

УДК 004.4'23

DOI: [10.26102/2310-6018/2025.50.3.024](https://doi.org/10.26102/2310-6018/2025.50.3.024)

Применение гибридного алгоритма балансировки для управления распределением вычислительных задач в высоконагруженных системах

В.С. Дойчев✉

Сибирский федеральный университет, Красноярск, Российская Федерация

Резюме. Актуальность исследования обусловлена стремительным ростом объемов обрабатываемых данных и широким распространением облачных технологий, что делает эффективное распределение вычислительных задач в высоконагруженных системах ключевой проблемой современной информатики. Существующие методы балансировки нагрузки часто не учитывают гетерогенность ресурсов, динамику workloads и необходимость многоцелевой оптимизации, что ограничивает их эффективность. Целью работы является разработка гибридного алгоритма балансировки нагрузки, сочетающего преимущества алгоритмов Artificial Bee Colony (ABC) и Max-Min для повышения производительности и ресурсоэффективности распределенных систем. В исследовании использован метод имитационного моделирования в среде CloudSim для оценки предложенного алгоритма при различных сценариях нагрузки (от 100 до 5000 задач). Задачи классифицируются на «легкие» и «тяжелые» на основе их вычислительной сложности (MIPS), после чего ABC применяется для быстрого распределения простых задач, а Max-Min – для оптимизации выполнения ресурсоемких заданий с целью минимизации общего времени выполнения (makespan). Сравнительный анализ с базовыми алгоритмами (FCFS, SJF, Min-Min, Max-Min, PSO, ABC) показал, что гибридный подход обеспечивает на 15–30 % лучшее время выполнения задач при высокой нагрузке (5000 задач), демонстрируя высокую адаптивность и масштабируемость. Результаты исследования подтверждают, что гибридные алгоритмы, объединяющие эвристические и метаэвристические методы, представляют собой перспективное решение для динамических облачных сред. Предложенный метод эффективно сочетает оперативность распределения легких задач и стратегическое планирование ресурсоемких операций, что делает его применимым в реальных ЦОДах и распределенных системах. Практическая значимость работы заключается в повышении энергоэффективности, снижении затрат и обеспечении качества обслуживания (QoS) в облачных вычислениях.

Ключевые слова: облачные вычисления, планирование, распределение задач, виртуальные машины, гибридный алгоритм, балансировка нагрузки, оптимизация, CloudSim.

Для цитирования: Дойчев В.С. Применение гибридного алгоритма балансировки для управления распределением вычислительных задач в высоконагруженных системах. *Моделирование, оптимизация и информационные технологии*. 2025;13(3). URL: <https://moitvvt.ru/ru/journal/pdf?id=1998> DOI: 10.26102/2310-6018/2025.50.3.024

Application of a hybrid load balancing algorithm for managing the distribution of computational tasks in high-load systems

V.S. Doichev✉

Siberian Federal University, Krasnoyarsk, the Russian Federation

Abstract. The growing volume of processed data and the widespread adoption of cloud technologies have made efficient task distribution in high-load computing systems a critical challenge in modern computer science. However, existing solutions often fail to account for resource heterogeneity, dynamic workload variations, and multi-objective optimization, leaving gaps in achieving optimal resource utilization. This study aims to address these limitations by proposing a hybrid load-balancing algorithm

that combines the strengths of Artificial Bee Colony (ABC) and Max-Min scheduling strategies. The research employs simulation in the CloudSim environment to evaluate the algorithm's performance under varying workload conditions (100 to 5000 tasks). Tasks are classified into "light" and "heavy" based on their MIPS requirements, with ABC handling lightweight tasks for rapid distribution and Max-Min managing resource-intensive tasks to minimize makespan. Comparative analysis against baseline algorithms (FCFS, SJF, Min-Min, Max-Min, PSO, and ABC) demonstrates the hybrid approach's superior efficiency, particularly in large-scale and heterogeneous environments. Results show a 15–30% reduction in average task completion time at high loads (5000 tasks), confirming its adaptability and scalability. The study concludes that hybrid algorithms, integrating heuristic and metaheuristic techniques, offer a robust solution for dynamic cloud environments. The proposed method bridges the gap between responsiveness and strategic resource allocation, making it viable for real-world deployment in data centers and distributed systems. The practical significance of the work lies in increasing energy efficiency, reducing costs and ensuring quality of service (QoS) in cloud computing.

