

На правах рукописи



Соколов Александр Михайлович

**Аналитические и программные методы оценки
характеристик производительности вычислительных
систем с приоритетным обслуживанием**

Специальность 2.3.5 —

«Математическое и программное обеспечение вычислительных
систем, комплексов и компьютерных сетей»

Автореферат

диссертации на соискание учёной степени
кандидата технических наук

Москва — 2024

Работа выполнена в Федеральном государственном бюджетном учреждении науки Институте проблем управления им. В.А. Трапезникова Российской академии наук (ИПУ РАН).

Научный руководитель: доктор технических наук, профессор
Вишневский Владимир Миронович

Официальные оппоненты: **Гайдамака Юлия Васильевна**,
доктор физико-математических наук,
профессор, Федеральное государственное
автономное образовательное учреждение выс-
шего образования «Российский университет
дружбы народов имени Патриса Лумумбы»,
профессор кафедры теории вероятностей и
кибербезопасности.

Степанов Михаил Сергеевич,
кандидат технических наук, доцент, Феде-
ральное государственное бюджетное образо-
вательное учреждение высшего образования
«Московский технический университет связи
и информатики», доцент кафедры «Сети свя-
зи и системы коммутации».

Ведущая организация: Федеральное государственное бюджетное
учреждение науки «Институт радиотехники
и электроники имени В.А. Котельникова»
Российской академии наук

Защита состоится 16 сентября 2024 года в 15:00 на заседании диссертационного совета 24.1.107.03 при Институте проблем управления им. В.А. Трапезникова Российской академии наук по адресу: 117997 Москва, ул. Профсоюзная, 65.

С диссертацией можно ознакомиться в библиотеке Федерального государственного бюджетного учреждения науки Института проблем управления имени В.А. Трапезникова Российской академии науки и на сайте www.ipu.ru.

Автореферат разослан _____ 2024 года.

Ученый секретарь
диссертационного совета
24.1.107.03,
д.т.н., доцент



Барабанова Е.А.

Общая характеристика работы

Актуальность темы. Одной из важных задач при проектировании и анализе вычислительных систем является исследование характеристик производительности. Во многих вычислительных системах и сетях связи данная задача усложняется использованием приоритетного доступа пользователей к приложениям и услугам. Примерами систем с приоритетным трафиком являются системы IoT (Internet of Things), системы для распределенных вычислений, сетевые балансировщики нагрузки и др. При проектировании систем важным аспектом являются характеристики производительности: время отклика, время до начала обслуживания заявки, среднее число заявок в системе, среднее число заявок в очереди и др. Учитывая стохастический характер поступления и обработки задач, в рассматриваемых системах основной интерес представляют стационарные характеристики производительности. В качестве математического аппарата для отыскания и исследования стационарных характеристик производительности приоритетных стохастических систем эффективно применяются методы теории очередей (теории массового обслуживания).

В качестве входных потоков рассматриваемых систем массового обслуживания (СМО) используются коррелированные МАР-потоки, описывающие трафик в компьютерных системах и сетях. Естественным обобщением МАР-потоков на случай приоритетного неоднородного трафика является ММАР (Marked Markov Arrival Process) - маркированный Марковский входной поток, который позволяет описывать коррелированное поступление заявок для произвольного предусмотренного в системе числа приоритетов $K \geq 2$. Для моделирования обработки заявки в теории очередей используется РН-распределение (Phase type, распределение фазового типа). РН - распределение является обобщением ряда распределений, таких как: распределение Эрланга k -го порядка, гиперэкспоненциальное распределение и др. Более того, для любого неотрицательного распределения случайной величины $A(t)$ можно подобрать РН-распределение сколь угодно близкое к распределению $A(t)$ в условиях слабой сходимости.

Особый интерес для исследования характеристик производительности вычислительных систем представляют системы с несколькими обслуживающими приборами. Также на практике реализуется вариант системы с ограниченным буфером памяти. Например, в системе балансировки сетевой нагрузки есть ограничения на число одновременных сетевых запросов, часть из которых находится в ожидании (в буфере) до тех пор, пока не освободится сервер для обработки. Аналогично в системах для распределенных вычислений, где для обработки задач предусмотрено несколько серверов и буфер ограниченной памяти для задач. В общем виде приоритетные системы в терминах Кендалла обозначаются как ММАР/РН/ N/R , где ММАР - входной поток заявок с приоритетами,

в котором предусмотрено K классов приоритетов, RH - распределение времени обслуживания, N - число обслуживающих приборов, а R - размер буфера в системе .

Исследованиям вычислительных приоритетных систем на базе математических моделей теории очередей посвящен ряд работ, среди которых следует особо отметить работы отечественных и зарубежных учёных: В.М. Вишневский, В.И. Клименок, А.Н. Дудин, В.В. Рыков, К.Е. Самуйлов, С.Н. Степанов, Ю.В. Гайдамака, С.П. Моисеева, А.А. Назаров, Е.В. Морозов, Р.Л. Смелянский, В.Г. Карташевский, Р.В. Разумчик, D. Lucantoni, M.F. Neuts, G. Horvath, B. Sun, V. Ramaswami, A. Joohnson, G. Horvath, A. Krishnamoorthy и др.

Несмотря на большое число работ отечественных и зарубежных исследователей в области приоритетных систем, до настоящего времени отсутствуют методы отыскания характеристик производительности систем большой размерности с входным ММАР-потокотом и RH -распределением времени обслуживания. Нахождение стационарных характеристик производительности вычислительных систем с приоритетным обслуживанием большой размерности - сложная задача в силу экспоненциального роста числа состояний в системе в зависимости от ее параметров. Поэтому разработка аналитических и программных методов оценки характеристик производительности вычислительных систем с приоритетным обслуживанием обуславливает актуальность и новизну диссертационной работы.

Объектом исследования являются вычислительные системы с приоритетным обслуживанием.

Предметом исследования являются методы получения оценок характеристик производительности вычислительных систем с приоритетным обслуживанием.

Целью исследования является создание комплекса моделей и программных средств для анализа характеристик производительности вычислительных систем с приоритетным обслуживанием.

