На правах рукописи

Тутов Андрей Владимирович

МОДЕЛИ И МЕТОДЫ РАСПРЕДЕЛЕНИЯ ИНФОРМАЦИОННЫХ И ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ ГЕТЕРОГЕННЫХ ЦЕНТРОВ ОБРАБОТКИ ДАННЫХ

Специальность 2.3.8 —
«Информатика и информационные процессы»

Автореферат
диссертации на соискание учёной степени
кандидата технических наук

Москва — 2026

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

Научный руководитель: **Фархадов Маис Паша оглы**
доктор технических наук, старший научный сотрудник, заведующий лабораторией «Эргатических систем» ИПУ РАН

Официальные оппоненты:

Ведущая организация:

Защита состоится « » _____2026 г. в ____ на заседании диссертационного совета 24.1.107.03 при Федеральном государственном бюджетном учреждении науки Институте проблем управления им. В.А. Трапезникова Российской академии наук по адресу: 117997, Москва, Профсоюзная ул., 65, ИПУ РАН. С диссертацией можно ознакомиться в научно-технической библиотеке и на сайте ИПУ РАН https://www.ipu.ru

Автореферат разослан «__» _____ 2026 года.

Ученый секретарь
диссертационного совета,
доктор технических наук, доцент Е.А. Барабанова

Общая характеристика работы

Актуальность темы. За последние несколько лет во всем мире наблюдается стремительный рост использования цифровых сервисов, объемов информационных потоков в телекоммуникационных сетях и центрах обработки данных (ЦОД). Перед операторами связи и сервис-провайдерами стоят задачи совершенствования собственной ИТ-инфраструктуры с использованием отечественного программного обеспечения для быстрого реагирования на возрастающие потребности общества.

Современные дата-центры должны располагать запасом вычислительной мощности, достаточным для стабильного функционирования сервисов даже при пиковых нагрузках. Расширяется не только сегмент классических веб-решений — активно развиваются направления, требующие значительных вычислительных ресурсов: глубокое обучение, большие данные и поддержка инфраструктуры виртуальных рабочих столов. Высокая потребность в параллельных вычислениях стимулирует внедрение ускорителей на базе графических процессоров (Graphics Processing Unit — GPU). Как следствие, архитектура ЦОД эволюционирует в сторону гетерогенности, сочетая классические вычислительные узлы и серверы с GPU.

В основе работы современных цифровых платформ лежат облачные технологии, которые, в свою очередь, опираются на виртуализацию. Без применения технологий виртуализации становятся невозможными такие важные свойства облачной среды, как динамическое масштабирование (эластичность) ресурсов, использование ресурсов по требованию, оплата только за использованные ресурсы. Данные требования к качеству обслуживания оформляются в виде соглашений об уровне сервиса (Service Level Agreement — SLA).

Разнородность требований к вычислительным средам диктует необходимость построения гибкой системы управления ресурсами в облачных ЦОД. Ключевыми инструментами здесь выступают механизмы горизонтального и вертикального масштабирования, а также технология живой миграции виртуальных машин. Их применение позволяет оперативно перераспределять мощности инфраструктуры в ответ на колебания нагрузки. В результате удается гарантировать заданный уровень качества обслуживания без необходимости создавать избыточный резерв аппаратных ресурсов.

Повышение энергоэффективности ЦОД остается одной из главных проблем. На долю цифровых технологий приходится около 12% общемирового потребления электроэнергии. Неэффективное управление ресурсами может привести к неравномерному распределению нагрузки и, как следствие, невыполнению SLA, появлению «горячих» точек в машинных залах, усиленной работе системы охлаждения и, в итоге, к увеличенному энергопотреблению.

Системы управления ресурсами в облачных ЦОД должны быть автономными и поддерживать устойчивое состояние с учетом воздействия

внешних факторов. Как и в других автономных вычислительных системах в них реализуются функции мониторинга и прогноза нагрузки, планирования и распределения ресурсов.

Необходимо совершенствовать процесс распределения ресурсов, разрабатывать и внедрять эффективные модели, методы и алгоритмы управления ресурсами для нахождения компромисса между множеством противоречивых критериев, таких как обеспечение заданного в SLA-соглашениях качества обслуживания, равномерной загрузки вычислительных ресурсов, минимизации энергопотребления.

Степень разработанности темы. На сегодняшний день существует большое количество работ, посвященных моделированию и оценке производительности инфокоммуникационных систем и сетей, повышению эффективности их функционирования, среди которых следует особо отметить отечественные научные группы под руководством Вишневского В.М., Самуйлова К.Е., Воеводина В.В., Смелянского Р.Л., Шабанова Б.М., Каляева И.А., Корнеева В.В., а также групп зарубежных исследователей Buyya R., Menasce D. A. и других.

Число публикаций, в которых отражены результаты исследований таких систем, с каждым годом возрастает, что подтверждает актуальность данной проблемы. Однако в них рассмотрены преимущественно традиционные и облачные ЦОД, без учета стремительно возросшего спроса на GPU-вычисления. Управление ресурсами на основе совокупности эффективных моделей и методов, которые могут быть положены в основу планировщика вычислительных ресурсов облачного гетерогенного ЦОД, остается нерешенной задачей.

Целью диссертационной работы является обеспечение высокой энергоэффективности инфокоммуникационной системы гетерогенного ЦОД и стабильности облачных сервисов путем разработки моделей и методов управления ресурсами программно-аппаратного комплекса распределенного планировщика ресурсов.

Для достижения поставленной цели в работе решены следующие **задачи**:

1. Анализ проблем повышения эффективности функционирования гетерогенных центров обработки данных.
2. Разработка модели и метода многокритериального первоначального размещения разнородных виртуальных машин с учетом критериев энергопотребления, нарушения SLA-соглашений и загрузки ресурсов.
3. Исследование метода прогнозирования загрузки серверов для принятия решения о необходимости миграции виртуальных машин.
4. Разработка и исследование критерия устойчивости инфокоммуникационной системы ЦОД путем определения оптимального размера окна наблюдения за серверами с учётом возможных помех, вызванных миграциями виртуальных машин.

5. Исследование характеристик процесса миграции виртуальных машин. Разработка метода оценки длительности миграции виртуальных машин.
6. Разработка метода многокритериального выбора целевых серверов для миграции виртуальных машин, обеспечивающий баланс между эффективностью использования вычислительных ресурсов и соблюдением SLA-соглашений.
7. Разработка имитационных моделей. Подтверждение эффективности разработанных моделей и методов.
8. Разработка алгоритмов и рекомендаций по практической реализации моделей и методов программно-аппаратного комплекса распределенного планировщика ресурсов в инфокоммуникационной системе гетерогенного облачного ЦОД, которые могут быть включены в отечественные облачные платформы.

Объектом исследования являются процессы распределения информационных и вычислительных ресурсов в гетерогенных ЦОД.

Предметом исследования являются математические модели и методы эффективного распределения информационных и вычислительных ресурсов в гетерогенных ЦОД.

Методы исследования. В ходе исследования использовались методы теории вероятностей и математической статистики, многокритериальной оптимизации, прогнозирования и имитационного моделирования.

Научная новизна работы заключается в исследовании и разработке критериев, моделей и методов повышения эффективности программно-аппаратного комплекса планировщика информационных и вычислительных ресурсов гетерогенного ЦОД.

Основными научными результатами являются:

1. Разработаны модель и метод первоначального размещения виртуальных машин в гетерогенном ЦОД, которые*, в отличие от существующих аналогов*, *учитывают* ресурсы GPU-серверов и *обеспечивают* сбалансированное решение по множеству критериев, в то время как применяемые на практике методы обычно ограничиваются единственным критерием (п.1, пнс 2.3.8);
2. Разработаны модель и метод рационального динамического размещения виртуальных машин в ЦОД, направленные на одновременную минимизацию критериев энергопотребления и нарушений SLA-соглашений. Предложенный метод *в отличие от используемых на практике*, позволяет находить точное решение задачи оптимизации в режиме реального времени и дает в среднем в 3,5 раза лучшие результаты (п.3, пнс 2.3.8);
3. Разработан *новый* критерий определения рациональной длительности окна наблюдения за серверами. При его расчете используется предложенный метод оценки времени «живой» миграции виртуальных

машин, что в совокупности позволяет повысить качество мониторинга и, как следствие, стабильность облачных сервисов (п.3, пнс 2.3.8).