Keywords: cloud computing, scheduling, task allocation, virtual machines, hybrid algorithm, load balancing, optimization, CloudSim.

For citation: Doichev V.S. Application of a hybrid load balancing algorithm for managing the distribution of computational tasks in high-load systems. *Modeling, Optimization and Information Technology*. 2025;13(3). (In Russ.). URL: <https://moitvvt.ru/ru/journal/pdf?id=1998> DOI: 10.26102/2310-6018/2025.50.3.024

Введение

В условиях стремительного роста объемов обрабатываемых данных и распространения облачных технологий управление распределением вычислительных задач в высоконагруженных системах становится одной из ключевых проблем современной информатики. Повышенные требования к производительности, отказоустойчивости, энергоэффективности и стоимости эксплуатации вычислительных ресурсов требуют использования интеллектуальных и гибких подходов к планированию и балансировке нагрузки. В частности, особую актуальность приобретают исследования, направленные на оптимизацию размещения задач с учетом ограниченности ресурсов, гетерогенности инфраструктуры и динамики нагрузки.

В статье [1] описывается использование CloudSim для симуляции и анализа предложенных методов оптимизации виртуальных машин. Автор рассматривает применение алгоритма прогнозируемого планирования для балансировки нагрузки в целях эффективного управления вычислительными ресурсами. Также автор сравнивает предложенный им метод с существующими алгоритмами Non-Dominated Sorting Genetic Algorithm (NSGA-II и NSGA-III). В работе отмечается, что облачные сервисы стали востребованным решением в крупных дата-центрах для энергоэффективного распределения ресурсов и повышения их использования. Большинство существующих решений учитывают лишь ограниченное количество категорий ресурсов, что приводит к разбалансировке нагрузки или избыточной активации отдельных физических серверов. В связи с этим был предложен и проанализирован алгоритм размещения виртуальных машин на основе метода bin padding, который учитывает сбалансированное использование нескольких видов ресурсов.

В исследовании [2] авторы решают задачу управления виртуальными машинами (ВМ) через модель марковских процессов принятия решений (MDP). Этот подход, основанный на реальных данных, обеспечивает оптимизацию распределения ресурсов. Однако при масштабировании возникает проклятие размерности из-за экспоненциального роста числа состояний. Для его преодоления предложен алгоритм MadVM, сочетающий приближенные методы MDP с минимальными затратами на

миграцию ВМ. Авторы доказали его сходимость и оценили границы погрешности, выделив применимость в распределенных системах.

Работа [3] фокусируется на интеллектуальном размещении ВМ в ЦОД. Ключевыми критериями стали энергоэффективность, отказоустойчивость и безопасность. Использование прогнозного управления (MPC) позволило динамически адаптировать распределение ресурсов, учитывая изменения нагрузки. Особый акцент сделан на анализе использования памяти ВМ для повышения производительности.

В статье [4] представлен алгоритм Кукушки (CA), вдохновленный биологическим поведением. Его преимущество – в гибкой адаптации к колебаниям нагрузки, что подтверждено сравнением с классическим Round Robin.

Исследование [5] анализирует энергоэффективность планирования задач в CloudSim. Рассмотрены модели облаков (частные, публичные, гибридные, коммунальные) и уровни сервисов (IaaS, PaaS, SaaS), подчеркивая важность управления ресурсами для разных сценариев.

В ряде исследований рассматриваются различные подходы к гибридизации методов планирования в облачных средах для повышения эффективности использования ресурсов и минимизации задержек. Одним из таких примеров является работа [6], в которой предложен новый алгоритм планирования заданий – SG-PBFS. Он сочетает стратегию приоритезации задач на основе алгоритма PBFS (Priority-Based Fair Scheduling) с подходом размещения задач в кратчайшие свободные интервалы времени (Shortest Gap), что позволяет более рационально использовать временные слоты в расписании облачных заданий.