Для достижения поставленной цели были решены следующие задачи:

1. Разработка и реализация алгоритма вычисления стационарных характеристик производительности системы массового обслуживания вида ММАР/ RH / N / R с приоритетным обслуживанием.
2. Разработка имитационной модели для получения оценок стационарных характеристик производительности системы ММАР/ RH / N / R большой размерности с приоритетным обслуживанием.
3. Разработка комбинированного метода получения быстрых оценок стационарных характеристик производительности вычислительной системы с приоритетным обслуживанием.

4. Проектирование архитектуры, разработка аналитической модели и программного комплекса для потоковых вычислений с контейнеризацией и приоритизацией задач, позволяющего ускорить процесс получения численных результатов, в том числе результатов моделирования.

Научная новизна полученных результатов заключается в следующем:

1. Разработана математическая модель и программный комплекс для получения точного аналитического решения и стационарных характеристик производительности системы с распределенной обработкой данных и приоритетным обслуживанием. (п.9 паспорта специальности 2.3.5)
2. Впервые предложена программная инфраструктурная организация комплекса имитационного моделирования, позволяющая анализировать характеристики производительности системы с распределенной обработкой данных и приоритетным обслуживанием. (п.8 паспорта специальности 2.3.5)
3. Впервые в теории очередей разработан метод исследования приоритетных систем большой размерности, основанный на комбинации методов имитационного моделирования и машинного обучения, и инструментальное средство для его реализации. (п.4 паспорта специальности 2.3.5)
4. Предложена оригинальная архитектура, разработаны алгоритмы организации взаимодействия программ распределенной системы для организации потоковых вычислений, позволяющие повысить скорость получения оценок характеристик производительности вычислительных систем с приоритетным обслуживанием. (п.3 паспорта специальности 2.3.5)

Область исследования. Диссертационная работа соответствует паспорту специальности 2.3.5 «Математическое и программное обеспечение вычислительных систем, комплексов и компьютерных сетей», а именно следующим пунктам:

- п.3. Модели, методы, архитектуры, алгоритмы, языки и программные инструменты организации взаимодействия программ и программных систем.
- п.4. Интеллектуальные системы машинного обучения, управления базами данных и знаний, инструментальные средства разработки цифровых продуктов.
- п.8. Модели и методы создания программ и программных систем для параллельной и распределенной обработки данных, языки и инструментальные средства параллельного программирования.

- п.9. Модели, методы, алгоритмы, облачные технологии и программная инфраструктура организации глобально распределенной обработки данных.

Теоретическая и практическая значимость. Аналитические и имитационные модели и методы, разработанные в диссертации, могут эффективно использоваться для оценки характеристик производительности многолинейных приоритетных вычислительных систем. Разработанный программный комплекс в рамках диссертационного исследования может применяться для улучшения эффективности получения результатов научных расчетов. Практическая значимость диссертационной работы подтверждается актами о внедрении, полученными от НИИ «Центрпрограммсистем» и МФТИ ГУ. Результаты работы также были представлены в исследованиях, проводимых по грантам Российского фонда фундаментальных исследований №19-07-00919, №20-37-70059 и Российского научного фонда №22-49-02023.

Методология и методы исследования. Для решения задач, поставленных в диссертации, использовались методы теории массового обслуживания, теории вероятности, теории случайных процессов, методы имитационного моделирования. При разработке программного обеспечения использовались методы и алгоритмы программных систем.

Основные положения, выносимые на защиту:

1. Математическая модель двухприоритетной системы массового обслуживания вида $M/M/1/N/R$ и программный комплекс для вычисления стационарных характеристик производительности вычислительных систем с приоритетным обслуживанием.
2. Дискретно-событийная модель для получения оценок стационарных характеристик производительности системы большой размерности с приоритетным обслуживанием.
3. Комбинированный метод получения быстрых оценок стационарных характеристик производительности вычислительной системы с использованием методов имитационного моделирования и машинного обучения.
4. Математическая модель системы массового обслуживания и программный комплекс для вычисления стационарных характеристик производительности вычислительных систем с групповым поступлением заявок.
5. Архитектура распределенной системы и программный комплекс для потоковых вычислений с контейнеризацией и приоритизацией задач.

Достоверность полученных результатов обеспечивается использованием строгих математических моделей, сравнением результатов аналитического и имитационного моделирования. Полученные оценки стационарных

характеристик производительности согласуются с результатами, полученными в ходе экспериментов.

Апробация работы. Основные результаты работы докладывались на конференциях: «Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems» (Москва, 13-17 апреля 2020); «Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems» (Москва, 19-23 апреля 2021); «5th International Scientific Conference on Information, Control, and Communication Technologies» (Астрахань, 4-7 октября 2021); «6th International Scientific Conference on Information, Control, and Communication Technologies» (Астрахань, 2-7 октября 2022); «International Conference named after A. F. Terpugov Information Technologies and Mathematical Modelling» (онлайн, 1-5 декабря, 2021); «International Conference on Distributed Computer and Communication Networks: Control, Computation, Communications» (Москва, 25-29 сентября 2023); «XVIII Всероссийская школа-конференция молодых ученых Управление Большими Системами» (Воронеж, 5-8 сентября 2023).

Личный вклад. Основные результаты диссертации, выносимые на защиту, получены автором самостоятельно.

Публикации. Основные результаты по теме диссертации изложены в 15 печатных изданиях, 3 из которых изданы в журналах, рекомендованных ВАК, 7 — в периодических научных журналах, индексируемых Web of Science и Scopus, 5 — в тезисах докладов. Зарегистрирована 1 программа для ЭВМ.

Содержание работы

Во введении обосновывается актуальность исследований, проводимых в рамках данной диссертационной работы, формулируется цель, ставятся задачи работы, излагается научная новизна.

Первая глава посвящена описанию СМО вида ММАР/РН/Н/Р и исследованию частного случая - двухприоритетной системы: поиску стационарного состояния системы, а также характеристик производительности. В данной главе приведен обзор современных исследований СМО с приоритетным обслуживанием, показана актуальность и новизна настоящего исследования, а также приведена постановка задачи исследования.