Практическая значимость

1. Исследовано применение метода группового учета аргументов для прогнозирования загрузки физических серверов. Показано, что данный метод позволяет добиться более высокой точности прогноза по сравнению с известными аналогами, что, в свою очередь, ведет к снижению количества необоснованных миграций виртуальных машин.
2. Созданы имитационные модели распределения ресурсов ЦОД, которые позволяют проверить эффективность предложенных методов и алгоритмов динамического распределения ресурсов.
3. Предложены алгоритмы, рекомендации, архитектура и программные модули для практической интеграции моделей и методов программно-аппаратного комплекса распределенного планировщика информационных и вычислительных ресурсов в гетерогенных ЦОД.

Предложенные в диссертации математические модели и методы были внедрены в центрах обработки данных ПАО «МТС», ПАО «ВымпелКом», ООО «АЭР - Технологические Решения» (AliExpress Россия), АО «БВТ БАРЬЕР РУС» (международный холдинг BWT).

Результаты диссертационной работы используются в дисциплине «Вычислительные системы» для направления магистратуры 09.04.01 «Информатика и вычислительная техника» (программа «Архитектура информационных систем»).

Соответствие паспорту специальности

Работа выполнена в соответствии со следующими пунктами паспорта специальности 2.3.8 «Информатика и информационные процессы».

- П.1. Разработка компьютерных методов и моделей описания, оценки и оптимизации информационных процессов и ресурсов, а также средств анализа и выявления закономерностей на основе обмена информацией пользователями и возможностей используемого программно-аппаратного обеспечения.
- П.9. Разработка архитектур программно-аппаратных комплексов поддержки цифровых технологий сбора, хранения и передачи информации в инфокоммуникационных системах, в том числе, с использованием «облачных» интернет-технологий и оценка их эффективности.
- П.14. Разработка и исследование принципов организации и функционирования распределенных информационных систем и баз данных, прикладных протоколов информационных сетей, форматов представления данных и языков информационного поиска в распределенных информационных ресурсах.

Основные положения, выносимые на защиту

1. Модель и метод многокритериального первоначального размещения разнородных виртуальных машин позволяет достичь баланса между множеством противоречивых критериев, таких как энергопотребление, качество обслуживания, использование ресурсов по сравнению с распространенными на практике алгоритмами «первый подходящий» и «наиболее подходящий».
2. Метод прогнозирования перегрузки и недогрузки серверов на основе метода группового учета аргументов позволяет минимизировать число прямых и обратных миграций виртуальных машин, тем самым повысить стабильность облачных сервисов.
3. Метод расчета длительности живой миграции виртуальных машин для нахождения приближенного аналитического выражения плотности распределения вероятностей позволяет наиболее точно определить размер скользящего окна наблюдения за серверами.
4. Метод многокритериального динамического размещения виртуальных машин по критериям нарушения SLA-соглашений и эффективности использования вычислительных ресурсов, по сравнению с распространенным на практике эвристическим алгоритмом "первый подходящий с упорядочиванием по убыванию", позволяет получить в среднем в 3,5 раза лучшие результаты по комбинированному критерию энергопотребления и нарушения SLA-соглашений.

Достоверность результатов обеспечивается за счет корректного применения математического аппарата, программного обеспечения и подтверждается результатами расчетов и имитационного моделирования.

Апробация работы. Основные положения и выводы диссертационного исследования прошли апробацию на всероссийских и международных научных конференциях, в том числе международной научно-технической конференции «Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь» (Москва, DCCN-2016, 2017, 2018, 2025); международной отраслевой научно-технической конференции «Технологии информационного общества» (Москва, 2017 гг., 2023 г.); конференции «Телекоммуникационные и вычислительные системы» (Москва, 2017, 2018, 2021 гг.), на международной научно-практической конференции «Менеджмент качества, транспортная и информационная безопасность, информационные технологии» (Москва, IT&MQ&IS–2018), на конференции 2018 Systems of Signals Generating and Processing in the Field of on Board Communications (Москва, 2018); на 24-й и 28-й международных конференциях Ассоциации открытых инноваций FRUCT (Москва, 2019, 2021 гг.); International Workshop on Model-Driven Organizational and Business Agility (Москва, 2022); 2022 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED) (Москва, 2023); XXI Международной научно-практической конференции «Новые информационные технологии в

образовании (Технологии 1С в цифровой трансформации экономики и социальной сферы)» (Москва, 2022).

Публикации. По материалам диссертационного исследования опубликовано 30 печатных работ. Из них 13 статей размещены в рецензируемых изданиях, входящих в перечень ВАК, 8 — в журналах, индексируемых международными базами Scopus и Web of Science.

Содержание работы

Во **введении** обоснована актуальность темы диссертационного исследования, сформулированы цель и задачи работы. Представлены основные научные результаты, полученные в ходе исследования, показана их практическая значимость, приведены сведения об апробации. Изложены положения, выносимые на защиту.

В первой главе проведен анализ основных проблем, стоящих перед операторами связи и провайдерами облачных услуг. В ответ на современные вызовы, такие как взрывной рост использования цифровых сервисов в мире, Интернета вещей, способствующих появлению больших данных и внедрению машинного обучения, перед операторами связи и сервис-провайдерами стоят задачи совершенствования собственной ИТ-инфраструктуры для быстрого реагирования на возрастающие потребности общества. ЦОДы становятся гетерогенными. Ряд провайдеров предоставляет услуги аренды виртуальных машин с поддержкой графических ускорителей (vGPU). Согласно прогнозам iKS-Consulting, к 2030 году сегмент аренды GPU-серверов покажет существенный рост: его доля в общей структуре рынка достигнет 8,6%, увеличившись практически вдвое по сравнению с текущими показателями.

Технологической основой облачных платформ является виртуализация, которая позволяет формировать изолированные вычислительные среды — виртуальные машины и реализовать основные требования облачных сервисов.

Пользователи взаимодействуют с облачным провайдером через веб-портал, заказывая виртуальные машины с определенными характеристиками. Конфигурация создаваемых виртуальных машин может быть двух типов: стандартные ВМ, которые характеризуются лишь выделенными виртуальными процессорами (vCPU) и объемом виртуальной памяти (vRAM), либо ВМ, дополнительно оснащаемые vGPU для решения специализированных задач. Технологии виртуализации GPU развиваются. Появилась технология NVIDIA Multi-Instance GPU (MIG), позволяющая на аппаратном уровне разделять процессор и память видеокарты между несколькими виртуальными машинами.

Основной проблемой современных ЦОД является высокое энергопотребление, которое продолжает расти в связи с высоким спросом на высокопроизводительные вычисления на графических процессорах. Одним из путей решения данной проблемы является эффективное управление вычислительными ресурсами.

Проведенный анализ работ по управлению ресурсами в инфокоммуникационных системах ЦОД позволяет классифицировать их следующим образом:

- работы, посвященные статическому размещению ВМ;
- исследования в области динамического управления, которые включают в себя блоки мониторинга и прогноза нагрузки, а также вопросы выбора ВМ и физических серверов для миграции;
- работы, изучающие свойства и параметры различных типов миграции виртуальных машин.

Существующие подходы к управлению ресурсами часто не учитывают гетерогенность ЦОД. При этом выделение виртуальных машин с графическими процессорами накладывает дополнительные ограничения, связанные с памятью и конфигурацией GPU, что остается за рамками большинства исследований.

Сформулированы цель и задачи исследования, ориентированные на развитие подходов к управлению вычислительными мощностями (как физическими, так и виртуальными) в инфокоммуникационной составляющей гетерогенных облачных ЦОД.

