

## Анализ производительности облачных систем хранения данных на основе моделей массового обслуживания

*С.А. Ямашкин, И.С. Седов, К.Т. Прибылов*

*Мордовский государственный университет им. Н.П.Огарёва*

**Аннотация:** В статье рассматривается использование модели массового обслуживания М/М/п для анализа производительности облачных систем хранения данных. Проведено моделирование, позволяющее выявить влияние параметров системы на среднее время ожидания, вероятность блокировки и пропускную способность. Полученные результаты демонстрируют, как оптимизация числа серверов и интенсивности обслуживания позволяет повысить эффективность работы системы и минимизировать задержки. Актуальность исследования обусловлена необходимостью повышения производительности облачных решений в условиях роста объемов данных и увеличения нагрузки на системы хранения.

**Ключевые слова:** облачные системы хранения данных, теория массового обслуживания, модель М/М/п, Python, моделирование, анализ производительности.

### Введение

В современном мире цифровых технологий, облачные платформы для хранения информации занимают центральное место в структуре информационных ресурсов, предоставляя гибкий и адаптируемый доступ к данным для множества сфер деятельности — от коммерческого сектора до научных исследований и государственного управления. Рост объемов данных и потребность в их быстрой обработке предъявляют все более строгие требования к эффективности этих систем.

Для гарантии высокой производительности облачных хранилищ необходимы средства, которые позволяют изучать и предсказывать их функционирование под воздействием различных нагрузок [1]. Одним из наиболее эффективных методов для решения этой задачи является применение теории массового обслуживания. Эта теория предлагает математические инструменты для анализа систем, где происходят процессы приема и обработки запросов, что делает ее особенно важной для исследования облачных технологий [2-3].

Современные системы хранения данных (СХД) представляют собой сложные архитектуры, включающие элементы облачных, гибридных и распределенных решений. Развитие технологий систем хранения позволяет значительно повысить производительность и снизить задержки в обработке запросов. Некоторые модели делают эти системы гибкими и экономически эффективными для предприятий любого масштаба.

Цель данного исследования – разработка модели, основанной на теории массового обслуживания (СМО), для анализа производительности облачных систем хранения данных. В статье будут рассматриваться ключевые показатели производительности, такие как пропускная способность, время отклика и вероятность отказа.

Актуальность исследования обусловлена необходимостью повышения надёжности и эффективности облачных систем хранения данных в условиях растущей нагрузки и растущей популярности облачных технологий.

### **Теоретическая основа СМО и облачных СХД**

Облачные системы хранения данных охватывают распределённые инфраструктуры, обеспечивающие доступ к данным через сеть. Ключевые преимущества таких систем включают масштабируемость, высокую доступность и адаптивность к изменениям нагрузки. Однако с ростом объёмов данных и запросов возникают проблемы с обеспечением производительности и надёжности этих систем [4].

Для анализа производительности облачных хранилищ часто применяется теория массового обслуживания (ТМО), которая изучает потоки заявок и ресурсы для их обработки [5-6]. Модели ТМО, такие как М/М/1 и М/М/п, позволяют формализовать работу систем и анализировать параметры, такие как время ожидания, пропускная способность и вероятность отказа, что важно для прогнозирования поведения облачных хранилищ. Например, в

---

модели M/M/1 время ожидания и время пребывания в системе для одиночного канала вычисляются с помощью следующих формул:

среднее время ожидания в очереди [7]:

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)},$$

среднее время пребывания в системе (включая время обслуживания) [8]:

$$W_q = \frac{1}{\mu - \lambda},$$

пропускная способность (среднее количество обслуживаемых запросов в единицу времени) [9]:

$$X = \lambda,$$

вероятность того, что система будет пустой (то есть нет заявок на обработку) [10]:

$$P_0 = 1 - \frac{\lambda}{\mu}.$$

Для многоканальной системы M/M/n, где n — количество серверов, формулы усложняются, и для оценки производительности могут использоваться более сложные методы, такие как вычисление средней загрузки каждого канала и вероятности отказов из-за недостаточности серверов [11].

### **Анализ и применение модели в облачной СХД**

Для анализа производительности облачных систем хранения данных можно использовать модель массового обслуживания, например, модель M/M/n, где n — это количество обслуживающих каналов. Такая модель подходит для облачных хранилищ, где несколько серверов (каналов) обрабатывают поступающие запросы на чтение или запись данных [12-13].

Для расчета всех параметров смоделируем облачную систему хранения данных с несколькими серверами на Python и приведем код (табл. 1).

