УДК 004.021:004.42 **DOI** 10.15622/ia.25.2.10

Е.С. Трушкин, В.И. Фрейман

ПРЕДСКАЗАТЕЛЬНЫЙ МЕТОД РАСПРЕДЕЛЕНИЯ РЕСУРСОВ В ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ НА ОСНОВЕ МНОГОКРИТЕРИАЛЬНОЙ МОДЕЛИ ПРИНЯТИЯ РЕШЕНИЙ

Трушкин Е.С., Фрейман В.И. **Предсказательный метод распределения ресурсов в вычислительных системах на основе многокритериальной модели принятия решений.**

Аннотация. Современные вычислительные системы функционируют, как правило, в условиях гетерогенности и переменной нагрузки. Важным инструментом обеспечения высоких показателей функционирования (например, производительность, надежность, устойчивость) является эффективное распределение вычислительных ресурсов. В связи с этим актуальной проблемой является разработка методов распределения задач, позволяющих улучшать несколько показателей одновременно. Предлагаемый в работе подход представляет собой расширение предсказательного метода распределения ресурсов. Это осуществляется за счет введения многокритериальной модели принятия решений, включающей прогнозное время выполнения, текущую загрузку узлов и достоверность прогноза, оцениваемую по статистике расхождений фактических и прогнозных значений. Объект исследования – вычислительные системы с неоднородными узлами, обрабатывающими потоки задач переменной сложности. Предмет исследования – модели и алгоритмы предсказательного распределения вычислительных ресурсов на основе многокритериального принятия решений. Цель исследования – повышение эффективности и устойчивости функционирования гетерогенных вычислительных систем за счет использования более обоснованного механизма выбора узла на основе совокупности критериев. Выполнен обзор существующих динамических и предсказательных методов распределения, выявлены их преимущества, недостатки и ограничения по эффективному применению. Разработана многокритериальная модель принятия решений, реализующая построение множества Парето-оптимальных решений и процедуру арбитража. Проведено программное моделирование в различных сценариях функционирования системы, включая условия со сниженной достоверностью статистики. Результаты исследования показали, что предлагаемый предсказательный метод на основе многокритериального принятия решений обеспечивает снижение среднего времени выполнения задач и повышение равномерности нагрузки узлов по сравнению с известными подходами. Полученные результаты предлагается использовать при построении гетерогенных вычислительных систем с адаптивными системами управления ресурсами.

Ключевые слова: вычислительная система, методы распределения ресурсов, Парето-оптимальность, многокритериальный выбор, прогнозирование нагрузки, статистические данные.

1. Введение. Одной из ключевых задач, стоящих перед разработчиками современных вычислительных систем, является эффективное распределение вычислительных ресурсов. От того, насколько рационально назначаются задачи на элементы (узлы) вычислительной системы, напрямую зависят ее основные показатели, например, такие как производительность, масштабируемость

и отказоустойчивость. В условиях увеличивающейся динамики потоков задач, разнообразия типов нагрузок и неоднородности узлов становится очевидным, что существующие методы, которые можно разделить на статические и динамические, требуют серьезной модернизации [1].

Статические алгоритмы планирования распределяют задачи по узлам заранее, исходя из известных характеристик задач и ресурсов. Классический пример – методы планирования, ориентированные на графы задач (DAG), такие как алгоритмы HEFT и CPoP [2]. Они рассчитывают приоритеты задач (например, по «весу» выполнения) и закрепляют их за компонентами вычислительных систем (например, процессорами) так, чтобы минимизировать общее время выполнения. Преимущество таких методов – высокая эффективность расписания при выполнении предпосылок (известность времени выполнения и связи задач) и небольшое время планирования [3]. Однако эти алгоритмы сложно адаптировать к изменениям: в статической схеме нет обратной связи с текущей загрузкой, поэтому при колебаниях нагрузки или неточности оценок, такая схема перестает эффективно работать. Как отмечается в работе [4], статические схемы в современных многомерных системах часто оказываются нерезультативными: они плохо справляются с переменным спросом и неоднородностью ресурсов. Так, в опытах по объединенным суперкомпьютерным системам статический подход приводил к высокому уровню сбоев (более 30%), тогда как альтернативное (динамическое) планирование снижало этот показатель до ~8%. Кроме того, задача статического планирования является NP-полной, что вынуждает использовать эвристики: простейшие алгоритмы (например, с поиском лучшего места) при большом числе задач дают высокую вычислительную сложность [5]. Таким образом, ограничения статических методов – это зависимость от корректности исходных данных и отсутствие адаптивности: они не учитывают изменчивость реальных условий и не справляются с неожиданными пиками нагрузки.

Классические динамические методы распределения хорошо изучены и широко применяются благодаря простоте реализации и малым вычислительным издержкам; они основываются на реактивном контроле (мониторинг текущей загрузки, очередей и т.п.) [6 – 7]. Алгоритмы типа Least-Connections [8] легко реализуются и имеют низкие вычислительные затраты, поскольку основываются на простом мониторинге показателей (CPU, очередь запросов и т.п.). Такой подход хорошо подходит для адаптивности: планировщик

быстро реагирует на перегрузку того или иного узла и перераспределяет нагрузку, что значительно повышает эффективность по сравнению со статикой [9]. Например, H. Shim в работе [4] проводит сравнение и показывает, что динамический алгоритм обеспечил до 3,5 раз более эффективное использование ресурсов, чем статический.