В исследовании [7] представлен гибридный многоцелевой метод балансировки нагрузки для облачных вычислительных платформ, который направлен на оптимизацию распределения виртуальных машин (VM) по физическим серверам (PM) с учетом множества критериев. В частности, учитывается использование памяти, затраты на миграцию виртуальных машин, энергопотребление, а также параметры балансировки нагрузки, такие как время отклика, общее время выполнения задач и загрузка серверов. Для выбора оптимального распределения используется новый гибридный алгоритм, основанный на концептуальном объединении двух методов – Cat and Mouse-Based Optimizer (CMBO) и Standard Dingo Optimizer (DXO), что дало начало методу Dingo Customized Cat Mouse Optimization (DCCO).

Данный подход обеспечивает значительное улучшение показателей нагрузки на серверы, снижая ее в исследуемой среде до уровня, превосходящего по эффективности многие существующие алгоритмы, такие как MOA, ABC, CSO, SSO, SSA, ACSO, SMO, CMBO, BOA, DOX и FF-PSO. Оценка метода DCCO проводилась по комплексному набору метрик, включающему время выполнения задач (makespan), использование памяти, затраты на миграцию, время отклика, энергопотребление, загрузку серверов, общее время обработки, пропускную способность и скорость сходимости алгоритма. Результаты показывают, что предложенная гибридная модель является мощным инструментом для эффективного балансирования нагрузки в облачных IaaS-средах, способствуя повышению производительности и снижению эксплуатационных затрат.

В статье [8] представлен метод планирования задач в реальном времени на облачных платформах, который направлен на обеспечение энергосбережения, соблюдения требований качества обслуживания (QoS) и снижения затрат. Авторы используют динамическое управление напряжением и частотой (DVFS) на уровне отдельных ядер многоядерных гетерогенных процессоров, а также применяют методы приближенных вычислений для заполнения временных промежутков в расписании. При этом учитывается влияние ошибок во входных данных на время обработки отдельных задач.

Основная цель предложенного подхода заключается в достижении своевременного выполнения задач и энергоэффективности за счет разумного компромисса между точностью результата и его качеством, при этом удерживая стоимость выполнения в приемлемых пределах. Для оценки эффективности разработанного эвристического алгоритма его сравнили с двумя базовыми методами планирования под различными требованиями QoS.

Результаты моделирования показали, что предложенный метод превосходит альтернативные подходы, демонстрируя высокую эффективность и потенциал для использования в облачных системах, ориентированных на энергосбережение и экономичность при обеспечении высокого уровня качества обслуживания.

В работе [9] рассматривается важная задача балансировки между временем выполнения рабочих процессов и затратами на использование ресурсов в облачных вычислениях. Современные бизнесы все чаще переходят на облачные платформы для развертывания своих рабочих процессов, однако эффективное планирование задач в таких системах затрудняется высокой вычислительной сложностью, взаимозависимостью задач и гетерогенностью облачных ресурсов. В процессе распределения ресурсов возникают значительные проблемы, связанные с увеличением времени выполнения и ростом затрат, что может негативно сказываться на качестве обслуживания пользователей.

Авторы исследуют многокритериальную задачу оптимизации, в которой требуется одновременно минимизировать время выполнения (makespan) и стоимость использования виртуальных машин. Для решения этой задачи они предлагают гибридный метод HEFT-ACO, который сочетает алгоритм Heterogeneous Earliest Finish Time (HEFT) с муравьиным алгоритмом (Ant Colony Optimization, ACO). В ходе экспериментов, проведенных на трех типах научных рабочих процессов с учетом характеристик платформы Amazon EC2, предложенный алгоритм показал лучшие результаты по сравнению с базовым ACO, PEFT-ACO и FR-MOS, демонстрируя эффективный компромисс между скоростью выполнения и затратами на ресурсы в облачной среде.

В статье [10] рассматривается инновационный подход к планированию научных рабочих процессов в облачных средах, основанный на комбинации генетического алгоритма с эвристикой HEFT (Heterogeneous Earliest Finish Time). Авторы акцентируют внимание на критической важности оптимального распределения взаимозависимых задач между доступными вычислительными ресурсами для достижения двух ключевых целей: минимизации общего времени выполнения рабочего процесса и сокращения вычислительных затрат при строгом соблюдении временных и бюджетных ограничений. Особенность предложенного гибридного подхода заключается в использовании генетического алгоритма, где начальная популяция решений формируется с применением эвристики HEFT, что существенно ускоряет процесс поиска оптимальных решений и повышает их качество.