---

### Программный код на Python для моделирования системы

```
import numpy as np
import matplotlib.pyplot as plt

# Параметры системы
lambda_rate = 5 # интенсивность поступления запросов ( $\lambda$ )
mu_rate = 3 # интенсивность обслуживания ( $\mu$ )
n_servers = 3 # количество серверов (каналов)
n_requests = 10000 # количество запросов

arrival_times = np.cumsum(np.random.exponential(1/lambda_rate, n_requests))
service_times = np.random.exponential(1/mu_rate, n_requests)

start_times = np.zeros(n_requests)

for i in range(n_requests):
    if i < n_servers:
        start_times[i] = arrival_times[i] # Первоначально запросы обрабатываются сразу
    else:
        # Запросы начинают обслуживаться, когда хотя бы один сервер свободен
        start_times[i] = max(arrival_times[i], np.min(start_times[:i]) +
service_times[:i].min())

wait_times = start_times - arrival_times

avg_wait_time = np.mean(wait_times) # среднее время ожидания
throughput = n_requests / (arrival_times[-1] - arrival_times[0]) # пропускная
способность
blocking_prob = np.sum(wait_times > 0) / n_requests # вероятность блокировки
(ожидания)

# Вывод результатов
print(f"Среднее время ожидания: {avg_wait_time:.2f} секунд")
print(f"Пропускная способность: {throughput:.2f} запросов/сек")
print(f"Вероятность блокировки: {blocking_prob:.2f}")

plt.figure(figsize=(10, 6))

plt.hist(wait_times, bins=50, alpha=0.75, color='blue', edgecolor='black',
label="Время ожидания")

plt.axvline(avg_wait_time, color='red', linestyle='dashed', linewidth=2,
label=f"Среднее время ожидания: {avg_wait_time:.2f} секунд")

plt.title(f"Распределение времени ожидания запросов\n $\lambda = \{lambda\_rate\}$ ,  $\mu =$ 
 $\{mu\_rate\}$ ,  $n = \{n\_servers\}$ ", fontsize=14)
plt.xlabel("Время ожидания (сек)", fontsize=12)
plt.ylabel("Частота", fontsize=12)

plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```

Примером может служить облачная система хранения данных, в которой запросы поступают от клиентов, а данные обслуживаются несколькими серверами. Учитываем, что запросы на обслуживание могут поступать с определённой интенсивностью, а время обслуживания зависит от размера данных и скорости обработки. Каждый запрос может быть на чтение или запись данных, и каждый сервер имеет ограниченную пропускную способность. Анализ производительности системы поможет определить, сколько запросов система способна обслужить за определённое время, какое время ожидания испытывают запросы в очереди, и как это время меняется при увеличении количества серверов [14].

Исходя из наших данных вывод результатов показан на рис. 1.

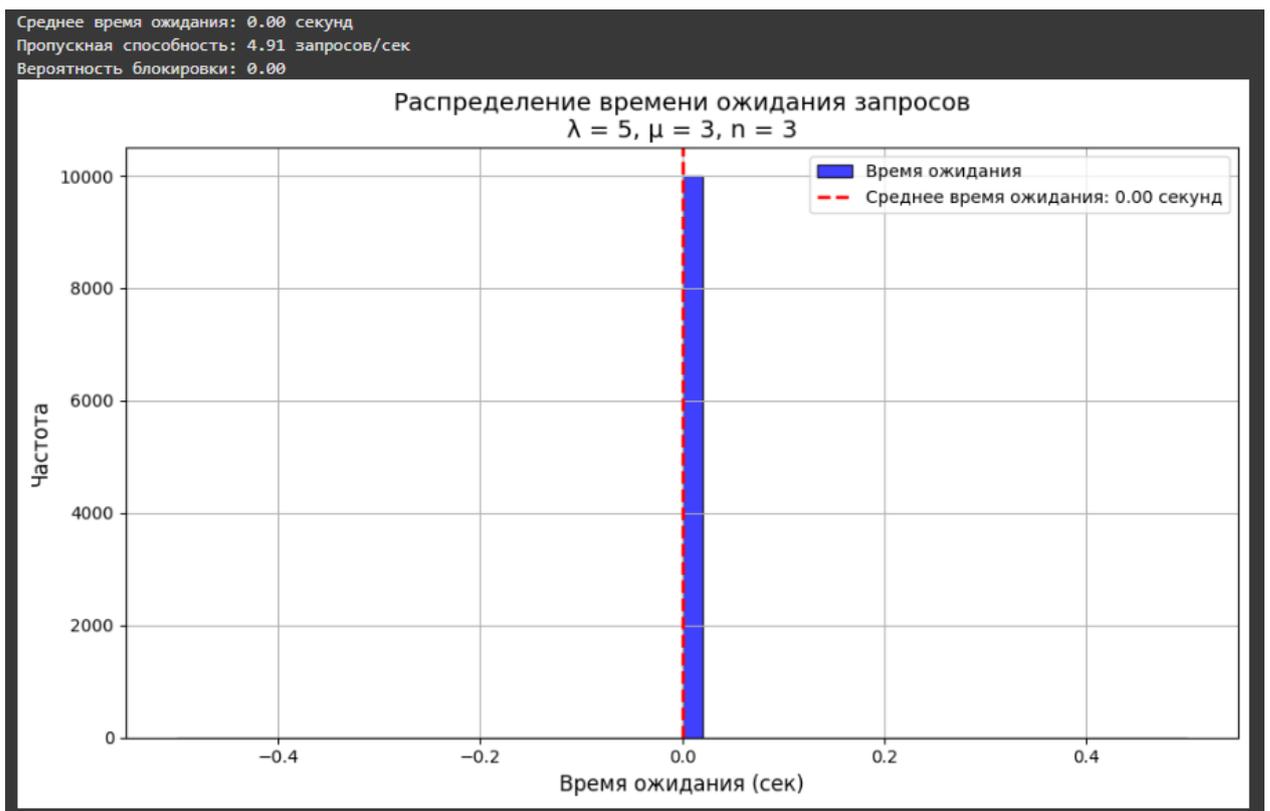


Рис. 1. – Результат выполнения программного кода при заданных параметрах

Наша система моделирует очередь М/М/1, создавая случайные интервалы времени для поступления заявок и их обслуживания. Далее, на

основе этих значений, рассчитываются такие параметры, как время ожидания и пропускная способность системы.