<u>Во второй главе</u> диссертационной работы рассмотрено первоначальное размещение виртуальных машин в гетерогенном ЦОД. Приведена многокритериальная постановка задачи оптимизации размещения виртуальных машин по физическим серверам. Критериями оптимизации являются следующие:

- **Энергопотребление** сервера с видеокартой является суммарным энергопотреблением хост-системы и видеокарты:
  $f_{power} = f_{pow_cpu}(u_{cpu}) + f_{pow_gpu}(u_{gpu})$:

$$f_{pow_cpu}(u_{CPU}) = \begin{cases} p_0 + (p_1 - p_0)\cdot u_{CPU}, & u_{CPU} > 0; \\ 0, & u_{CPU} = 0, \end{cases}$$

  где p_0 – энергопотребление простаивающего сервера без GPU;
  p_1 – энергопотребление полностью загруженного сервера без GPU.
  u_{cpu} – усредненная загрузка центральных процессоров (CPU) физического сервера.

$$f_{pow_gpu}(u_{GPU}) = \begin{cases} g_0 + (g_1 - g_0)\cdot u_{GPU}, & u_{GPU} > 0; \\ 0, & u_{GPU} = 0, \end{cases}$$

  где g_0 – уровень энергопотребления незагруженной видеокарты;
  g_1 – уровень энергопотребления видеокарты при полной загрузке;
  u_{gpu} – средняя загрузка GPU.
  Значения f_{power} приводится к интервалу от 0 до 1.
- **Неиспользованные ресурсы** (f_{res}):

$$f_{res}(u_{CPU}, u_{RAM}, u_{GPU}, u_{GDDRAM}) = 1 - (u_{CPU}\cdot u_{RAM} + u_{GPU}\cdot u_{GDDRAM})/2$$

  где u_{RAM} – средняя загрузка памяти;

u_{GDDRAM} – средняя загрузка памяти видеокарты.

Данный показатель характеризует степень сбалансированности загрузки процессора, оперативной памяти сервера и графических ускорителей.

– **Нарушения SLA-соглашений** (f_{SLA}):

$$f_{SLA}(u_{CPU}, u_{GDDRAM}) = 1 - \left(\frac{1}{1+e^{(u_{CPU} \cdot 100 - a)}} + \frac{1}{1+e^{(u_{GDDRAM} \cdot 100 - b)}} \right) / 2,$$

где u_{CPU} — загрузка центрального процессора. При превышении порогового значения a наблюдается снижение производительности вычислительной системы;

u_{GDDRAM} — загрузка видеопамяти GDDRAM. Превышение порога b также влечет за собой деградацию производительности.

Данные критерии зависят от загрузки вычислительных ресурсов, т.е. от размещения ВМ на физических серверах.

Математическая постановка задачи по трем критериям имеет следующий вид:

$$\min_{x_{ij}} \frac{1}{M} \sum_{j=1}^{M} f_{power}^{j}(x_{ij}),$$

$$\min_{x_{ij}} \frac{1}{M} \sum_{j=1}^{M} f_{SLA}^{j}(x_{ij}),$$

$$\min_{x_{ij}} \frac{1}{M} \sum_{j=1}^{M} f_{res}^{j}(x_{ij}),$$

при ограничениях

$$\sum_{i=1}^{N} r_i^{CPU} \cdot x_{ij} < c_j^{CPU},\ j = [1, ..., M],$$

$$\sum_{i=1}^{N} r_i^{RAM} \cdot x_{ij} < c_j^{RAM},\ j = [1, ..., M],$$

$$\sum_{i=1}^{N} r_i^{GPU} \cdot x_{ij} < c_j^{GPU},\ j = [1, ..., M],$$

$$\sum_{i=1}^{N} r_i^{GDDRAM} \cdot x_{ij} < c_j^{GDDRAM},\ j = [1, ..., M],$$

$$\sum_{j=1}^{M} x_{ij} = 1,\ i = [1, ..., N].$$

где x_{ij} – двоичные переменные для индикации принадлежности i-ой ВМ к конкретному j-му физическому серверу. Они принимают единичное значение, если ВМ с индексом i назначена на сервер j, и нулевое — во всех остальных ситуациях.

$[r_i^{CPU}, r_i^{RAM}, r_i^{GPU}, r_i^{GDDRAM}]$ — вектор потребляемых ресурсов i-й ВМ;

$[c_j^{CPU}, c_j^{RAM}, c_j^{GPU}, c_j^{GDDRAM}]$ — вектор ресурсов, которыми располагает физический сервер j.

Одним из подходов к решению многокритериальных задач является сведение задачи к однокритериальной. Предлагается использовать

аддитивную свертку критериев, а саму однокритериальную задачу решать с использованием муравьиного алгоритма.

В ходе работы алгоритма каждый муравей генерирует решение — список размещения ВМ по серверам ($VM_i \rightarrow PM_j$). Вероятность выбора пары VM_i – PM_j определяется уровнем феромона на данной связи и эвристическим значением, которое вычисляется на основе текущей загрузки серверов.

Заданное количество муравьев формирует множество потенциальных решений. Для создания уникального распределения каждый муравей использует случайную перестановку списка ВМ, после чего пытается разместить машины в этом порядке на серверы, контролируя наличие требуемых ресурсов (vCPU, vRAM, vGPU, vGDDRAM) на каждом шаге.

Целевая функция формируется как взвешенная комбинация трех критериев:

$$fitness = \alpha_1 \cdot f_{power} + \alpha_2 \cdot f_{res} + \alpha_3 \cdot f_{sla} \rightarrow \min, \quad \sum \alpha_i = 1, \alpha_i \geq 0.$$

Если значение целевой функции (fitness) улучшается за время работы муравья, то её значение запоминается, а значения феромонов, соответствующие эффективному размещению VM_i – PM_j увеличиваются, а для неэффективных размещений уменьшаются. Блок-схема алгоритма приведена на рис. 1.

Путем имитационного моделирования выполнено сравнение предложенного алгоритма с базовыми стратегиями «самый подходящий» (First Fit – FF) и «наилучший подходящий» (Best Fit – BF), применяемыми для размещения виртуальных машин с графическими ускорителями.

Имитационная модель включала варьируемое количество гетерогенных серверов и ВМ. Требования к ресурсам ВМ генерировались случайным образом:

- vCPU (ядра) и vRAM (ГБ) — равномерно из набора {0,25, 0,5, 1, 1,5, 2, 2,5, 3, 4};
- пары vGPU/vGDDRAM — как профили NVIDIA MIG: (1,5), (2,10), (3,20), (7,40).

Объем доступных ресурсов серверов задавал общую емкость ЦОД. Для каждого сценария проводилось 20 независимых экспериментов с последующим усреднением результатов.

Эксперименты на конфигурации 100 ВМ / 60 серверов (рис. 2) подтвердили особенности исследуемых алгоритмов. FF и BF, минимизируя количество используемых серверов, провоцируют перегрузки и нарушения SLA. Предложенный ACO-алгоритм, напротив, жертвует энергоэффективностью ради сбалансированного распределения ресурсов, что существенно снижает уровень нарушений SLA и потери ресурсов. Подбор весовых коэффициентов целевой функции позволяет адаптировать решение под требуемый компромисс между критериями.

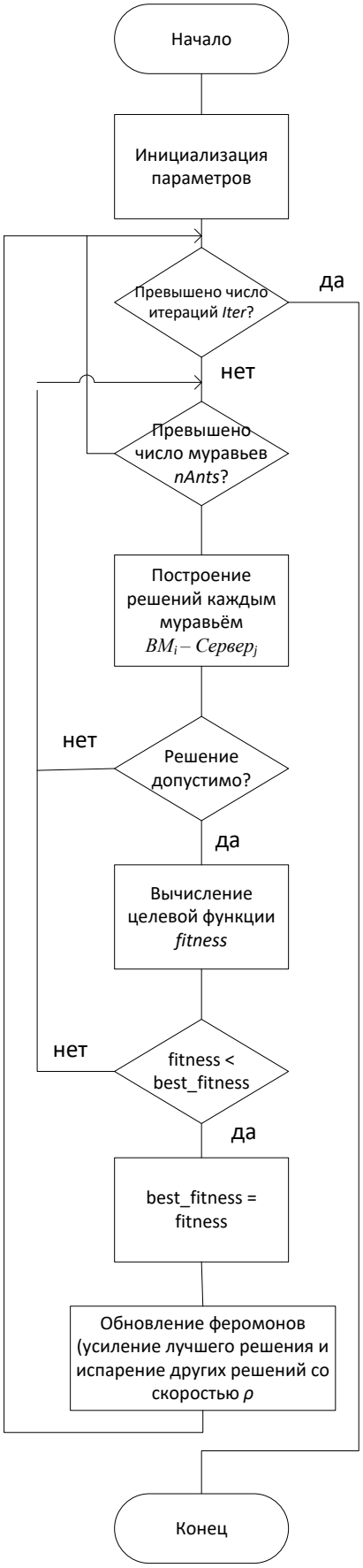

Рис. 1 – Блок-схема алгоритма размещения виртуальных машин методом муравьиной колонии