СМО с приоритетным обслуживанием (рис. 1) работает следующим образом. Заявки разных приоритетов поступают в ММАР-поток под управлением неприводимой цепи Маркова с непрерывным временем ν_t , $t \geq 0$, которая принимает значения на множестве $\{0, 1, 2, \dots, W\}$. Всего в системе предусмотрено K приоритетов. Система состоит из буфера емкости R и N обслуживающих приборов. Полагаем, что все обслуживающие приборы одинаковы и независимы друг от друга. Время обслуживания

любым прибором заявки k -го ($k = 1, \dots, K$) типа имеет фазовое распределение (PH - Phase type distribution), которое задается парой (β_k, S_k) . Здесь β_k —вектор-строка порядка M_k , а S_k —квадратная матрица порядка M_k .

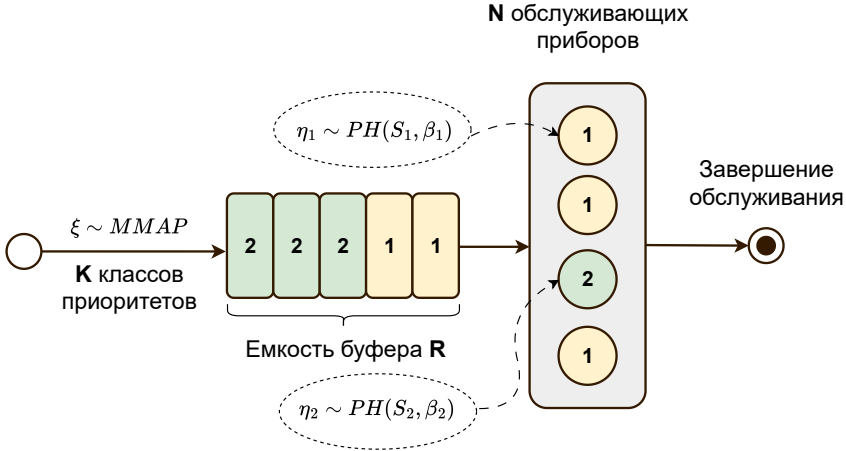


Рис. 1 — Система массового обслуживания вида MMAP/PH/N/R с приоритетным обслуживанием заявок

Предполагаем, что заявки типа $k' < k$, $k, k' = \overline{1, K}$ обладают относительным приоритетом. Это означает, что более приоритетные заявки стоят в буфере впереди всех менее приоритетных заявок. Поступающая заявка типа k' , заставшая все приборы занятыми и в очереди $i, i = 0, R - 1$, заявок, с вероятностью $q_i^{(k')}$ ставится впереди всех неприоритетных заявок типов $k = k' + 1, k' + 2, \dots, K$ и в конце очереди приоритетных заявок типов $k = 1, 2, \dots, k'$. С вероятностью $1 - q_i^{(k')}$ новая заявка решает не присоединяться к очереди и уходит из системы навсегда. Если заявка любого типа, поступающая в систему, застает систему полностью занятой, то она покидает систему навсегда. Для упрощения записи, в случае $K = 2$ типов заявок, будем обозначать вероятности присоединения к очереди длины i приоритетных заявок класса $k = 1$ как $q_i \equiv q_i^{(1)}$, а неприоритетных заявок класса $k = 2$ как $f_i \equiv q_i^{(2)}$.

В силу сложности построения инфинитезимального генератора для системы общего вида и, соответственно, поиска стационарных вероятностей и характеристик производительности в данной главе рассмотрен частный случай системы - двухприоритетная система ($K=2$). Для данного случая в работе построен инфинитезимальный генератор:

Утверждение 1. Инфинитезимальный генератор цепи Маркова ξ_t имеет блочную структуру:

$$Q = \begin{pmatrix} T & Q_{0,1} & O & \dots & O & O \\ Q_{1,0} & Q_{1,1} & Q_{1,2} & \dots & O & O \\ O & Q_{2,1} & Q_{2,2} & \dots & O & O \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ O & O & O & \dots & Q_{R-1,R-1} & Q_{R-1,R} \\ O & O & O & \dots & Q_{R,R-1} & Q_{R,R} \end{pmatrix}, \quad (1)$$

где ненулевые блоки T , $Q_{i,i'}$ - некоторые блочные матрицы. Алгоритм вычисления этих матриц подробно описан в диссертационной работе.

Вектор стационарного состояния системы $\mathbf{p}$ определяется как единственное решение системы линейных алгебраических уравнений

$$\mathbf{p}Q = \mathbf{0}, \mathbf{p}\mathbf{e} = 1. \quad (2)$$

Рассчитав стационарное распределение, можно найти ряд важных стационарных характеристик производительности системы. В работе приведены формулы для вычисления характеристик производительности, включая среднее число заявок в буфере, среднее число заявок в системе, среднее число приоритетных заявок в системе, среднее число занятых приборов и др.

Зависимость времени расчета стационарных характеристик производительности системы от параметров системы

Порядок инфинитезимального генератора Q цепи Маркова приоритетной системы имеет оценку сверху:

$$P \leq \frac{(R+1)(R+2) + 2N + 2}{2} (W+1)(N+1) \frac{(N+M)^{(N+M+\frac{1}{2})}}{M^{M+\frac{1}{2}} N^{N+\frac{1}{2}}}, \quad (3)$$

где R - размер буфера системы, N - число обслуживающих приборов, M - порядок матрицы S распределения времени обслуживания P_H , W - порядок матриц D_i ММАР-потока.

В качестве иллюстрации влияния параметров системы на время вычисления стационарных вероятностей состояний системы в табл. 1 приведено время вычисления для следующих параметров: матрицы D_i имеют размер 5×5 , размер матрицы S равен 5×5 , размер буфера равен 8. Вычисления производились на компьютере, имеющем процессор Intel Core i7-9750H и 16 ГБ ОЗУ, при меняющемся числе обслуживающих приборов. При числе обслуживающих приборов $N = 5$ для получения результата не хватило оперативной памяти компьютера.

Из рис. 2 видно, что размер матрицы инфинитезимального генератора экспоненциально зависит от порядка матрицы генератора ММАР-потока и от числа обслуживающих приборов. Кроме того, размер матрицы

линейно зависит от порядка матрицы S PH-распределения и квадратично от размера буфера.