Результаты анализа позволяют оценить эффективность работы системы, а также графически представить распределение времени ожидания запросов. Здесь подтверждаются основные теоретические принципы, в которых производительность облачной системы хранения данных определяется балансом между интенсивностью поступления запросов и возможностями системы по их обслуживанию.

### **Заключение**

В ходе моделирования облачной системы хранения данных с использованием модели  $M/M/n$  были получены результаты, которые иллюстрируют поведение системы в зависимости от различных параметров: интенсивность поступления запросов, интенсивность обслуживания, количество серверов и количество запросов.

Результаты демонстрируют, что данная модель позволяет эффективно анализировать производительность системы и графически отображать ожидание запросов.

Также они подчеркивают важность балансировки нагрузки между серверами и оптимизации параметров обслуживания для поддержания стабильности системы при высоком объеме запросов. Увеличение числа серверов или повышение интенсивности обслуживания значительно снижает вероятность блокировки и время ожидания, что, в свою очередь, улучшает качество обслуживания пользователей.

### **Литература**

1. Бородко А.В., Пантюхин О.И. Реализация облачных систем хранения на основе центров обработки данных // 3-я международная научно-



техническая конференция студентов, аспирантов и молодых ученых, Санкт-Петербург. 2017. – С. 62-67.

2. Ekwonwune E.N. Analysis of Secured Cloud Data Storage Model for Information // Journal of Software Engineering and Applications. 2024. Т. 17. №. 5. С. 297-320.

3. Гордюшин, А.В., Лебедева С.В. Облачные технологии. Технология создания "облака" // Вестник молодых ученых Санкт-Петербургского государственного университета технологии и дизайна. 2014. № 3. С. 53-57.

4. George A.S., Sagayarajan S. Securing cloud application infrastructure: understanding the penetration testing challenges of IaaS, PaaS, and SaaS environments // Partners Universal International Research Journal. 2023. Т. 2. №. 1. pp. 24-34.

5. Афонин В.В., Никулин В.В. Оптимизация многоканальных систем массового обслуживания при больших нагрузках // Вестник Астраханского государственного технического университета. Серия: управления, вычислительная техника и информатика. 2020. №. 3. С. 105-115.

6. Попова Т.М., Слободчиков В.А. Моделирование работы web-сервера на основе системы массового обслуживания // Инженерный вестник Дона, 2024. № 8. URL: [ivdon.ru/ru/magazine/archive/n8y2024/9438/](http://ivdon.ru/ru/magazine/archive/n8y2024/9438/).

7. Афонин В.В., Никулин В.В. Оптимизация марковских систем массового обслуживания с ожиданием в системе MATLAB // Вестник Астраханского государственного технического университета. Серия: управления, вычислительная техника и информатика. 2017. №. 2. С. 39-47.

8. Ахметшин Д.А., Титовцев А.С. Система массового обслуживания с взаимопомощью между каналами и ограниченным временем пребывания в очереди // Инженерный вестник Дона, 2024. № 12. URL: [ivdon.ru/ru/magazine/archive/n12y2024/9688/](http://ivdon.ru/ru/magazine/archive/n12y2024/9688/).

---

9. Полетаев, В.П., Ершов А.Н. Оптимизация управления пропускной способностью системы массового обслуживания с ожиданием // Труды международного симпозиума "Надежность и качество". 2007. Т. 1. С. 239-240.