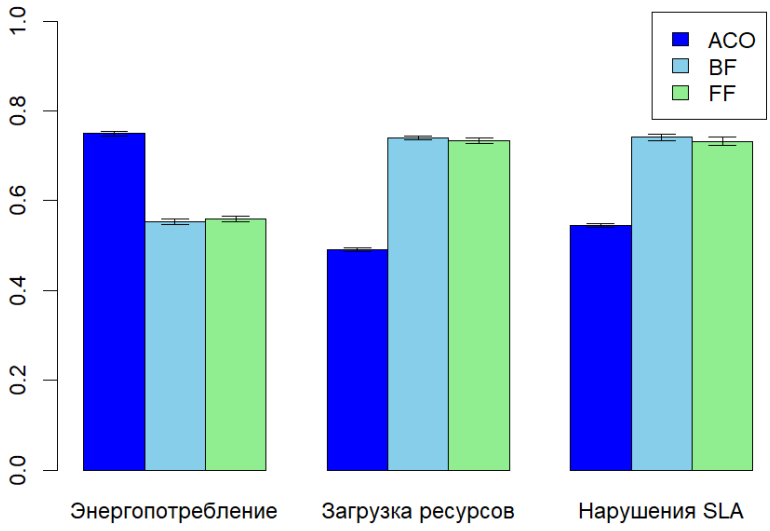

Рис. 2 – Сравнение предлагаемого алгоритма с известными эвристиками FF и BF

Установлено, что временные затраты алгоритма растут пропорционально квадрату числа ВМ — $O(n^2)$. Для 500 виртуальных машин расчет на Intel Core i5-10210U (1,6 ГГц, 8 ГБ) занимает в среднем 2 минуты.

Чувствительность параметров ACO исследовалась путем 100 независимых прогонов модели с последующим усреднением. Наилучшая комбинация параметров приведена в таблице 1.

Таблица 1 —Значения параметров муравьиного алгоритма

Число муравьев	**Число итераций**	**Относительная важность эвристического значения β**	**Параметр испарения феромона ρ**	**Параметр обновления феромона q**
6	30	2	0,3	1

<u>В третьей главе</u> рассматриваются вопросы динамического распределения ресурсов облачного ЦОД, включающие:

- мониторинг и прогноз дисбаланса загрузки серверов;
- отбор виртуальных машин для миграции;
- выбор принимающих физических серверов;
- проведение миграции.

Определяющим является этап прогнозирования: точная оценка будущей загрузки процессора сервера на ближайшем интервале наблюдения позволяет снизить число избыточных миграций и улучшить общую сбалансированность системы.

Формально постановку задачи прогнозирования можно сформулировать следующим образом. Пусть задан временной ряд загрузки процессора $Y = \{y_1, y_2, ..., y_T\}$. Необходимо найти функцию f, которая по предшествующим

значениям ряда $Y = \{y_{t-h}, y_{t-h-1}, ..., y_{t-1}\}$ позволяет получить прогноз $\hat{y}_t$ на горизонт h.

Обозначим:

$\tau_1, \tau_2 \in$ Ў – критические пороги недогрузки/перегрузки соответственно;

f – модель (алгоритм) прогнозирования, с параметрами θ, подлежащими оптимизации.

Требуется найти оптимальные параметры θ^* модели f, которые минимизируют ожидаемые потери на новых данных:

$$\theta^* = \arg\min_\theta \mathrm{M}_{(x,z):\ P_{data}}[L(z, f(x;\theta))],$$

где P_{data} – это неизвестное распределение данных, порождающее пары «признаки — целевая переменная» ($X = \{y_{t-h}, ..., y_{t-1}\}$, $Y = y_t$), а L — функция потерь, например, среднеквадратическая ошибка.

В качестве метода прогнозирования выбран метод группового учета аргументов (МГУА). Сглаживание временных рядов загрузки процессора выполняется с помощью скользящего окна. Миграция запускается контроллером при условии, что прогнозируемое на следующем окне значение загрузки процессора превышает верхний порог τ_2 или опускается ниже нижнего порога τ_1.

Важно оценить предлагаемые алгоритмы распределения ресурсов в крупномасштабной виртуализированной инфраструктуре центра обработки данных, однако проведение повторяемых крупномасштабных экспериментов на реальной инфраструктуре невозможно. Поэтому для подтверждения эффективности и настройки параметров выбранного метода прогнозирования было проведено имитационное моделирование с использованием пакета CloudSim на языке программирования Java. Среда моделирования CloudSim стала стандартом для оценки эффективности алгоритмов в области распределения ресурсов облачных ЦОД, последние обзоры часто ссылаются на CloudSim и для сравнения алгоритмов используют реализованные там метрики.

Моделирование проводилось для ЦОД, состоящего из 800 физических серверов двух типов и четырех типов ВМ. Использовалась реальная нагрузка, имеющаяся в CloudSim в рамках проекта PlanetLab, а также данные с кластеров Alibaba. Число виртуальных машин составило более 1000. Замеры загрузки производились каждые 5 минут

В начале процесса моделирования на каждый сервер произвольно размещались 1-2 виртуальные машины. В процессе моделирования осуществлялось размещение виртуальных машин на меньшем числе серверов и незагруженные серверы для экономии электроэнергии выключались. При прогнозировании загрузки хоста свыше 85% и ниже 20% принималось решение о миграции виртуальных машин.

Выполнено три серии экспериментов. Первая серия направлена на сравнение качества прогноза методом наименьших квадратов и локальной регрессии, как лучших в опубликованных работах. Эти методы сравнивались

с МГУА. Классические методы прогнозирования подходят для данной задачи, не требуют больших вычислительных ресурсов и позволяют получить приемлемый прогноз на малых выборках.

В работе используются следующие показатели эффективности:

- *OTF* — доля времени, в течение которого активные физические серверы функционируют в режиме полной загрузки;
- *PDM* — совокупное ухудшение производительности виртуальных машин, обусловленное процессом миграции;
- *SLAV* — интегральный показатель нарушения соглашений об уровне обслуживания, рассчитываемый как произведение *OTF* и *PDM*;
- *E*—энергопотребление физического сервера;
- *ESV* — комбинированный критерий: $ESV = E \cdot SLAV$.

Сравнение проводилось по комбинированному критерию *ESV*, являющего произведением критерия энергопотребления (*E*) и критерия нарушения SLA-соглашений (*SLAV*). Показано, что использование МГУА для прогнозирования перегрузки физических серверов на всех исследованных выборках по большей части превосходят результаты, опубликованные ранее и полученные методом локальной регрессии при размере скользящего окна в 10 точек. Результаты экспериментов приведены в таблице 2.

Таблица 2 — Сравнение методов прогнозирования перегрузки серверов

Метод	ESV ($\times 10^{-5}$)	Энерго-потреб-ление (кВт×ч)	SLAV ($\times 10^{-5}$)	OTF	PDM	Число миграций ($\times 10^{3}$)	Число обратных миграций ($\times 10^{3}$)
Локальная регрессия	17,94	163,15	0,11	0,14	0,08	27,63	4,6
МНК	46,29	185,17	0,25	0,15	0,17	33,86	5,46
МГУА	16,53	165,25	0,1	0,1	0,1	29,04	4,6

Вторая серия экспериментов была направлена на определение оптимального размера обучающей N_A и экзаменационной N_B выборки в МГУА. Построение прогностической модели осуществлялось с использованием полинома Колмогорова – Габора. Обучающая выборка использовалась для оценивания коэффициентов полинома, а экзаменационная — для выбора оптимальной структуры модели. В процессе генерации множества моделей-претендентов производилась селекция нескольких наилучших на основе критерия регулярности:

$$AR(B) = \frac{1}{N_B} \sum_{i=1}^{N_B} (y_i - y_i(B))^2 \to \min.$$

В ходе экспериментов размер скользящего окна принимал значения от 7 до 15 точек. Размер обучающей выборки в экспериментах составил 2/3 от всего объема. Лучшее качество прогнозирования достигнуто при использовании экзаменационной выборки, состоящей из двух последних точек ряда. Это

позволило более полно учесть последние данные выборки для более точного краткосрочного прогноза.