Таблица 1 — Зависимость времени построения генератора и нахождения стационарных вероятностей состояний системы MMAP/PH/N/R от числа обслуживающих приборов

Число приборов	Время, с
1	0.03
2	1.2
3	34
4	1155
5	Недостаточно памяти

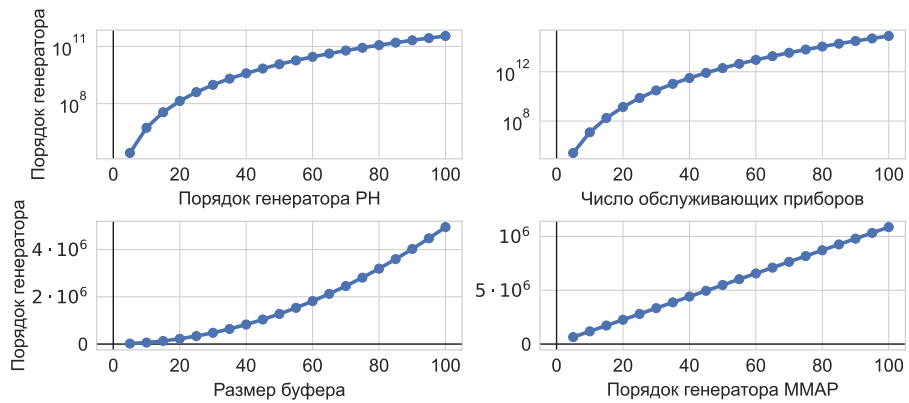


Рис. 2 — Зависимость порядка матрицы инфинитезимального генератора от параметров системы

Все расчеты аналитической модели выполнены на языке Python с использованием библиотеки `numpy`. Численные результаты по данной главе доступны в GitLab репозитории по ссылке <https://gitlab.com/lab69/priority-queues>.

Вторая глава посвящена исследованию приоритетных систем большой размерности. В предыдущем разделе аналитическое решение приведено только для приоритетных систем с двумя классами трафика ($K=2$). Для систем общего вида из-за большого числа состояний цепи Маркова построить инфинитезимальный генератор, найти стационарное распределение, вычислить все необходимые стационарные характеристики производительности системы - сложная аналитическая задача. В связи с этим для получения оценок характеристик производительности системы используется численный метод - имитационное (дискретно-событийное)

моделирование. Также для получения быстрых оценок стационарных характеристик производительности системы с приоритетным обслуживанием предложен комбинированный алгоритм, базирующийся на методах имитационного моделирования и методах машинного обучения.

Дискретно-событийная модель системы с приоритетным обслуживанием

Дискретно-событийная имитационная модель позволяет вычислять характеристики производительности приоритетной системы, моделируя происходящие в ней события: приход новых заявок, начало и завершение их обслуживания. Время в модели изменяется скачками в моменты начала обработки событий. Во время исполнения модели вычисляются значения характеристик: длины очередей, задержки, общее время занятости приборов и др. По завершении выполнения собранные данные усредняются для получения оценок стационарных значений. Главное ограничение метода имитационного моделирования - необходимость моделирования очень большого числа событий для получения устойчивых оценок. Число событий зависит как от параметров системы (емкость буфера, число приборов, число классов приоритетов, размерностей управляющих цепей), так и от характеристик случайных распределений.

Реализация дискретно-событийной модели приоритетной системы выполнена на языке Python с использованием библиотеки для научных расчетов PyQumo, которая доступна по ссылке <https://github.com/ipu69/pyqumo>. Для валидации дискретно-событийной модели использовались аналитические расчеты для двухприоритетной системы. На рис. 3 показан результат сравнения работы имитационной модели и аналитических расчетов в зависимости от изменяющихся параметров: размера буфера и числа обслуживаемых приборов. Точность результата, полученного с помощью метода Монте-Карло, зависит от числа сгенерированных заявок. В ходе валидации число сгенерированных заявок равнялось 1 млн. Погрешность составила не более одного процента для каждой характеристики производительности.

Получение быстрых оценок стационарных характеристик производительности системы

Для получения быстрых оценок характеристик производительности приоритетных систем большой размерности предложен комбинированный метод, базирующийся на методах имитационного моделирования и машинного обучения. Схема метода показана на рис. 4. Суть метода заключается в том, что на основе сгенерированного с помощью дискретно-событийной модели набора данных обучить модель машинного обучения для получения оценок характеристик производительности. Сгенерированный набор данных состоял из 200000 записей. В данной главе приводится сравнение использования различных алгоритмов при получении быстрых оценок стационарных характеристик производительности системы, а также решается

несколько прикладных задач. Первая задача - предсказание времени отклика системы, вторая задача - классификация системы по вероятности потери приоритетной заявки. Если вероятность потери меньше 0.1, то система считается надежной, в противном случае - нет.

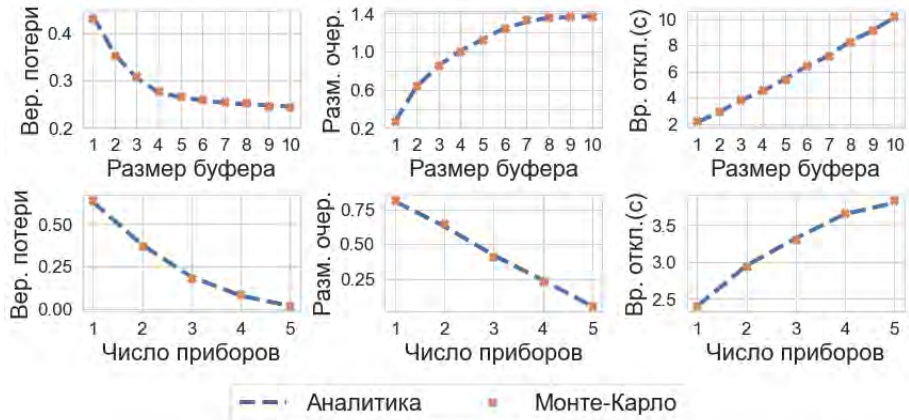


Рис. 3 — Валидация дискретно-событийной модели. Сравнение результатов выполнения имитационной модели и аналитического решения