Практическая реализация алгоритма выполнена в виде специализированного модуля, интегрированного в симулятор WorkflowSim, что обеспечило удобство тестирования и подтвердило практическую применимость метода. Проведенные эксперименты с использованием реальных научных рабочих процессов продемонстрировали явное преимущество гибридного подхода по сравнению как с классической эвристикой HEFT, так и с другими анализируемыми стратегиями. Полученные результаты свидетельствуют о высокой эффективности предложенного решения и его значительном потенциале для внедрения в реальные облачные инфраструктуры.

Данное исследование убедительно доказывает, что гибридные методы, сочетающие классические эвристики с эволюционными алгоритмами, способны существенно улучшить качество планирования и повысить общую эффективность выполнения сложных научных рабочих процессов в облачных вычислительных платформах. При этом как статические, так и динамические методы балансировки нагрузки обладают определенными ограничениями, которые наиболее эффективно преодолеваются путем комбинирования различных подходов с метаэвристическими алгоритмами.

Проведенный анализ свидетельствует о том, что гибридные алгоритмы, объединяющие различные эвристические и метаэвристические подходы, представляют собой наиболее перспективное направление для оптимизации распределения вычислительных ресурсов. Такие решения демонстрируют способность адаптироваться к быстро меняющимся условиям работы, сохраняя при этом высокий уровень производительности и устойчивости системы.

Классические методы планирования, несмотря на их простоту и низкие накладные расходы, постепенно уступают место более совершенным алгоритмам, способным учитывать множество взаимосвязанных факторов.

Дальнейшее совершенствование гибридных методов открывает новые возможности для повышения производительности облачных платформ, снижения операционных затрат и оптимизации энергопотребления. Развитие в этом направлении будет способствовать созданию более устойчивых и эффективных вычислительных систем, способных удовлетворять растущие потребности современного цифрового общества.

Материалы и методы

Разрабатываемый гибридный алгоритм будет состоять из двух подалгоритмов.

Данный алгоритм разбивает подаваемые задачи на две очереди в зависимости от параметра MIPS. Более легкие задачи балансируются с помощью алгоритма ABC, более сложные – MaxMin.

Основные шаги алгоритма:

- получение параметра MIPS поступающих задач;
- расчет 80 перцентиля времени выполнения поступивших задач;
- разделение выборки задач на две, согласно рассчитанному значению на шаге 2;
- вызов алгоритма ABC для более легких задач и алгоритма MaxMin для более сложных задач.

Для сравнения гибридного алгоритма с базовыми на основе анализа литературы и научных статей были разработаны различные типы алгоритмов:

- алгоритмы балансировки нагрузки: Min-Min и Max-Min;
- алгоритмы управлением распределением задач: FCFS и SJF;
- эвристические алгоритмы балансировки: PSO и ABC.

Разработка и моделирование экспериментов проводилось в среде CloudSim¹.

Для проведения экспериментов было 5 сценариев, в которых меняется количество подаваемых задач и способность их обработки в секунду.

Количество подаваемых задач ранжируется от 100 до 1000. Количество обрабатываемых задач в секунду – от низкой (от 500 до 600) до высокой (от 100 до 1000). В Таблице 1 представлено подробное описание моделируемой среды.

¹ What is CloudSim? GeeksforGeeks. URL: <https://www.geeksforgeeks.org/what-is-cloudsim/> (дата обращения: 04.06.2025).

Таблица 1 – Характеристики моделируемой среды
Table 1 – Simulated environment characteristics

VM ID	CPU	RAM (MIB)	Storage (GB)	Bandwidth (MBps)
1	2	4096	100	1000
2	3	8192	200	2000
3	3	16384	500	5000

Результаты

На Рисунке 1 представлен график сравнения работы гибридного алгоритма с базовыми.