Целью третьей серии экспериментов было определение рационального размера скользящего окна. Малый размер обеспечивает высокую чувствительность, но увеличивает риск ложных срабатываний. Большой размер окна эффективно подавляет случайные всплески, однако снижает оперативность обнаружения устойчивых перегрузок серверов.

Из-за низкого качества прогноза возможна ситуация, когда сервер, с которого была мигрирована ВМ, на следующем шаге наблюдения снова примет ВМ, т.е. возникнет обратная миграция.

В общем случае размер окна наблюдения должен быть больше длительности миграции, поскольку миграция вносит помехи в процесс наблюдения, дополнительно загружая процессор физического сервера и сеть.

Для определения рационального размера скользящего окна наблюдения введем коэффициент устойчивости $K_{уст}$, учитывающий вероятностный характер длительности миграции. Поскольку время миграции T является случайной величиной с функцией плотности распределения $f(t)$, использование среднего значения может приводить к недооценке необходимого размера окна. Более надежным подходом является выбор квантиля распределения.

Определим $t_{migration}$ как квантиль уровня 0,9 функции распределения времени миграции: $P(T \le t_{migration}) = \int_0^{t_{migration}} f(t)dt = 0,9.$

Тогда коэффициент устойчивости вычисляется как: $K_{уст} = t_{migration}/t_{window}$, где t_{window} — размер скользящего окна наблюдения. Поддержание $K_{уст} \le 1$ обеспечивает, что в 90% случаев миграция завершится в пределах текущего окна наблюдения, что минимизирует помехи, вносимые процессом миграции в мониторинг загрузки серверов.

Для определения плотности вероятности **в третьей главе** разработан метод расчета длительности миграции.

Для анализа временных характеристик процесса миграции использовался набор экспериментальных данных, включающий более 40 000 записей о выполнении живой миграций виртуальных машин. Исследовались пять различных алгоритмов живой миграции: пост-копирование (post-copy); предварительное копирование (pre-copy) в трех модификациях: с дросселированием процессора, с дельта-сжатием и со сжатием данных. Каждый из пяти алгоритмов представлен в выборке примерно 8000 наблюдениями.

Производительность живой миграции не только сильно различается для различных алгоритмов, но и сильно зависит от рабочих нагрузок, выполняемых внутри виртуальной машины.

Многие существующие работы для расчета общей длительности миграции используют средние значения. Однако в реальных системах важна предсказуемость и максимальная длительность для заданной доли миграции, так как именно это определяет качество обслуживания и соблюдение SLA-

соглашений. Ответ на вопрос какова вероятность получения того или иного значения длительности миграции можно получить, зная закон распределения вероятностей. Особенно это критично для приложений с высокой степенью случайности, вносящих наибольший разброс во время миграции.

В датасете представлено 9 типов таких приложений. Установлено, что тип приложения определяет как вид эмпирического распределения длительности миграции, так и его параметры, что требует получения аналитического закона распределения с учетом данного фактора.

Как наиболее распространенный, алгоритм pre-copy заслуживает отдельного рассмотрения. Длительность миграции с предварительным копированием складывается из этапов:

$$\tau = T_I + T_R + \sum_i T_{PCi} + T_{SC} + T_C + T_A$$

где T_I – время на инициализацию;

T_R – время на резервирование ресурсов;

T_{PCi} –время на предварительное копирование i-ой итерации;

T_{SC} –время на остановку и копирование;

T_C –время на завершение;

T_A время на активацию.

Время миграции складывается из детерминированной и случайной частей. Детерминированная (постоянная для данной системы) включает все операции кроме итерационного копирования. Случайная зависит от трех основных факторов: пропускной способности сети, объема памяти ВМ и скорости модификации страниц памяти.

Распределение длительности миграции и простоя соответствует в большинстве выборок логарифмически-нормальному и гамма распределениям. Для упрощения расчетов в работе предлагается аппроксимировать плотность распределения вероятностей рядами. Рассматривались ряды Грама-Шарлье и Лагерра. Показано, что ряды Грама-Шарлье не подходят для аппроксимации общей длительности миграции в явном виде, так как эмпирическая функция распределения имеет большой коэффициент асимметрии. Данные ряды можно использовать, если предварительно преобразовать длительность, взяв ее логарифм и нормировать случайную величину. Ряд Лагерра дает наилучшее приближение и может быть рекомендован для определения длительности миграции. Оценка адекватности аппроксимации рядом Лагерра проводилась с использованием критерия Колмогорова-Смирнова с уровнем значимости 0,05. Установлено, что для преобладающей части выборок расчетное значение статистики не превышает критического, что свидетельствует о согласии теоретического распределения с эмпирическими данными. Визуальное сопоставление (рис. 3) также подтверждает качество приближения.

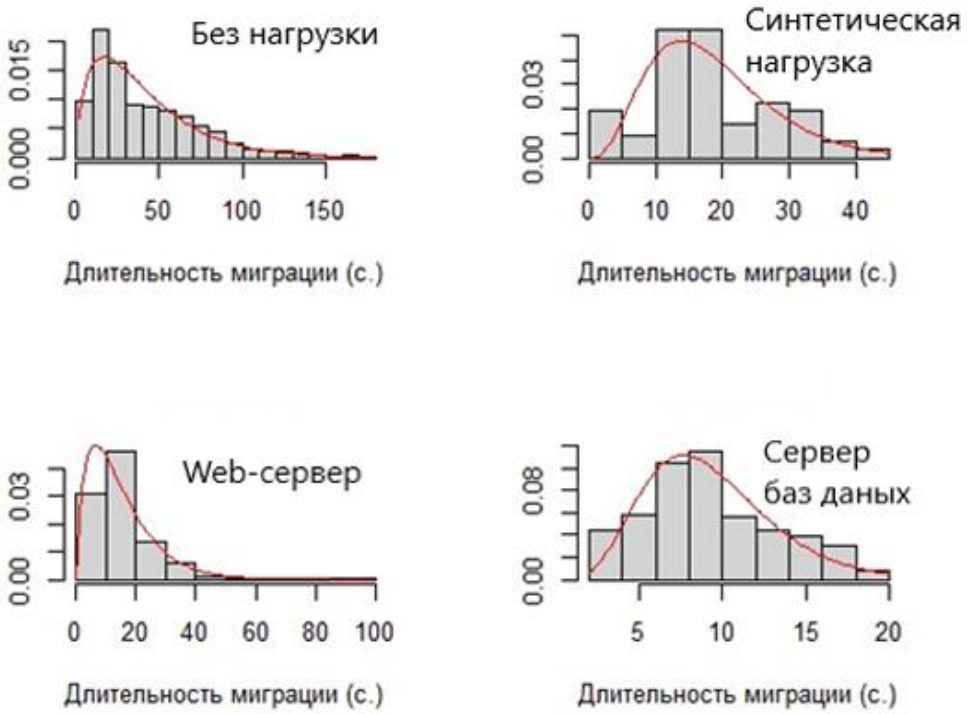

Рис. 3 — Аппроксимация плотности распределения вероятностей длительности миграции рядом Лагерра

Четвертая глава диссертации посвящена выбору серверов назначения для мигрирующих виртуальных машин.

Сами виртуальные машины для миграции предлагается выбирать на проблемных серверах исходя из минимального объема оперативной памяти, так как длительность миграции такой виртуальной машины будет минимальной.