Для параметризации ММАР-потока и РН-распределения и использования их параметров в качестве признаков модели машинного обучения в работе использовались алгоритмы восстановления Джонсона. Восстановить ММАР-поток можно, используя характеристики распределения: коэффициент вариации, асимметрии и коэффициент автокорреляции. Также для восстановления необходимы интенсивности поступления заявок каждого класса. Для параметризации интенсивностей использовалась схема, при которой для восстановления интенсивностей всех заявок нужно знать: интенсивность поступления заявок наиболее и наименее приоритетных классов, а также аппроксимирующую функцию, по которой восстанавливались интенсивности для заявок остальных классов приоритетов. Для параметризации РН-распределения использовалась аналогичная схема, но с ограничением - все распределения времен обслуживания заявок каждого класса имели одинаковые коэффициенты вариации и асимметрии. А для задания интенсивностей обслуживания использовалась схема аналогичная описанной для ММАР-потока. Более подробно алгоритм параметризации и входные параметры модели описаны в диссертационной работе.

В табл. 2 приведены значения различных метрик эффективности различных алгоритмов ML. По результатам обучения оказалось, что лучшие

метрики показывает искусственная нейронная сеть. В задаче классификации по вероятности потери заявки (табл. 3) лучшие значения показывает также искусственная нейронная сеть. Результаты представлены по ссылке <https://gitlab.com/lab69/priority-queues>.

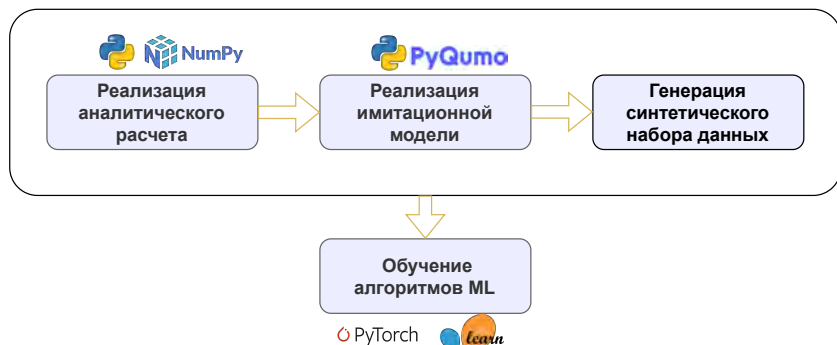


Рис. 4 — Схема обучения алгоритмов ML для получения оценок характеристик производительности приоритетной системы

Таблица 2 — Значения метрик для моделей регрессии для предсказания времени отклика системы

Модель	Корреляция	R2	Ошибка
Решающее дерево	0.944	0.891	0.135
Случайный лес	0.980	0.961	0.047
Градиентный бустинг	0.980	0.961	0.047
Нейронная сеть	0.987	0.973	0.031

Таблица 3 — Сравнение различных методов машинного обучения для задачи классификации системы по вероятности потери заявки

Модель	precision	recall	f1 score
Случайный лес	0.93	0.94	0.93
Градиентный бустинг	0.95	0.95	0.95
Нейронная сеть	0.97	0.98	0.97

Третья глава посвящена разработке программного комплекса для потоковых вычислений с целью ускорения получения результатов научных расчетов. Предлагаемая система для потоковых вычислений отличается от существующих систем. Во-первых, при реализации вычислительных

алгоритмов не требуется использовать какие-либо специальные библиотеки, а применение контейнеризации позволяет эффективно изолировать окружение, в котором выполняются задачи, от операционной системы узла-обработчика, требуя меньше ресурсов по сравнению с традиционной виртуализацией. Во-вторых, в предлагаемой системе задачи и подзадачи при выполнении никак не связаны и не синхронизируются. Более того, порядок выполнения подзадач внутри одной задачи никак не определен заранее. Соответственно, не требуется использовать библиотеки для меж-процессного взаимодействия наподобие MPI.

В рамках диссертационного исследования данная система ускорила получение результатов с помощью имитационного моделирования. Также данная система использовалась для генерации синтетического набора данных, который был использован для обучения алгоритмов ML с целью быстрых оценок характеристик производительности приоритетных систем.

Архитектура программного комплекса для потоковых вычислений.

По своей структуре программный комплекс для потоковых вычислений является веб-приложением, которое состоит из нескольких базовых сервисов, показанных на рис. 5. Пользователь взаимодействует с системой с помощью графического интерфейса. Система включает в себя серверную часть, которая необходима для управления задачами, авторизации пользователей и сбора статистики. Супервизор используется для управления обработчиками подзадач и мониторинга очереди задач/подзадач. Программа-обработчик (обработчик) используется для выполнения отдельной подзадачи. Обработчик можно запустить на любом вычислительном ресурсе для вычисления подзадачи. Для организации очереди задач/подзадач используется Redis. Для хранения остальных данных используется база данных Postgres.

Для запуска и работы с системой пользователь загружает на вход JSON-файл с описанием задачи, а также указывает docker-образ, на котором система будет выполнять данную задачу. Описание задачи в JSON-файле представляет собой массив наборов входных параметров, на которых нужно выполнить задачу. Отдельный элемент этого массива называется подзадачей.

Приоритизация задач

В соответствии с важностью задач в системе задачам назначаются приоритеты от первого (высокого) до пятого (низкого) с целью ускорения их выполнения. Система поддерживает четыре стратегии приоритизации. **FIFO** - задача, поступившая в систему раньше, выполняется раньше. В этом режиме приоритет не имеет значения. **UNIFORM** - программы-обработчики поровну делятся между выполняющимися задачами. Если обработчиков меньше, чем задач, обработчики берут на выполнение задачи, которые появились в системе раньше. **PRIORITY FIFO** -

задачи сортируются по приоритету и времени поступления. **PRIORITY UNIFORM** - задачи с наивысшим приоритетом вычисляются первыми, обработчики распределяются поровну между задачами с одинаковым приоритетом. По результатам исследования наилучшей стратегией для выполнения приоритетных задач оказалась **PRIORITY FIFO**. Если рассматривать случай, когда приоритеты не учитываются, то наиболее эффективной стратегией оказалась **FIFO**.