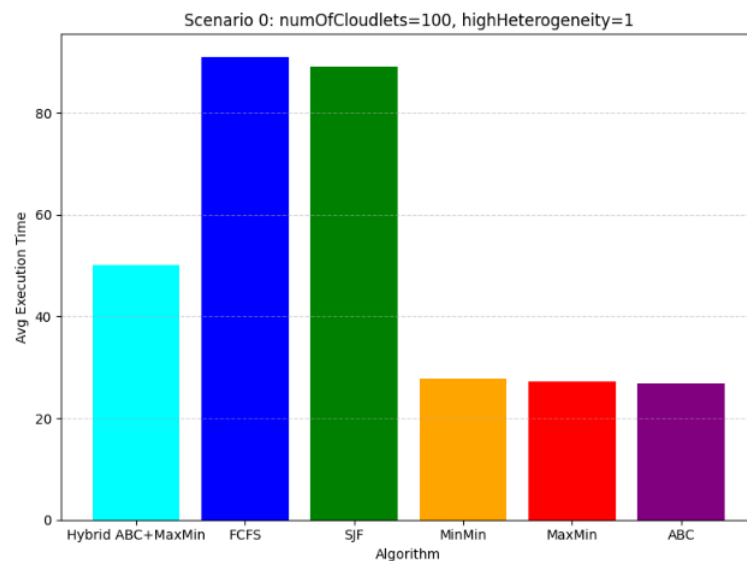


Рисунок 1 – Сценарий с 100 задачами
Figure 1 – Scenario with 100 tasks

На Рисунке 2 представлен график сравнения работы гибридного алгоритма с базовыми со сценариями с 200 задачами.

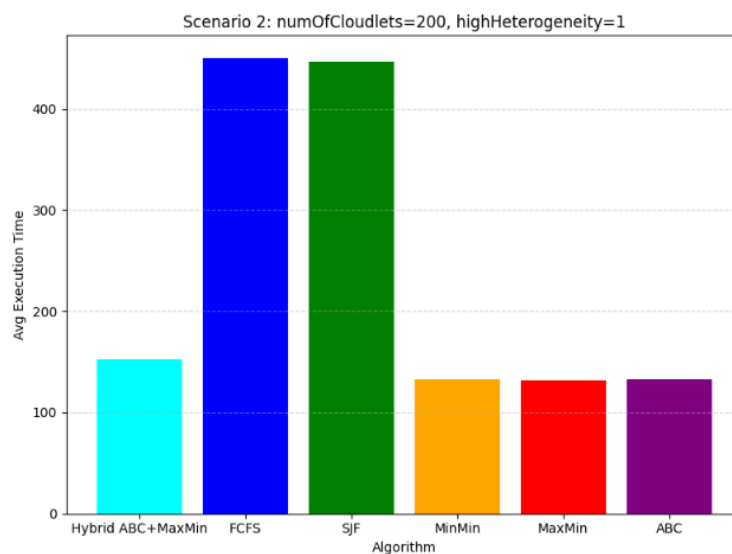


Рисунок 2 – Сценарий с 200 задачами
Figure 2 – Scenario with 200 tasks

На Рисунке 3 представлен график сравнения работы гибридного алгоритма с базовыми со сценариями с 400 задачами.

