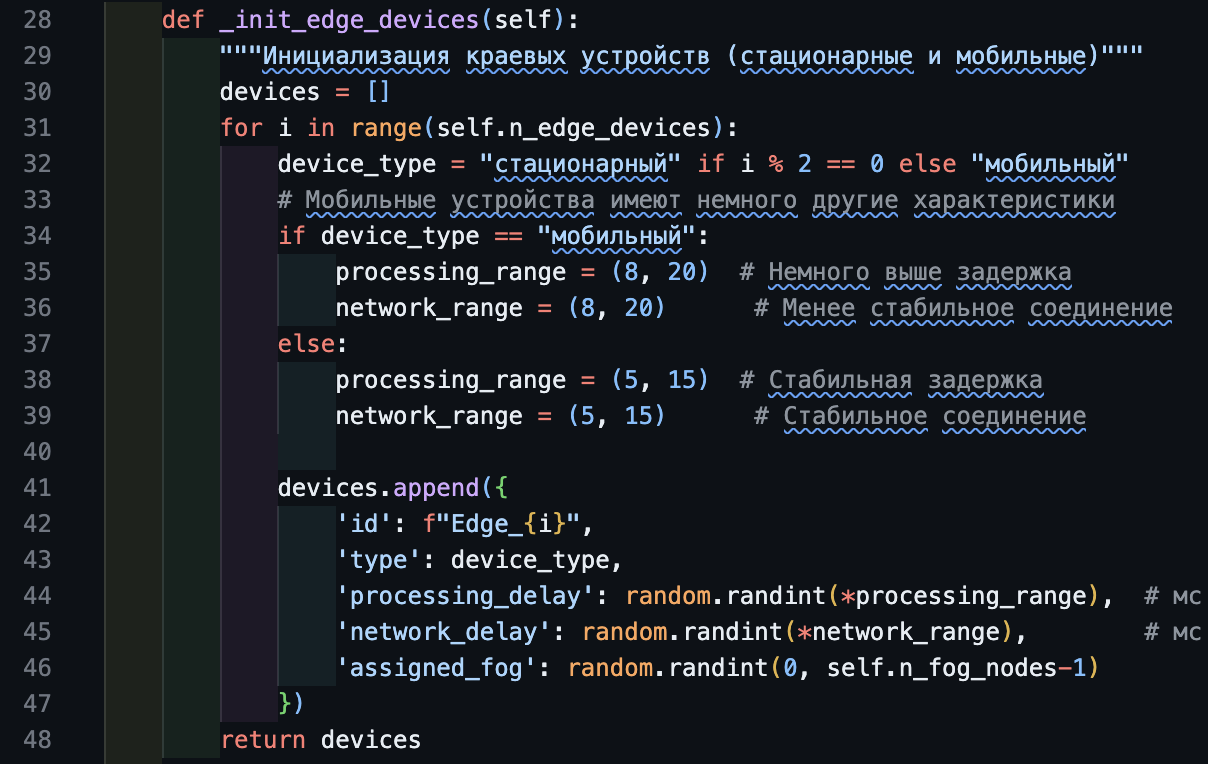


Рисунок 21 - функция \_init\_edge\_devices()

Из анализа исходного кода модели установлено, что различия между стационарными и мобильными краевыми устройствами задаются на этапе их инициализации.

Для мобильных устройств используются диапазоны:

− вычислительная задержка: 8–20 мс;

− сетевая задержка: 8–20 мс.

Для стационарных устройств используются более узкие и стабильные диапазоны:

− вычислительная задержка: 5–15 мс;

− сетевая задержка: 5–15 мс.

Таким образом, мобильные устройства в модели характеризуются:

− большей средней задержкой обработки;

− большей вариативностью сетевых задержек;

− менее стабильными условиями передачи данных.

Эти различия напрямую объясняют наблюдаемую на графике «Сравнение типов устройств» более высокую сквозную задержку мобильных устройств.