Задача выбора серверов назначения сформулирована в виде линейной двухкритериальной задачи с критериями минимума неиспользуемых ресурсов и нарушения SLA-соглашений с бинарными переменными, означающими наличие или отсутствие виртуальной машины на сервере. Постановка задачи имеет следующий вид:

$$\sum_{i=1}^{N}\sum_{j=1}^{N} f_{res}^{j}(u_{ij}^{CPU}, u_{ij}^{RAM}) \cdot x_{ij} \to \min_{x}$$

,

$$\sum_{i=1}^{N}\sum_{j=1}^{N} f_{SLA}^{j}(u_{ij}^{CPU}) \cdot x_{ij} \to \min_{x},$$

при ограничениях:

$$\begin{cases} \sum_{j=1}^{N} x_{ij} = 1, \forall i \in \{1,2,\text{K}, N\}, \\ \sum_{i=1}^{N} x_{ij} = 1, \forall j \in \{1,2,\text{K}, N\}, \\ x_{ij} \in \{0,1\}, \forall i, j \in \{1,2,\text{K}, N\}. \end{cases}$$

В качестве метода решения многокритериальной задачи выбран метод линейной свертки критериев, который позволяет перейти от многокритериальной задачи к однокритериальной.

По своей структуре, системе ограничений и типу используемых переменных сформулированная задача относится к классу задач о назначении. Как известно, эффективным методом решения таких задач является венгерский алгоритм. Альтернативный подход к решению заключается в преобразовании исходной постановки к закрытой транспортной задаче, что достигается заменой условия целочисленности переменных на требование неотрицательности $x_{ij} \geq 0$.

Для оценки эффективности предложенного метода ВМ было проведено три серии экспериментов. Первая серия экспериментов направлена на сравнение многокритериального алгоритма с эвристическими алгоритмами задачи об упаковке в контейнеры, используемые на практике. Во второй серии экспериментов исследовалась вычислительная сложность алгоритма в зависимости от масштаба задачи. Для этих двух экспериментов моделирование выполнялось с использованием языка R.

В третьем наборе экспериментов оцениваются предлагаемые алгоритмы распределения ресурсов в крупномасштабной виртуализированной инфраструктуре ЦОД с использованием платформы имитационного моделирования CloudSim.

Таблица 3 содержит исходные данные для трех серий экспериментов. Для каждого набора параметров генерировались случайные входные данные. Каждый эксперимент повторялся 20 раз с последующим усреднением результатов.

Таблица 3 – Исходные данные для экспериментов

Серия экспериментов	**Размер ЦОД**
1. Эффективность размещения	50 ВМ, 50 серверов
2. Масштабируемость	50~250 ВМ и серверов
3. Имитационное моделирование крупного ЦОД	1000 ВМ, 800 серверов

В первой серии экспериментов проводилось сравнение предложенного алгоритма с эвристиками BFD и FFD. При использовании метода свертки весовые коэффициенты последовательно принимали значения от (0,1; 0,9) до (0,9; 0,1) с шагом 0,2.

После нормализации показателей относительно их экстремальных значений построены графики для критериев SLA и эффективности использования ресурсов (рис. 4, 5). Сравниваются алгоритмы bfd_cpu, bfd_mem, ffd_cpu, ffd_mem и предложенный метод с весами α_1, α_2 = 0,1; 0,3; 0,5; 0,7; 0,9.

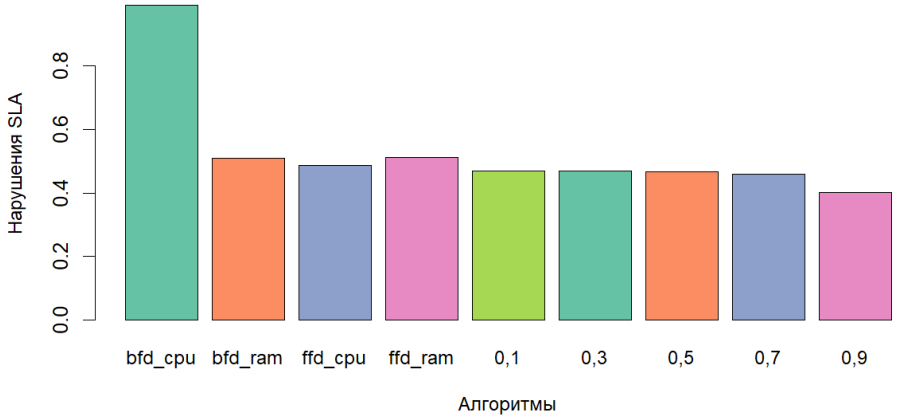

Рис. 4 — Нормализованные значения критерия нарушения SLA-соглашений для различных алгоритмов

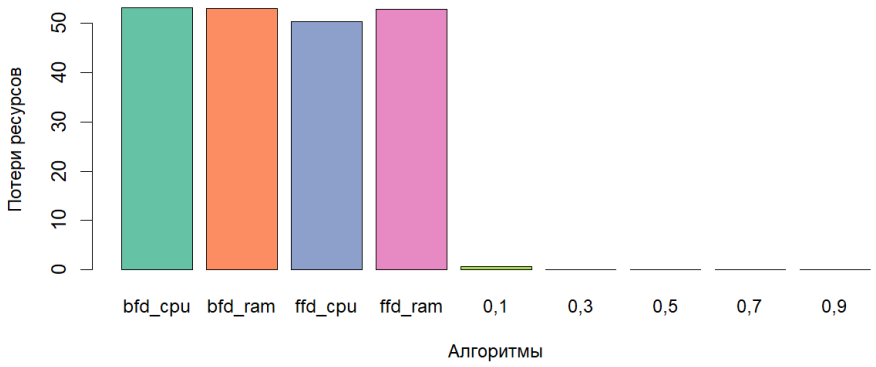

Рис. 5 — Нормализованные значения критерия неэффективности использования ресурсов

Рисунок 4 показывает, что алгоритм bfd_cpu, обеспечивая максимальную плотность размещения, дает и наибольший уровень нарушений SLA. Однокритериальные эвристики (bfd_cpu, bfd_mem, ffd_cpu, ffd_mem) неэффективно используют разнотипные ресурсы. В отличие от них, обобщённый критерий позволяет получать решения, сбалансированные по двум критериям.

Во второй серии экспериментов исследовалась скорость работы алгоритма при различной размерности задачи. Использовался компьютер с

процессором Pentium(R) Dual-Core CPU E5700 3 ГГц и 4 ГБ ОЗУ. Временные характеристики представлены в таблице 4.

Таблица 4 – Зависимость времени решения от размерности задачи

Размер ЦОД	**Время решения симплекс-методом (с.)**	**Время решения венгерским методом (с.)**
50 ВМ, 50 серверов	0,73	0,37
100 ВМ, 100 серверов	1,72	0,41
150 ВМ, 150 серверов	4,84	0,57
200 ВМ, 200 серверов	11,44	0,88
250 ВМ, 250 серверов	22,17	1,16
500 ВМ, 500 серверов	Не хватает памяти	3.46
1000 ВМ, серверов	Не хватает памяти	11,89
1500 ВМ, серверов	Не хватает памяти	33,87

Эмпирическая оценка вычислительной сложности симплекс-метода демонстрирует квадратичную зависимость времени решения от размерности задачи $O(n^2)$. Для венгерского метода вычислительная сложность составляет $O(n^3)$, но в целом венгерский метод быстрее решает задачу и экономичнее использует память.

Также проведена серия экспериментов на платформе имитационного моделирования CloudSim. Всего было смоделировано 800 хостов и 1000 виртуальных машин. Весовые коэффициенты изменялись от 0,1 до 0,9 с шагом 0,1. Результаты сравнивались с FFD. Проведено 100 экспериментов, один из которых показан на рис. 6.

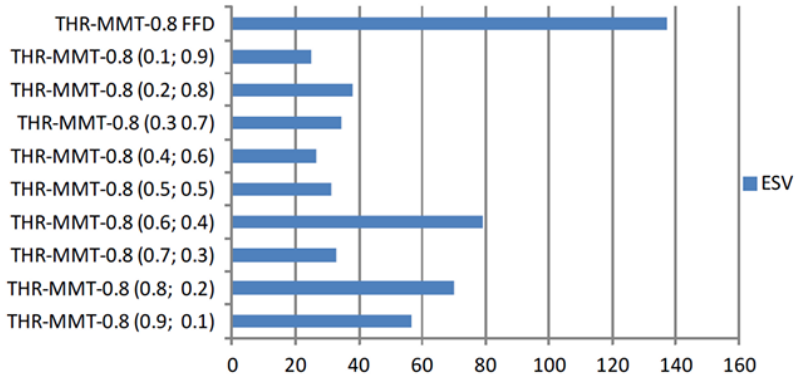

Рис. 6 — Значения комбинированного критерия ESV

Результаты моделирования показывают, что весовые коэффициенты частных критериев оптимизации в целом влияют на результат. Чем больше весовой коэффициент энергопотребления, тем меньше показатель *E*. Аналогично, чем больше значение весового коэффициента для нарушения SLA, тем меньше показатель *SLAV*.