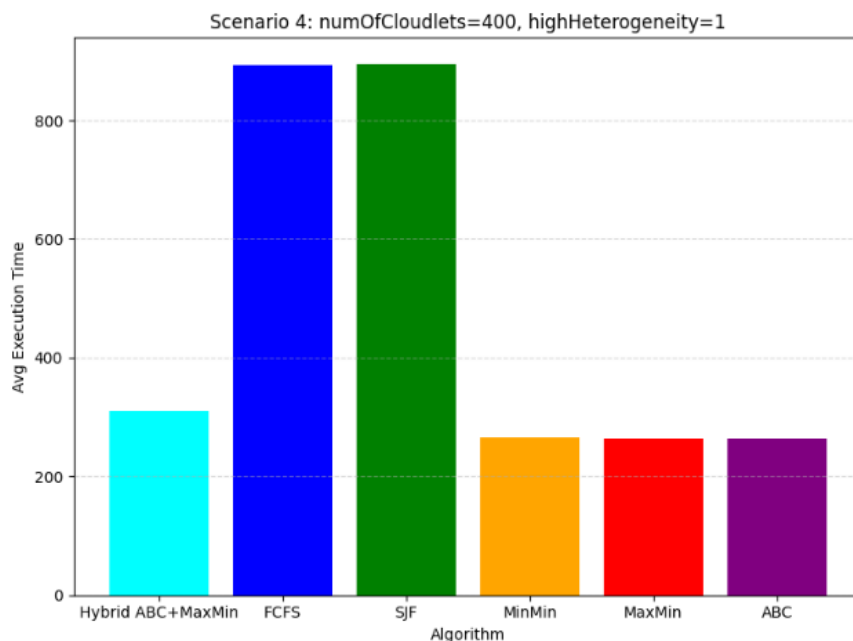


Рисунок 3 – Сценарий с 400 задачами
Figure 3 – Scenario with 400 tasks

На Рисунке 4 представлен график сравнения работы гибридного алгоритма с базовыми со сценариями с 1000 задачами.

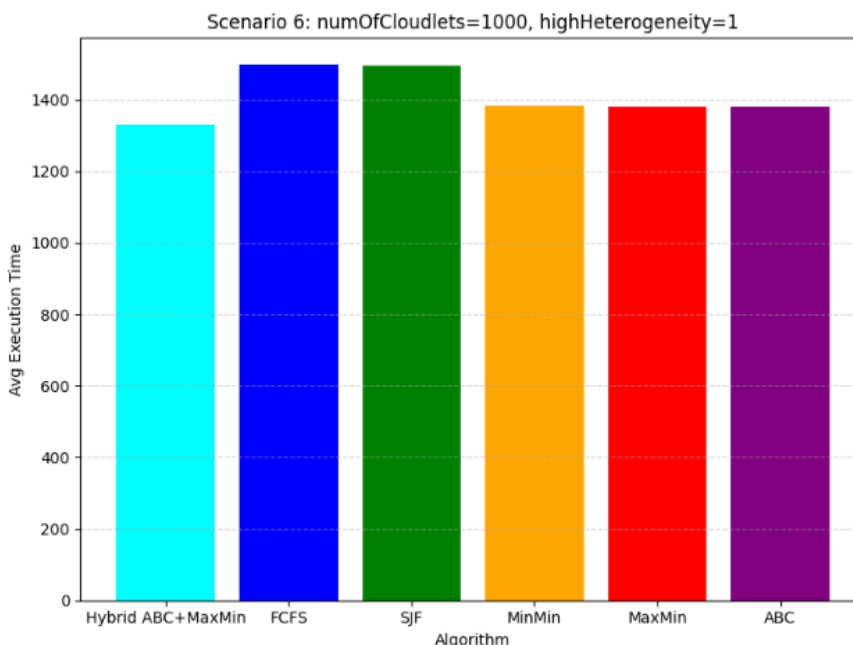


Рисунок 4 – Сценарий с 1000 задачами
Figure 4 – Scenario with 1000 tasks

На Рисунке 5 представлен график сравнения работы гибридного алгоритма с базовыми со сценариями с 5000 задачами.