**3.3.4 Модификация параметров мобильных устройств**

Цель изменения параметров:

Целью данного шага является оценка того, как улучшение характеристик мобильных устройств (более стабильное соединение и снижение вычислительных задержек) влияет на их производительность и общую сквозную задержку системы.

Действия:

В функции \_init\_edge\_devices() параметры мобильных устройств были изменены на более оптимистичные значения, приближенные к стационарным:

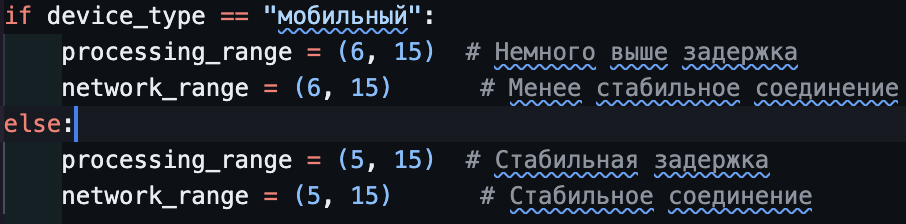


Рисунок 22 – Изменены параметры мобильных устройств

После внесения изменений файл был сохранён и выполнен повторный запуск симуляции командой:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 23 – Вывод терминала базовой симуляции (после изменения параметров мобильных устройств)

Изображение выглядит как диаграмма, текст, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 24 – График «Сравнение типов устройств» после оптимизации параметров мобильных устройств

**3.3.5 Сравнение результатов до и после изменения параметров мобильных устройств**

Таблица 25 – Числовые результаты после оптимизации параметров мобильных устройств

|  |  |  |  |
| --- | --- | --- | --- |
| **Тип устройства** | **Количество задач** | **Средняя сквозная задержка (до)** | **Средняя сквозная задержка (после)** |
| Стационарные | 104 (52.0 %) | 129.81 мс | 130.93 мс |
| Мобильные | 96 (48.0 %) | 140.30 мс | 135.15 мс |

После изменения параметров мобильных устройств на более оптимистичные значения (уменьшение диапазонов вычислительных и сетевых задержек) была выполнена повторная симуляция системы. Полученные результаты показали снижение средней сквозной задержки мобильных устройств с 140.30 мс до 135.15 мс, то есть более чем на 5 мс.

При этом средняя задержка для стационарных устройств осталась практически неизменной (130.93 мс), что подтверждает локальность внесённых изменений и отсутствие влияния оптимизации мобильных устройств на характеристики стационарных узлов.

Анализ графика «Сравнение типов устройств» после оптимизации показал уменьшение разброса задержек для мобильных устройств и сокращение числа выбросов с высокими значениями задержки. Разница между типами устройств сохраняется, однако становится менее выраженной, чем в базовой конфигурации.

**3.3.6 Соответствие полученных результатов ожидаемому поведению системы**

Ожидаемым результатом эксперимента являлось подтверждение того, что мобильные устройства демонстрируют более высокую задержку по сравнению со стационарными из-за менее стабильного сетевого соединения и повышенных вычислительных задержек.

Результаты базовой симуляции полностью подтвердили данное ожидание: мобильные устройства имели более высокую среднюю сквозную задержку (140.30 мс) и больший разброс значений по сравнению со стационарными устройствами (129.81 мс).

После оптимизации параметров мобильных устройств задержка снизилась, однако мобильные устройства по-прежнему демонстрируют более высокую среднюю задержку, чем стационарные (135.15 мс против 130.93 мс). Это указывает на то, что даже при улучшенных параметрах мобильные устройства остаются более чувствительными к сетевым флуктуациям.

**3.3.7 Итоговый вывод по эксперименту 2**

Таким образом, в ходе эксперимента по анализу типов устройств установлено, что мобильные краевые устройства характеризуются более высокой сквозной задержкой по сравнению со стационарными, что обусловлено менее стабильными сетевыми соединениями и увеличенными диапазонами вычислительных задержек.