Во всех тестовых сценариях предложенный алгоритм размещения ВМ превзошел FFD как по частным критериям, так и по интегральной метрике ESV. Моделирование показало, что выигрыш по комбинированному

показателю энергоэффективности и качества обслуживания (ESV) составляет в среднем 3,5 раза.

Пятая глава содержит описание экспериментальной проверки и практической реализации распределенного планировщика ресурсов для облачного ЦОД. Основное внимание уделено имитационному моделированию: представлено детальное описание разработанных имитационных моделей динамического и статического размещения виртуальных машин. Приведены алгоритмы обнаружения недогрузки и перегрузки хоста, выбора и размещения ВМ. Разработаны предложения по реализации методов распределения ресурсов.

Архитектура системы управления ресурсами облачного ЦОД является двухуровневой, состоящей из локальных и глобального контроллеров (рис. 7).

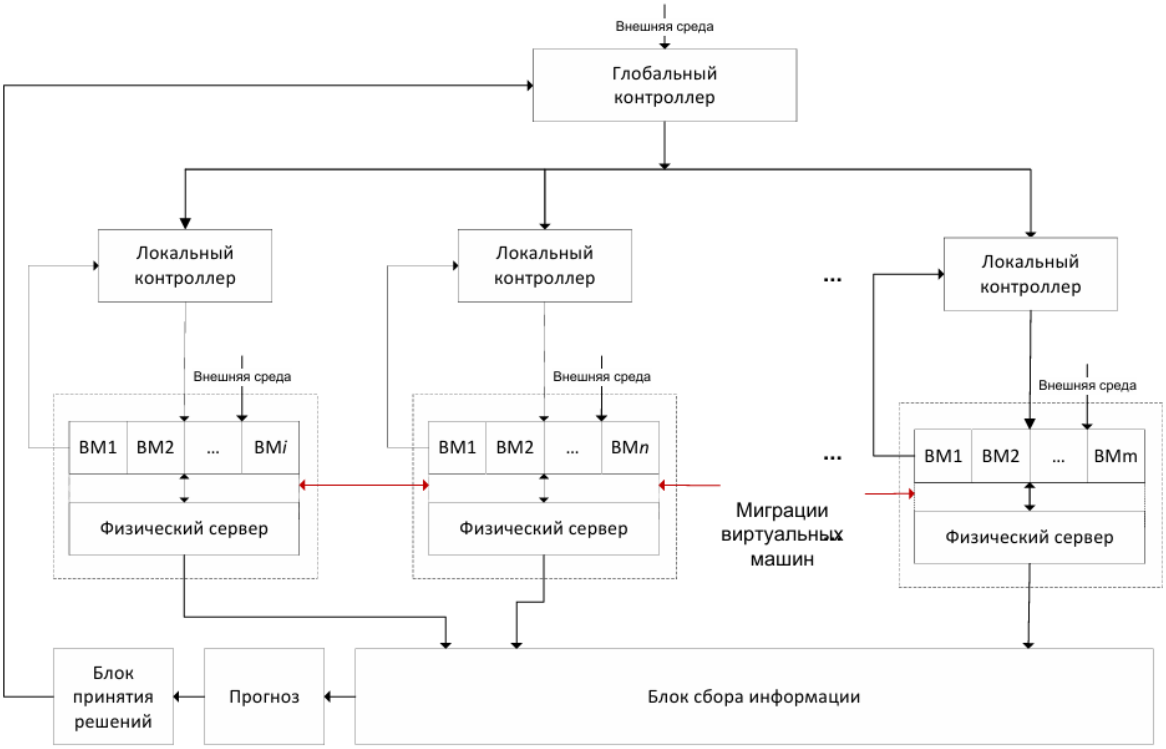

Рис. 7 — Архитектура программно-аппаратного комплекса планировщика ресурсов облачного ЦОД

Первоначальное размещение осуществляется **глобальным контроллером**, у которого имеется доступ к глобальному хранилищу данных, в котором содержится информация о текущем использовании ресурсов физических серверов и их видеокарт. Размещение производится в соответствии с методом первоначального размещения, предложенном во второй главе.

На каждом физическом сервере развернут **локальный контроллер** — специализированный модуль монитора виртуальных машин. В его обязанности входит непрерывное отслеживание загрузки CPU и отправка полученных показателей в центральное хранилище, поддерживаемое глобальным контроллером (**блок сбора информации**). Размера окна наблюдения вычисляется с помощью коэффициента устойчивости, для

которого рассчитывается длительность миграции с заданной вероятностью, метод оценки которой изложен в третьей главе.

Глобальный контроллер анализирует данную информацию, делает прогноз методом группового учета аргументов на следующие 1-2 окна наблюдения (**блок прогноза**) и в соответствии с ней принимает решения о перемещении виртуальных машин (**блок принятия решений**). Это предотвращает ситуации, требующие обратной миграции или повторного выбора сервера, что повышает устойчивость всего процесса.

Глобальный контроллер отвечает за выбор ВМ для миграции и определение серверов назначения. При отборе кандидатов учитываются объем памяти и интенсивность модификации страниц — чем они меньше, тем предпочтительнее миграция. Размещение выбранных ВМ выполняется путем решения оптимизационной задачи с аддитивным критерием (см. главу 4).

Также в пятой главе разработаны рекомендации по интеграции разработанных алгоритмов в открытую облачную платформу OpenStack.

В **заключении** приведены основные научные результаты работы:

1. Проанализированы проблемы повышения эффективности функционирования гетерогенных центров обработки, среди которых выделены недостаточная энергоэффективность инфокоммуникационной системы гетерогенного ЦОД и стабильность облачных сервисов.
2. Разработаны критерии и метод многокритериального первоначального размещения разнородных виртуальных машин на физических серверах с поддержкой GPU-вычислений. В основе данного метода лежит многокритериальная задача оптимизации с критериями энергопотребления, использования ресурсов, нарушения SLA-соглашения. Показано, что решение задачи муравьиным алгоритмом с использованием обобщённого критерия позволяет достичь баланса между данными противоречивыми критериями по сравнению с эвристическими алгоритмами размещения.
3. Исследовано применение метода группового учета аргументов для прогнозирования загрузки физических серверов на реальных данных, дающего более точный прогноз по сравнению с известными классическими статистическими методами, что подтверждено имитационным моделированием.
4. Разработан критерий определения рационального размера окна наблюдения за серверами для чего исследованы характеристики процесса миграции виртуальных машин и предложен метод расчета вероятности длительности миграции. Предложенный подход позволяет дать оценку вероятности миграции заданной длительности при определении размера скользящего окна.
5. Разработан метод многокритериального динамического размещения виртуальных машин, который позволяет найти лучшее размещение по

комбинированному критерию энергопотребления и нарушения SLA-соглашений за приемлемое время.

6. Разработаны имитационные модели распределения ресурсов облачного ЦОД, которые позволяют проверить эффективность разработанных методов и алгоритмов динамического распределения ресурсов.
7. Разработан программный модуль для прогнозирования загрузки серверов в системе имитационного моделирования облачных центров обработки данных CloudSim, который может быть встроен в монитор виртуальных машин (гипервизор). Также программный модуль может использоваться как реализация комбинаторного метода группового учета аргументов на языке Java.
8. Разработаны алгоритмы и практические рекомендации по реализации программно-аппаратного комплекса распределенного планировщика ресурсов для инфокоммуникационных систем облачных ЦОД. Предложенные решения ориентированы на интеграцию в отечественные облачные платформы.