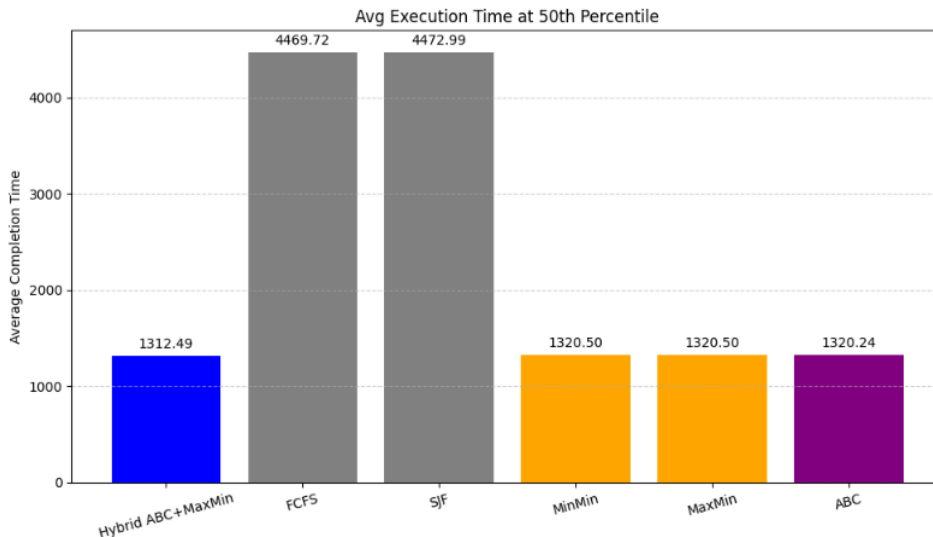


Рисунок 5 – Сценарий с 5000 задачами
Figure 5 – Scenario with 5000 tasks

Обсуждение

Заметно, что системы с более различными характеристиками более чувствительны к выбору алгоритма. Так, алгоритмы FCFS и SJF при всех подобных экспериментах показали результат значительно хуже остальных. Это предсказуемо, так как данные алгоритмы довольно просты. По мере роста количества задач SJF показывает результаты чуть лучше, так как учитывает в своей работе длительность выполнения задач. В целом оба алгоритма применимы только в небольших системах. Из плюсов данных алгоритмов можно выделить простоту их реализации. Если распределенная система имеет относительно однородную архитектуру и задачи обладают схожими характеристиками, где порядок обработки не влияет существенно на общую эффективность, эти алгоритмы могут быть приемлемыми благодаря своей простоте реализации. Если система ориентирована на обработку задач в порядке их поступления (например, очереди заданий в некоторых системах обслуживания), FCFS может оказаться удобным выбором.

Алгоритмы Min-Min и Max-Min показали очень похожие результаты. Небольшое преимущество Min-Min обеспечивается за счет быстрого выполнения маленьких задач. Однако при росте вычислительных задач Max-Min может быть более предпочтительным.

Алгоритмы PSO и ABC наиболее эффективны для ресурсоемких задач. При простых задачах время и вычислительные ресурсы на работу самих алгоритмов слишком высоки относительно времени выполнения самих задач. Алгоритм ABC кажется более универсальным, так как хорошо себя показывает как на простых, так и на более сложных задачах.

Можно заметить, что гибридный алгоритм не очень убедительно себя показывает при низком количестве задач, но его результат повышается при их росте. Так, в эксперименте с 5000 задачами гибридный алгоритм превзошел все базовые. Что подтверждает перспективность работы в данном направлении.

Заключение

Современные высоконагруженные системы, функционирующие в условиях динамически изменяющихся потоков задач, требуют надежных, адаптивных и эффективных алгоритмов распределения ресурсов. Стремительный рост объема данных,

повышение требований к времени отклика и соблюдению SLA (Service Level Agreement) делают традиционные подходы к планированию, такие как FCFS и SJF, все менее актуальными. В этой связи особое внимание заслуживают интеллектуальные и гибридные алгоритмы, способные гибко адаптироваться под характер нагрузки и конфигурацию ресурсов.

В процессе моделирования в среде CloudSim был разработан и протестирован гибридный алгоритм, сочетающий преимущества двух известных методов – ABC (Artificial Bee Colony) и Max-Min. Его основная идея заключается в предварительной классификации задач по их «тяжести» – на легкие и тяжелые. Такое разделение осуществляется на основании 50-го перцентиля длины задач, что позволяет динамически определять границу между типами задач в зависимости от условий конкретной симуляции. Далее, легкие задачи распределяются между виртуальными машинами с использованием алгоритма ABC, эффективно работающего при множественных коротких задачах, где требуется балансировка с учетом вероятностной эвристики. В то же время тяжелые задачи обрабатываются с применением Max-Min, алгоритма, способного стратегически назначать ресурсоемкие задачи на наиболее подходящие машины, минимизируя общее время исполнения (makespan).