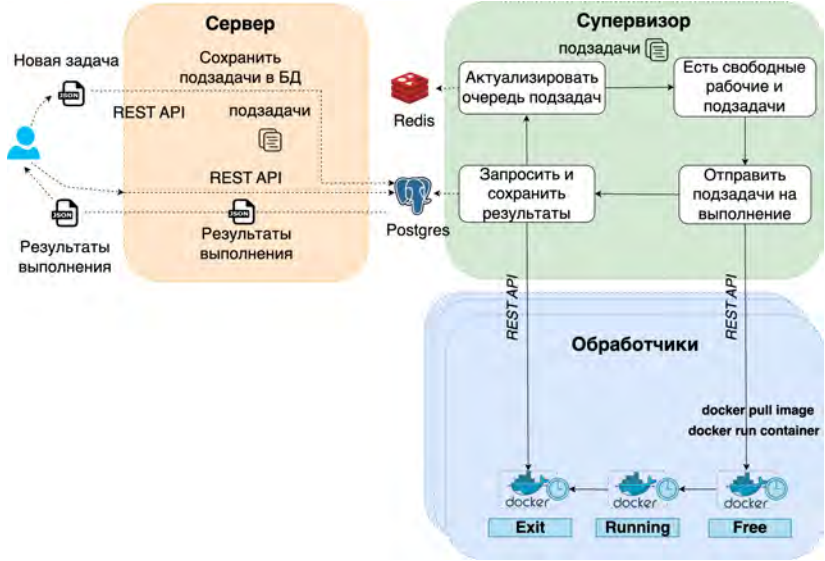


Рис. 5 — Архитектура программного комплекса для потоковых вычислений

Оценка длительности вычислений задачи и подзадачи

Обозначим число процессоров, предусмотренных в системе для расчета, как m и пусть n - число обработчиков, запущенных в системе, p - число подзадач, а t_0 - длительность вычисления одной подзадачи одним обработчиком на изначально свободном процессоре, то есть время вычисления подзадачи без учета времени ожидания освобождения процессора. Тогда теоретическая «оптимистичная» зависимость времени выполнения всей задачи от числа обработчиков и числа процессоров в системе определяется как:

$$T(n, m, p) = \begin{cases} \frac{pt_0}{n}, & n \leq m \\ \frac{pt_0}{m}, & n > m \end{cases} \quad (4)$$

Среднее время расчета одной подзадачи, учитывающее, что процессорное время может делиться между несколькими обработчиками, определяется как $\bar{t} = t_0 \max(1, \frac{n}{m})$

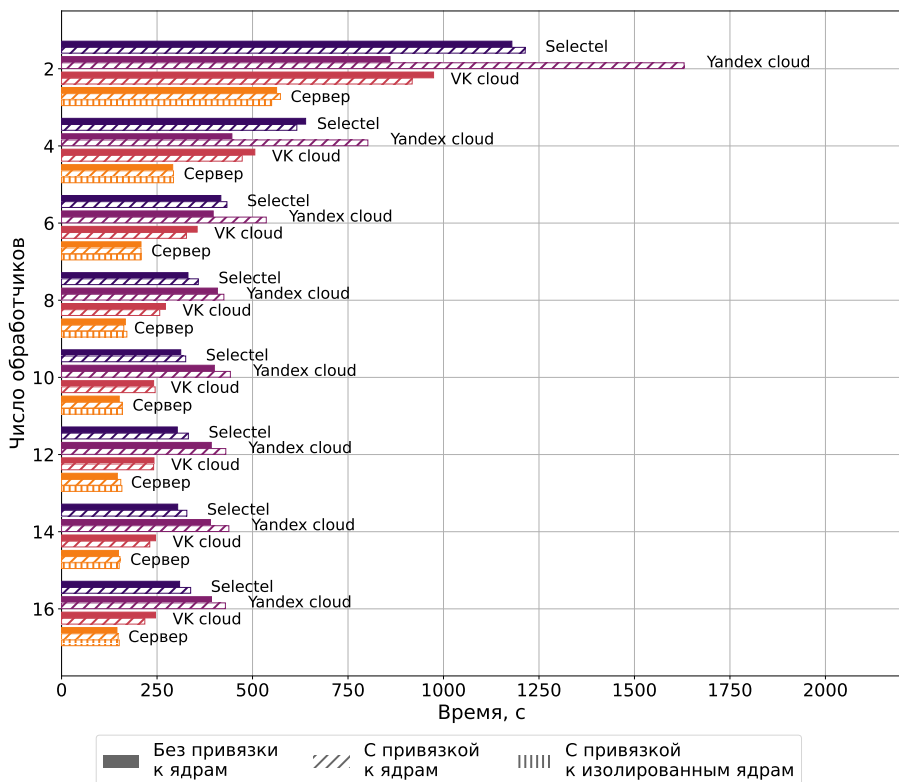


Рис. 6 — Время выполнения задачи при разных режимах выполнения: без привязки обработчиков к ядрам, с привязкой обработчиков к ядрам и с привязкой обработчиков к выделенным ядрам

В рамках диссертационной работы проведен эксперимент, в котором исследовалась зависимость времени выполнения задачи от числа обработчиков. Эксперимент проводился на следующих платформах: Yandex Cloud - процессор Intel Xeon Gold 6338, тактовая частота 2.00 ГГц; Selectel - процессор Intel Xeon Gold 6140, 2.3 ГГц; VK Cloud - Intel Xeon Gold 6230, 2.1 ГГц, а также на выделенной рабочей станции. Облачные серверы, использованные в экспериментах, имели по 8 виртуальных ядер. На рабочей станции был установлен процессор Intel Core i7-9700K, также имеющий 8 ядер и не поддерживающий hyper-threading. Эксперимент был произведен при разных режимах запуска контейнера: без привязки обработчика к ядру, с привязкой обработчика к ядру. Кроме того, для рабочей станции был произведен эксперимент с привязкой к изолированным ядрам. Для привязки обработчика к ядру использовалась опция для контейнера `cpuset_cpus`,

принимая значение в диапазоне доступных ядер, для машины с 8 ядрами от 0 до 7 включительно.