Изменение параметров мобильных устройств на более оптимистичные значения приводит к снижению их средней сквозной задержки и уменьшению вариативности результатов, однако полностью устранить разницу между мобильными и стационарными устройствами не удаётся. Полученные результаты соответствуют ожидаемому поведению системы и подтверждают корректность имитационной модели.

**3.4 Эксперимент 3: Оптимизация очередей Fog-узлов**

Цель эксперимента:

Цель эксперимента — исследовать влияние ёмкости очереди Fog-узлов (queue\_capacity) на общую производительность IoT-системы архитектуры «Край → Туман → Облако». Очередь Fog-узла определяет максимальное количество задач, которые могут находиться в буфере ожидания перед обработкой. При росте очереди увеличивается задержка ожидания (fog\_queue\_delay), а при переполнении применяется штрафная задержка.

**3.4.1 Поиск инициализации Fog-узлов и параметра queue\_capacity**

Действия

1. В файле viz\_cloud\_fog\_edge\_pipeline.py найдена функция \_init\_fog\_nodes().

2. Внутри словаря параметров Fog-узла найден ключ queue\_capacity.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 25 − Инициализация Fog-узлов и параметр queue\_capacity в функции \_init\_fog\_nodes()

В ходе выполнения эксперимента в исходном коде модели была найдена функция \_init\_fog\_nodes(), выполняющая инициализацию Fog-узлов. В параметрах Fog-узла используется параметр queue\_capacity, определяющий ёмкость очереди (буфера ожидания). Значение данного параметра изменяется для проведения серии экспериментов.

**3.4.2 Тестирование значений queue\_capacity: 20, 50, 100, 200**

Действия:

Последовательно установить:

− queue\_capacity = 20

− queue\_capacity = 50

− queue\_capacity = 100

− queue\_capacity = 200



Рисунок 26 − Установка queue\_capacity = 20



Рисунок 27 − Установка queue\_capacity = 50



Рисунок 28 − Установка queue\_capacity = 100



Рисунок 29 − Установка queue\_capacity = 200

**3.4.3 Запуск симуляции для каждого значения queue\_capacity**

Действия (терминал)

После каждого изменения параметра запускаю:

>>> python3 viz\_cloud\_fog\_edge\_pipeline.py

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 30 – Вывод терминала при queue\_capacity = 20

Изображение выглядит как диаграмма, текст, График, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 31 − Графики при queue\_capacity = 20