Результаты экспериментов продемонстрировали устойчивую эффективность предложенного гибридного подхода. На графиках сравнения видно, что среднее время выполнения задач при использовании ABC+MaxMin существенно ниже, чем при применении стандартных алгоритмов. Особенно ярко преимущество проявляется в условиях высокой гетерогенности ресурсов, где традиционные методы не учитывают различия между задачами и ВМ. Благодаря двойной стратегии, гибридный алгоритм успешно использует сильные стороны каждого подхода:

- ABC – быстрая реакция и вероятность выхода из локальных минимумов;
- MaxMin – стратегическое назначение наиболее критичных задач.

Применимость данного алгоритма выходит за рамки симуляционного моделирования. Он может быть адаптирован для реальных облачных платформ и высоконагруженных распределенных систем.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Durairaj S., Sridhar R. Coherent Virtual Machine Provisioning Based on Balanced Optimization Using Entropy-Based Conjectured Scheduling in Cloud Environment. *Engineering Applications of Artificial Intelligence*. 2024;132. <https://doi.org/10.1016/j.engappai.2024.108423>
2. Han Zh., Tan H., Wang R., Chen G., Li Yu., Lau F.Ch.M. Energy-Efficient Dynamic Virtual Machine Management in Data Centers. *IEEE/ACM Transactions on Networking*. 2019;27(1):344–360. <https://doi.org/10.1109/tnet.2019.2891787>
3. Gaggero M., Caviglione L. Model Predictive Control for Energy-Efficient, Quality-Aware, and Secure Virtual Machine Placement. *IEEE Transactions on Automation Science and Engineering*. 2019;16(1):420–432. <https://doi.org/10.1109/TASE.2018.2826723>
4. Mohammad O.K.J., Salih B.M. Improving Task Scheduling in Cloud Datacenters by Implementation of an Intelligent Scheduling Algorithm. *Informatica*. 2024;48(10). <https://doi.org/10.31449/inf.v48i10.5843>
5. Ahmed A., Adnan M., Abdullah S., Ahmad I., Alturki N., Jamel L. An Efficient Task Scheduling for Cloud Computing Platforms Using Energy Management Algorithm: A Comparative Analysis of Workflow Execution Time. *IEEE Access*. 2024;12:34208–34221. <https://doi.org/10.1109/ACCESS.2024.3371693>

6. Zolfaghari R., Rahmani A.M. Virtual Machine Consolidation in Cloud Computing Systems: Challenges and Future Trends. *Wireless Personal Communications*. 2020;115(3):2289–2326. <https://doi.org/10.1007/s11277-020-07682-8>
7. Geeta K., Prasad V.K. Multi-Objective Cloud Load-Balancing with Hybrid Optimization. *International Journal of Computers and Applications*. 2023;45(10):611–625. <https://doi.org/10.1080/1206212X.2023.2260616>
8. Stavrinides G.L., Karatza H.D. An Energy-Efficient, QoS-Aware and Cost-Effective Scheduling Approach for Real-Time Workflow Applications in Cloud Computing Systems Utilizing DVFS and Approximate Computations. *Future Generation Computer Systems*. 2019;96:216–226. <https://doi.org/10.1016/j.future.2019.02.019>
9. Belgacem A., Beghdad-Bey K. Multi-Objective Workflow Scheduling in Cloud Computing: Trade-Off Between Makespan and Cost. *Cluster Computing*. 2022;25(1):579–595. <https://doi.org/10.1007/s10586-021-03432-y>
10. Jena U.K., Das P.K., Kabat M.R. Hybridization of Meta-Heuristic Algorithm for Load Balancing in Cloud Computing Environment. *Journal of King Saud University – Computer and Information Sciences*. 2022;34(6):2332–2342. <https://doi.org/10.1016/j.jksuci.2020.01.012>

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Дойчев Владимир Сергеевич, магистрант кафедры программной инженерии Сибирского федерального университета, Красноярск, Российская Федерация. **Vladmir S. Doichev**, Master's Degree student at the Department of Software Engineering, Siberian Federal University, Krasnoyarsk, the Russian Federation.
e-mail: doichevvs@gmail.com

Статья поступила в редакцию 18.06.2025; одобрена после рецензирования 15.07.2025; принята к публикации 26.07.2025.

The article was submitted 18.06.2025; approved after reviewing 15.07.2025; accepted for publication 26.07.2025.