10. Бусарев, М.И., Кирпичников А.П., Флакс Д.Б. Одноканальная система массового обслуживания с ограниченным средним временем пребывания заявки в системе в целом // Вестник Казанского технологического университета. 2011. № 22. С. 155-161.

11. Афонин В.В., Никулин В.В. Оптимизация марковских систем массового обслуживания с отказами в системе MATLAB // Вестник Астраханского государственного технического университета. Серия: управления, вычислительная техника и информатика. 2018. №. 1. С. 112-120.

12. Шабанов А.П. Подход к оценке производительных ресурсов информационных систем // Бизнес-информатика. 2009. № 2(8). С. 58-63.

13. Афонин В.В., Федосин С.А. Моделирование систем. — 2-е изд., исправленное. Москва: ООО "Издательство "БИНОМ. Лаборатория знаний", 2016. 231 с.

14. Кузьмин А.Н. Оценка качества автоматизированной системы, основанной на технологии хранилищ данных // Вестник Казанского технологического университета. 2006. № 2. С. 315-323.

### References

1. Borodko A.V., Pantyukhin O.I. 3-ya mezhdunarodnaya nauchno-tekhnicheskaya konferentsiya studentov, aspirantov i molodykh uchenykh, Sankt-Peterburg. 2017. pp. 62-67.

2. Ekwonwune E.N. Journal of Software Engineering and Applications. 2024, T. 17. №. 5. pp. 297-320.

3. Gordyushin A.V., Lebedeva S.V. Vestnik molodykh uchenykh Sankt-Peterburgskogo gosudarstvennogo universiteta tekhnologii i dizayna. 201. №. 3. pp. 53-57.
4. George A.S., Sagayarajan S. Partners Universal International Research Journal. 2023. Т. 2. №. 1. pp. 24-34.
5. Afonin V.V., Nikulin V.V. Vestnik Astraxanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: upravleniya, vychislitel'naya tekhnika i informatika. 2020. №. 3. pp. 105-115.
6. Popova T.M., Slobodchikov V.A. Inzhenerny vestnik Dona, 2024, №. 8. URL: [ivdon.ru/ru/magazine/archive/n8y2024/9438/](http://ivdon.ru/ru/magazine/archive/n8y2024/9438/).
7. Afonin V.V., Nikulin V.V. Vestnik Astraxanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: upravleniya, vychislitel'naya tekhnika i informatika. 2017. №. 2. pp. 39-47.
8. Akhmetshin D.A., Titovtsev A.S. Inzhenerny vestnik Dona, 2024, №. 12. URL: [ivdon.ru/ru/magazine/archive/n12y2024/9688/](http://ivdon.ru/ru/magazine/archive/n12y2024/9688/).
9. Poletaev V.P., Ershov A.N. Trudy mezhdunarodnogo simpoziuma "Nadezhnost' i kachestvo". 2007. Т. 1. pp. 239-240.
10. Busarev M.I., Kirpichnikov A.P., Flaks D.B. Vestnik Kazanskogo tekhnologicheskogo universiteta. 2011. №. 22. pp. 155-161.
11. Afonin V.V., Nikulin V.V. Vestnik Astraxanskogo gosudarstvennogo tekhnicheskogo universiteta. Seriya: upravleniya, vychislitel'naya tekhnika i informatika. 2018. №. 1. pp. 112-120.
12. Shabanov A.P. Biznes-informatika. 2009. №. 2(8). pp. 58-63.
13. Afonin V.V., Fedosin S.A. Modelirovanie sistem [Modeling of systems]. 2-e izd., ispravlennoye. Moskva: OOO "Izdatel'stvo "BINOM. Laboratoriya znaniy", 2016. 231 p.
14. Kuzmin A.N. Vestnik Kazanskogo tekhnologicheskogo universiteta. 2006. №. 2. pp. 315-323.

**Дата поступления: 3.11.2024**

**Дата публикации: 1.01.2025**

---