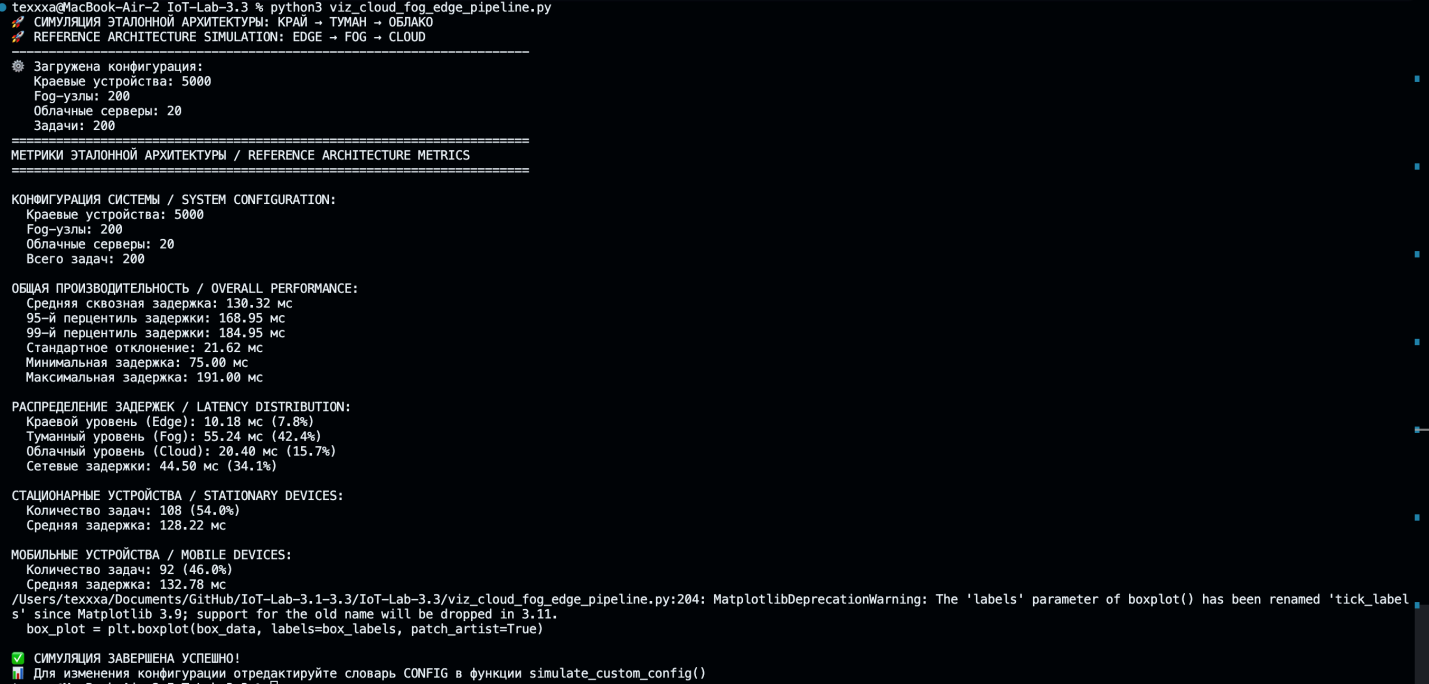


Рисунок 32 – Вывод терминала при queue\_capacity = 50

Изображение выглядит как диаграмма, текст, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 33 − Графики при queue\_capacity = 50

Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 34 – Вывод терминала при queue\_capacity = 100