Результаты (рис. 6) демонстрируют, что время выполнения задачи согласуется с теоретическими выкладками. До того момента, как в системе запущено 8 программ-обработчиков, которое равно числу ядер виртуальной машины, время выполнения уменьшается. Когда число обработчиков становится больше числа ядер виртуальной машины, время выполнения задачи остается на прежнем уровне. Также результаты эксперимента показывают, что большого смысла в явной привязке обработчиков к ядрам нет. Результаты исследований по данной главе диссертации доступны по ссылкам: <https://gitlab.com/lab69/multi-runner-metrics>, <https://github.com/timac11/multirunner-analytics-impl>.

В **четвертой главе** приведены примеры использования имитационной модели для получения оценок стационарных характеристик производительности технологии балансировки нагрузки с приоритизированным трафиком HaProxy, а также системы для распределенных вычислений с приоритизацией задач Celery. В рамках работы приведено подробное описание технологий, а также обоснована возможность использования предложенных в работе математической и имитационной моделей для получения оценок стационарных характеристик производительности. В рамках работы оценки, полученные с помощью моделирования, сравнивались с оценками характеристик производительности, полученными из собранного стенда. В экспериментах на вход балансировщику HaProxy и системе Celery генерировался ММАР-поток с различными параметрами. Время обработки одного запроса или задачи было распределено экспоненциально. Для сбора статистики использовались Prometheus-метрики, для их визуализации - сервис Grafana.

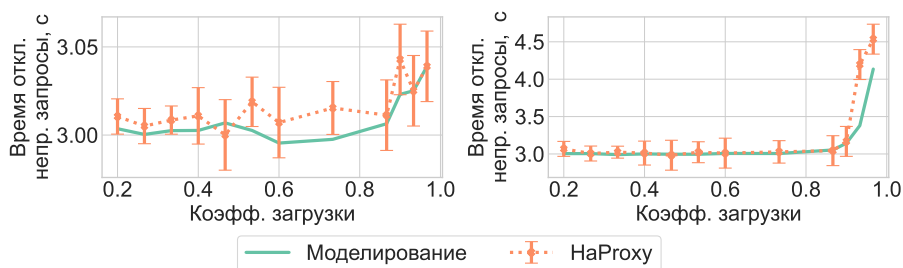


Рис. 7 — Сравнение характеристик производительности балансировщика нагрузки HaProxy с результатами имитационного моделирования

На рис. 7 показано сравнение времени отклика, вычисленного с помощью имитационного моделирования и полученного из тестового стенда

системы с балансировщиком нагрузки NaProху для приоритетных и не-приоритетных запросов, в зависимости от коэффициента загрузки. Измерение времени отклика производилось при следующих параметрах системы. Время обработки запроса было распределено экспоненциально, причем для приоритетных и не-приоритетных интенсивность обработки запросов была одинаковой $\mu_{pr} = \mu_{npr} = \mu$. Интенсивность поступления приоритетных и не-приоритетных заявок в ММАР была одинакова $\lambda_{pr} = \lambda_{npr} = \lambda$. Число серверов равнялось 3, максимальное число одновременных подключений для каждого сервера равнялось 30, то есть для системы в целом всего одновременных подключений равно 90. Если провести аналогию с СМО с приоритетным обслуживанием заявок, суммарное число одновременных подключений фактически равно числу обслуживающих приборов. В рамках работы проведено сравнение стационарных характеристик производительности балансировщика нагрузки с результатами имитационного моделирования, включая время отклика и время ожидания для приоритетных и не-приоритетных запросов, среднее число запросов в системе и др.

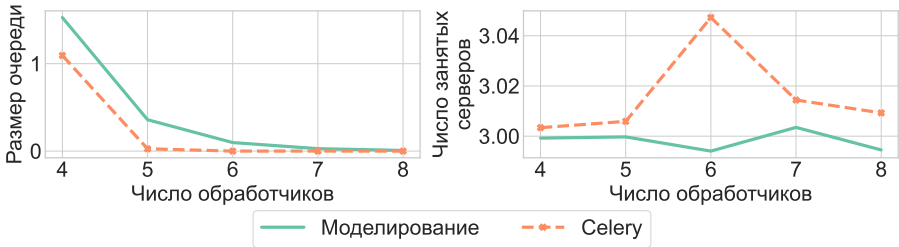


Рис. 8 — Сравнение характеристик производительности системы Celery с результатами имитационного моделирования

При исследовании применимости имитационной модели приоритетной системы для получения оценок характеристик производительности системы распределенных вычислений Celery в работе проведен эксперимент, результаты которого представлены на рис. 8. Стационарные характеристики производительности системы в ходе эксперимента были получены при следующих параметрах системы: интенсивность поступления приоритетных и не-приоритетных задач: $\lambda_{pr} = \lambda_{npr} = 3c^{-1}$; интенсивность обслуживания приоритетных и не-приоритетных задач $\mu_{pr} = \mu_{npr} = \mu = 1c^{-1}$. Число обработчиков менялось в интервале от 4 до 8. В рамках эксперимента для запуска обработчиков был собран тестовый стенд, состоящий из трех виртуальных машин.

По результатам проведения экспериментов имитационная модель показала высокую эффективность по времени и точности получения оценок стационарных характеристик производительности системы с балансировщиком нагрузки NaProху и системы для организации распределенных

вычислений Celery. Погрешность получения результатов не превышала 5-10%. Результаты доступны по ссылке <https://github.com/timac11/thesis-queueing-models-examples>.