Публикации автора по теме диссертации

В изданиях из списка ВАК РФ

1. **Тутов А.В.** Метод первоначального размещения виртуальных машин с графическим процессором в гетерогенных центрах обработки данных // Нейрокомпьютеры: разработка, применение. 2025. Т. 27. № 5. С. 5–16.
2. **Тутов А.В.**, Фархадов М.П. Метод и алгоритм статического размещения виртуальных машин для повышения эффективности функционирования инфокоммуникационной системы центров обработки данных // Нейрокомпьютеры: разработка, применение. 2024. Т. 26. № 5. С. 107-119.
3. **Тутов А.В.**, Таратухин А.В., Керимов С.С. Метод оценки характеристик процесса миграции виртуальных машин с учетом типа миграций и приложений // Динамика сложных систем. 2024. Т. 18. № 3. С. 48−59.
4. **Тутов А.В.**, Фархадов М.П., Таратухин А.В., Программно-аппаратный комплекс распределенного планировщика ресурсов инфокоммуникационной системы облачного центра обработки данных // Управление большими системами. Выпуск 109. М.: ИПУ РАН, 2024. С.268-292.
5. **Тутов А.В.**, Тутова Н.В., Ворожцов А.С., Андреев И.А. Многокритериальная оптимизация размещения виртуальных машин по физическим серверам в облачных центрах обработки данных // T-Comm: Телекоммуникации и транспорт. 2021. Том 15. №1. С. 28-34.
6. **Тутов А.В.** Модели и методы распределения ресурсов инфокоммуникационной системы облачных центров обработки данных // Наукоемкие технологии в космических исследованиях Земли. 2018. Т. 10. № 6. С. 100–107. (Англ.)

7. **Тутов А. В.** Метод расчета длительности миграции виртуальных машин в системах управления // Датчики и системы. 2018. № 7. С. 15—20.
8. **Тутов А.В.**, Тутова Н.В., Ворожцов А.С. Моделирование процессов распределения ресурсов в облачных центрах обработки данных // T-Comm–Телекоммуникации и транспорт. 2017. Т. 11, № 4. С. 76—80.
9. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Динамическое распределение вычислительных ресурсов центров обработки данных // T-Comm–Телекоммуникации и транспорт. 2016. Т. 10, № 7. С. 47—51.
10. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Оптимизация размещения облачных серверов в центрах обработки данных // T-Comm: Телекоммуникации и транспорт. 2015. Т. 9, № 6. С. 4—8.
11. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Методика оптимального распределения виртуальных серверов в центрах обработки данных // T-Comm-Телекоммуникации и Транспорт. 2015. Т. 9, № 7. С. 5—10.
12. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Оценка производительности облачных центров обработки // T-Comm–Телекоммуникации и Транспорт. 2014. Т. 8, № 5. С. 69—71.
13. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Управление вычислительными ресурсами центров обработки данных с учётом поведения пользователей коммерческих сайтов // T-Comm–Телекоммуникации и транспорт. 2012. Т. 6, № 10. С. 46—47.

В Scopus и Web of Science

14. **Toutov A.V.**, Toutova N.V., Bulanov G.A., Frolova E.A., Andreev I.A. Resource Allocation Algorithms for Single, Cluster and Tired Virtual Machines // 2023 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED). IEEE. 2023. С. 1—4. 14. (Англ.)
15. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S., Andreev I.A. Optimizing the Migration of Virtual Machines in Cloud Data Centers // International Journal of Embedded and Real-Time Communication Systems (IJERTCS). 2022. Т. 13, № 1. С. 1—19. (Англ.)
16. Toutova N.V., Erokhin A.G., **Toutov A.V.**, Vanina M.F., Frolova E.A. Principals of Optimal Cloud Resource Management // 2022 Intelligent Technologies and Electronic Devices in Vehicle and Road Transport Complex (TIRVED). IEEE. 2022. С. 1—7. (Англ.)
17. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S. Analysis of Data Center Development Problems in the Era of Digital Transformation // International Workshop on Model-Driven Organizational and Business Agility. Springer. 2022. С. 65—72. (Англ.)
18. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S., Andreev I.A. Multicriteria Optimization of Virtual Machine Placement in Cloud Data Centers // 28th Conference of Open Innovations Association (FRUCT). IEEE. 2021. С. 482—487. (Англ.)

19. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S. Analytical Approach to Estimating Total Migration Time of Virtual Machines With Various // International Journal of Embedded and Real-Time Communication Systems (IJERTCS). 2020. Т. 11, № 2. С. 58—75. (Англ.)
20. **Toutov A.V.,** Toutova N.V., Vorozhtsov A.S. Resource control system stability of mobile data centers // 2018 Systems of Signals Generating and Processing in the Field of on Board Communications. 2018. С. 1—4. (Англ.)
21. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S. Estimation of Total Migration Time of Virtual Machines in Cloud Data Centers // Proceedings of the 2018 IEEE International Conference "Quality Management, Transport and Information Security, Information Technologies". 2018. С. 389—393. (Англ.)

В прочих изданиях

22. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Об основном цикле вычислительных работ по поддержанию стабильности и качества облачных сервисов // Телекоммуникационные и вычислительные системы. Труды конференции. — Международная академия информатизации; Московский технический университет связи и информатики. 2022. С. 723-732.
23. **Тутов А.В.** Облачная платформа 1С в сетях центров обработки данных. / Сборник научных трудов XXII международной научно-практической конференции «Новые информационные технологии в образовании». Под общей редакцией Д.В. Чистова. Москва, 2022. С. 308-309.
24. **Toutov A.V.**, Toutova N.V., Vorozhtsov A.S. The Method of Calculation the Total Migration Time of Virtual Machines in Cloud Data Centers // Proceedings of the 24th Conference of Open Innovations Association FRUCT. FRUCT Oy. 2019. P. 110.
25. **Тутов А. В.**, Тутова Н. В., Ворожцов А. С. Indicators of Stability of the Cloud Data Center Resource Management System // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2018): Материалы XXI Международной научной конференции / под ред. К. Е. Самуйлова. Москва: РУДН, 09.2018. С. 121—126. (Англ.)
26. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Управление живой миграцией // Телекоммуникационные и вычислительные системы. Труды конференции. Международная академия информатизации; Московский технический университет связи и информатики. 10.2017. С. 229.
27. **Тутов А.В.**, Тутова Н.В., Ворожцов А.С. Live Migration Management // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2017): Материалы Двадцатой международной научной конференции / под ред. В. М. Вишневского. Рекламно-издательский центр "ТЕХНОСФЕРА"(Москва), 09.2017. С. 440—443. (Англ.)

28. Майоров А.А., **Тутов А.В.**, Тутова Н.В. Динамическое размещение виртуальных машин в облачных центрах обработки данных // 70-я региональная научно-техническая конференция студентов, аспирантов и молодых ученых "Студенческая весна - 2016": сб.науч.ст. / под ред. К. В. Дукельского. СПб: СПбГУТ, 2016. С. 121—126.
29. **Тутов А.В.**, Тутова Н.В., Ворожцов А.С. Dynamic multi-criteria virtual machine allocation in cloud data centers // Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь (DCCN-2016) материалы Девятнадцатой международной научной конференции: в 3 томах. / под ред. В. М. В. и К. Е. Самуйлова. Москва: РУДН, 10.2016. С. 174—182. (Англ.)
30. Ворожцов А.С., Тутова Н.В., **Тутов А.В.** Принцип индивидуальности в управлении вычислительными ресурсами центров обработки данных // Телекоммуникационные и вычислительные системы. Труды конференции. Международная академия информатизации; Московский технический университет связи и информатики. ООО "Брис-М", 10.2015. С. 229.

Свидетельство о регистрации программы для ЭВМ

31. **Тутов А.В.**, Тутова Н.В., Ворожцов А.С. Программа для прогнозирования перегрузки серверов с использованием комбинаторного метода группового учета аргументов на языке программирования Java: свидетельство о регистрации программы для ЭВМ RUS 2018666780. 07.12.2018.
32. **Тутов А.В.** Программа для динамического планирования размещения виртуальных машин по физическим серверам в центрах обработки данных: свидетельство о регистрации программы для ЭВМ №2025693891. 02.12.2025.
33. Фархадов М.П., **Тутов А.В.** Программа для первоначального размещения виртуальных машин в гетерогенном центре обработки данных: свидетельство о регистрации программы для ЭВМ №2025697058. 22.12.2025.