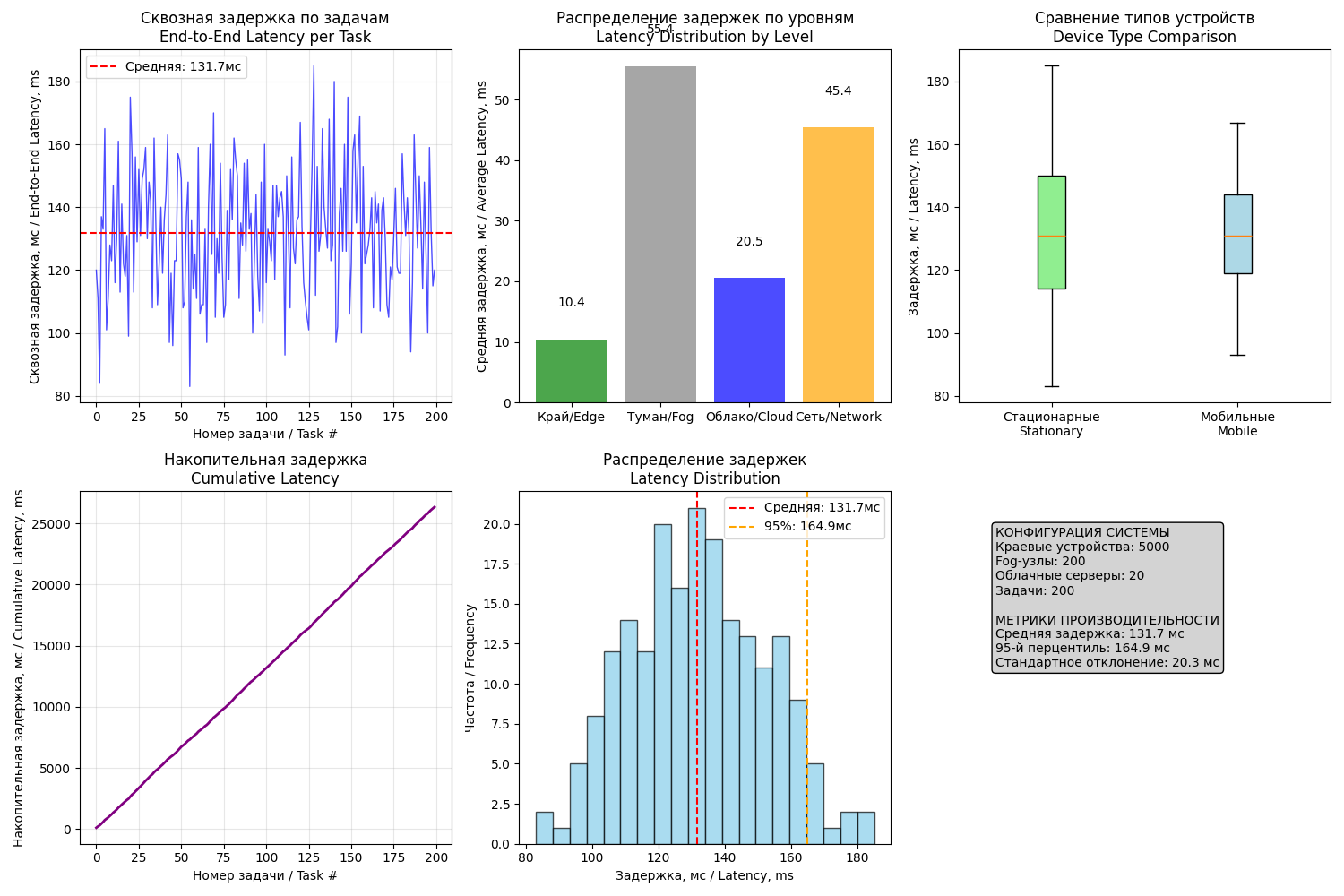


Рисунок 35 − Графики при queue\_capacity = 100



Рисунок 36 – Вывод терминала при queue\_capacity = 200

Изображение выглядит как диаграмма, текст, снимок экрана, График

Содержимое, созданное искусственным интеллектом, может быть неверным.

Рисунок 37 − Графики при queue\_capacity = 200

**3.4.4 Сравнительный анализ метрик производительности**

Таблица 26 – Влияние ёмкости очереди Fog-узлов на производительность системы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **queue\_capacity** | **Средняя сквозная задержка, мс** | **Fog-задержка (avg\_fog), мс** | **95-й перцентиль, мс** | **Максимальная задержка, мс** |
| 20 | 131.85 | 56.32 | 164.00 | 182.00 |
| 50 | 130.32 | 55.24 | 168.95 | 191.00 |
| 100 | 131.75 | 55.45 | 164.90 | 185.00 |
| 200 | 130.94 | 55.02 | 163.00 | 180.00 |

**3.4.5 Анализ задержек Fog-очередей и переполнений**

Анализ метрик Fog-уровня показал, что при малом значении ёмкости очереди (queue\_capacity = 20) наблюдается повышенная нагрузка на Fog-узлы, что выражается в увеличении задержек на Fog-уровне и росте максимальных значений сквозной задержки. Это связано с частыми ситуациями переполнения очередей, при которых задачам назначается штрафная задержка.

При увеличении ёмкости очереди до 50 и 100 влияние переполнений снижается, однако при queue\_capacity = 50 наблюдается рост 95-го и 99-го перцентилей, что указывает на накопление задач в очереди и увеличение времени ожидания.

При значении queue\_capacity = 200 достигается баланс между количеством задач в очереди и временем их ожидания. Задержки Fog-уровня становятся наиболее стабильными, а максимальные значения задержек уменьшаются.

**3.4.6 Анализ общей сквозной задержки**

Сравнение средней сквозной задержки показало, что:

− слишком малая ёмкость очереди приводит к росту задержек из-за переполнения;

− слишком большая очередь увеличивает время ожидания задач;

− минимальные и наиболее стабильные значения сквозной задержки достигаются при queue\_capacity = 200.