В **заключении** приведены основные результаты работы:

1. Разработана аналитическая модель двухприоритетной мультисервисной системы вида ММАР/РН/Н/Р с приоритетным обслуживанием, получены формулы для расчета стационарных характеристик производительности системы.
2. Реализован алгоритм расчета характеристик производительности вычислительной системы с приоритетным обслуживанием для случая, когда в системе предусмотрено два класса приоритетов.
3. Разработана дискретно-событийная модель для получения оценок стационарных характеристик производительности вычислительной системы большой размерности с приоритизацией заявок.
4. Разработан комбинированный метод для получения быстрых оценок стационарных характеристик производительности вычислительной системы с приоритетным обслуживанием, базирующийся на методе имитационного моделирования и методах машинного обучения.
5. Предложена архитектура и разработан программный комплекс для потоковых вычислений с контейнеризацией и приоритизацией задач, позволяющий ускорить процесс получения численных результатов, в том числе результатов моделирования.
6. Разработана аналитическая модель с групповым поступлением заявок для исследования системы потоковых вычислений, получены формулы для стационарных характеристик производительности.
7. Произведен анализ применимости дискретно-событийной модели вычислительной системы с приоритетным обслуживанием для получения оценок характеристик производительности системы с сетевым балансировщиком нагрузки, а также вычислительной системы для распределенных вычислений.

Публикации автора по теме диссертации

В изданиях из списка ВАК РФ

1. Исследование характеристик приоритетной мультисервисной системы ММАР/РН/М/Н с использованием метода Монте-Карло / В. Вишневецкий, В. Клименок, А. Ларионов, А. Мухтаров, А. М. Соколов // Управление большими системами. — 2023. — № 103. — С. 6—39.

2. Исследование fork-join системы с марковским входным потоком и распределением времени обслуживания фазового типа / В. Вишневский, В. Клименок, А. Соколов, А. Ларионов // Проблемы информатики. — 2023. — Т. 4. — С. 29—56.
3. Архитектура распределенной системы для потоковых вычислений с контейнеризацией и приоритизацией задач / А. М. Соколов, А. Ларионов, В. Вишневский, А. Мухтаров // Информационные технологии и вычислительные системы. — 2023. — Т. 4. — С. 5—18.

В изданиях, входящих в международную базу цитирования Scopus

4. Architecture of a Distributed Parallel Computing System Using Docker Cluster / A. Sokolov, A. Larionov, A. Mukhtarov, I. Fedotov // Proceedings of the 2022 International Conference on Information, Control, and Communication Technologies, ICCT 2022. — 2022.
5. *Sokolov, A.* Distributed System for Scientific and Engineering Computations with Problem Containerization and Prioritization / A. Sokolov, A. Larionov, A. Mukhtarov // Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). — 2024. — Vol. 14123 LNCS. — P. 68—82.
6. Performance evaluation of the priority multi-server system $mmap/ph/m/n$ using machine learning methods / V. M. Vishnevsky, V. I. Klimenok, A. M. Sokolov, A. A. Larionov // Mathematics. — 2021. — Vol. 9, no. 24.
7. Investigation of the Fork-Join System with Markovian Arrival Process Arrivals and Phase-Type Service Time Distribution Using Machine Learning Methods / V. M. Vishnevsky, V. I. Klimenok, A. M. Sokolov, A. A. Larionov // Mathematics. — 2024. — Vol. 12, no. 5.
8. *Larionov, A. A.* The calibration method of a tandem queueing model with PH service time using NS-3 simulation of a multihop wireless network / A. A. Larionov, A. A. Mukhtarov, A. M. Sokolov // Journal of Physics: Conference Series. — 2021. — Vol. 2091, no. 1. — P. 012030.
9. *Sokolov, A.* Examining the Performance of a Distributed System Through the Application of Queuing Theory / A. Sokolov, O. Semenova, A. Larionov // Communications in Computer and Information Science. — 2024. — Vol. CCIS 2129. — P. 16—32.
10. *Sokolov, A.* Application of Queueing Theory to Investigation of HaProxy Load Balancer Performance Characteristics / A. Sokolov // Communications in Computer and Information Science. — 2024. — Vol. CCIS 2129. — P. 89—100.

Зарегистрированные программы для ЭВМ

11. *Свидетельство о государственной регистрации программы для ЭВМ. Программный комплекс для расчета характеристик производительности систем массового обслуживания с приоритизацией заявок* / А. М. Соколов ; Федеральное государственное бюджетное учреждение науки Институт проблем управления им. В.А.Трапезникова РАН. — № 2024616177 ; заявл. 21.03.2024 ; опубл. 04.04.2024, 2024617735 (Российская Федерация).

В сборниках трудов конференций

12. *Mukhtarov, A. A base station placement of an wireless network with linear topology and a network performance evaluation with NS-3* / A. Mukhtarov, A. M. Sokolov // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем: материалы Всероссийской конференции с международным участием, Москва, 19–23 апреля 2021 года. — 2021. — С. 425–430.
13. *Sokolov, A. M. Architecture of a distributed system for parallel execution of numerical experiments on large data sets* / A. M. Sokolov, A. A. Larionov // Информационно-телекоммуникационные технологии и математическое моделирование высокотехнологичных систем: материалы Всероссийской конференции с международным участием, Москва, 19–23 апреля 2021 года. — 2021. — С. 474–480.
14. *Использование инструмента управления кластером Docker swarm в архитектуре распределенной системы параллельных вычислений* / А. Соколов, А. Ларионов, А. Мухтаров, И. Федотов // Материалы 15-й Международной конференции «Технические средства систем управления и связи» (Астрахань, 2022). — 2022. — С. 154–156.
15. *Sokolov, A. About an architecture and performance of calculations of a distributed system for parallel execution of numerical experiments on large data sets* / A. Sokolov, A. Larionov // Материалы 14-й Международной конференции «Технические средства систем управления и связи» (Астрахань, 2021). — 2021. — С. 84–85.
16. *Соколов, А. М. Разработка программного комплекса для потоковых вычислений с использованием технологии контейнеризации docker* / А. М. Соколов, А. А. Мухтаров // Труды 19-ой Всероссийской школы-конференции молодых ученых «Управление большими системами» (УБС'2023, Воронеж). — 2023. — С. 581–588.

Соколов Александр Михайлович

Аналитические и программные методы оценки характеристик
производительности вычислительных систем с приоритетным обслуживанием

Автореф. дис. на соискание ученой степени канд. тех. наук

Подписано в печать _____._____._____. Заказ № _____

Формат 60×90/16. Усл. печ. л. 1. Тираж 100 экз.

Типография _____