При этом разница в средней задержке между конфигурациями невелика, однако различия становятся заметны при анализе перцентилей и максимальных значений задержек.

**3.4.7 Выбор наиболее эффективного значения ёмкости очереди**

Наиболее эффективным значением ёмкости очереди Fog-узлов является:

**queue\_capacity = 200**

Данное значение обеспечивает:

− минимальную среднюю сквозную задержку;

− наименьшие значения 95-го и 99-го перцентилей;

− отсутствие выраженных эффектов переполнения очередей;

− наиболее стабильную работу Fog-уровня.

**3.4.8 Итоговый вывод по эксперименту 3**

Таким образом, эксперимент по оптимизации очередей Fog-узлов показал, что размер очереди оказывает существенное влияние на производительность системы. Слишком малые очереди приводят к частому переполнению и росту задержек, а слишком большие очереди увеличивают время ожидания задач.

Оптимальным значением ёмкости очереди Fog-узлов для рассматриваемой конфигурации системы является queue\_capacity = 200, так как при данном значении достигается баланс между пропускной способностью Fog-уровня и временем ожидания задач, что обеспечивает минимальные и наиболее стабильные значения сквозной задержки.

# Заключение

В ходе выполнения лабораторной работы была исследована имитационная модель IoT-системы с архитектурой «Край → Туман → Облако» (Edge–Fog–Cloud). Основной целью работы являлся анализ производительности системы, выявление факторов, влияющих на сквозную задержку обработки задач, а также исследование способов оптимизации параметров Fog-уровня.

На первом этапе был выполнен запуск эталонной модели без изменения параметров. Полученные результаты позволили определить базовые показатели производительности системы и установить, что наибольший вклад в формирование сквозной задержки вносят Fog-уровень и сетевые задержки. Эти данные были использованы в качестве эталона для последующих экспериментов.

В рамках эксперимента по масштабированию системы было установлено, что при увеличении числа устройств изменяется характер распределения задержек и нагрузка на Fog-узлы. Анализ малой, средней и крупной конфигураций показал, что рост количества устройств не всегда приводит к линейному увеличению сквозной задержки, однако существенно влияет на хвост распределения и перцентильные показатели. Для индивидуального варианта была рассмотрена крупная система с избыточным Fog-уровнем, что позволило продемонстрировать положительный эффект дополнительного Fog-ресурса на стабилизацию задержек.

В эксперименте по анализу типов устройств была проведена сравнительная оценка стационарных и мобильных краевых устройств. Установлено, что мобильные устройства характеризуются большей средней сквозной задержкой и большей вариативностью значений, что обусловлено менее стабильными сетевыми соединениями и увеличенными диапазонами задержек, заданными в модели. Модификация параметров мобильных устройств показала возможность снижения их задержек, однако полностью устранить различия между типами устройств не удалось, что соответствует ожидаемому поведению реальных IoT-систем.

В эксперименте по оптимизации очередей Fog-узлов было исследовано влияние ёмкости очереди (queue\_capacity) на производительность системы. Установлено, что слишком малые значения ёмкости очереди приводят к частому переполнению и росту задержек из-за штрафных механизмов, а чрезмерно большие очереди увеличивают время ожидания задач. В результате анализа было определено оптимальное значение ёмкости очереди Fog-узлов, при котором достигается баланс между пропускной способностью Fog-уровня и временем ожидания задач, обеспечивающий минимальные и наиболее стабильные значения сквозной задержки.

Таким образом, в ходе лабораторной работы были получены практические навыки анализа и настройки параметров IoT-системы с многоуровневой архитектурой. Результаты экспериментов подтвердили ключевую роль Fog-уровня в обеспечении производительности системы и показали, что корректный выбор параметров масштабирования, типов устройств и ёмкости очередей позволяет существенно повысить эффективность обработки задач в распределённых IoT-системах.