Тем не менее, у динамических методов есть ограничения. Во-первых, они принимают решения на основе только текущей информации, игнорируя тренды и исторические закономерности. Это означает, что при резких изменениях нагрузки или в условиях гетерогенности ресурсов реактивный планировщик может совершать повторяющиеся или контрпродуктивные действия, что говорит об ухудшении качества принятия решений [10]. Как показали исследования, реактивные схемы без предсказаний не учитывают предыдущие данные, что вызывает задержки и удлинение времени ожидания задач [11]. Во-вторых, даже при небольшой «стоимости» принятия решений динамика может дать неоптимальные результаты: например, если система перестраивается слишком часто, то растут издержки на переключения и коммуникацию. Кроме того, качественное распределение нагрузки лишь по одному критерию (занятость процессора) часто не учитывает других важнейших параметров (энергоэффективность, приоритеты задач, надежность), что ограничивает эффективность подхода в современных облачных средах.

Предсказательные (предиктивные) методы расширяют динамический подход заложенным прогнозом. Они используют исторические данные [12] и машинное обучение [13] для оценки будущей нагрузки или времени выполнения задач, что позволяет планировать распределение ресурсов заранее. С точки зрения принципа, такие методы могут сочетать текущий мониторинг с моделированием «на что можно рассчитывать в ближайшем будущем». Преимущество прогностических схем – возможность оптимизировать решение (например, заранее подготовить узлы к ожидаемому пику нагрузки, ускорить нужные серверы или более равномерно распределить большие задачи). Обзор современных исследований показывает, что ИИ-решения способны повысить адаптивность и энергоэффективность облачных сред, сократить простои и учесть сложные зависимости между задачами [14 – 15]. Несмотря на широкие возможности, предсказательные методы также имеют ряд ограничений. В первую очередь, точность принимаемых решений напрямую зависит от качества прогноза: ошибки в моделях

машинного обучения могут приводить к неверному выбору узла и, как следствие, к ухудшению производительности системы. Во вторую очередь, построение и поддержание таких моделей требует существенных вычислительных ресурсов и больших объемов обучающих данных, что делает применение подхода затруднительным в средах с ограниченными ресурсами. В работе [16] поднят вопрос о ресурсной стоимости процессов распределения вычислительных ресурсов в динамических, ресурсно-ограниченных вычислительных средах. Проблема рассматривается в плане снижения ресурсной стоимости многокритериальной оптимизации распределения вычислительных ресурсов, однако предложено использование комплексов метаэвристик, с их последовательным выбором и применением в зависимости от ограничений на время получения результата. Так же, ИИ-модели чувствительны к изменению характера нагрузки: при сдвиге распределений или появлении новых типов задач требуется дополнительное (fine-tuning) или новое обучение, иначе прогнозы теряют актуальность и обоснованность [17]. В статье [18] производится сравнение ресурсной эффективности эволюционных алгоритмов с интеграцией ламарковской и болдуинской эволюций в задачах распределения вычислительных ресурсов.

Таким образом, ни один из рассмотренных подходов не лишен недостатков. Статические методы не учитывают изменчивость нагрузки и зависят от точности исходных оценок. Динамические решения просты и быстры, но ориентированы лишь на текущие показатели, не предвидят изменения условий и могут сбалансировать систему всего по одному параметру. Прогностические схемы более гибкие, однако требуют сложной инфраструктуры ИИ: большие вычислительные затраты и массив обучающих данных в совокупности с чувствительностью к ошибкам прогнозов ограничивают их надежность. Такая совокупность проблем обуславливает потребность в более устойчивых методах, способных учитывать сразу несколько факторов при выборе назначения задачи.

Авторами в работе [19] был предложен предсказательный метод распределения, который применим в системах с жесткими ограничениями по вычислительным ресурсам. Он отличается от аналогичных тем, что прогнозирование будущей нагрузки и адаптивное перераспределение вычислительных задач выполняется с учетом совокупности текущего состояния системы и статистической информации, о предыдущих схожих по характеристикам задачах. Такой подход позволяет заранее оценивать будущую загруженность вычислительных узлов и принимать более обоснованные решения

о распределении ресурсов. Однако ранее авторами не рассматривался вопрос ошибок в прогнозах. Поэтому возникает необходимость в расширении модели принятия решений, реализуемой в рамках предсказательного метода, до многокритериальной модели, способной учитывать несколько факторов одновременно. Настоящая статья развивает этот подход и формализует процесс выбора узла, как многокритериальную задачу. Ключевым отличием является введение критерия достоверности прогноза, построенного на статистике расхождений между прогнозным и фактическим временем выполнения.

Под вычислительной задачей в настоящей работе понимается законченный процесс или работа, требующая выделения ресурсов узла (CPU, памяти, дисковых операций и т.д.) и обладающая определенными характеристиками (например, оценочным временем выполнения или приоритетом). Такие задачи в реальных системах могут существенно различаться по своей природе, требованиям и временным ограничениям. Приведем некоторые типичные примеры практических задач, встречающихся в современных вычислительных средах:

1. Научные и инженерные вычисления. К ним относятся ресурсоемкие пакетные задачи на суперкомпьютерах и кластерах. Примеры – численное моделирование физических процессов (гидродинамика, аэродинамика), климатические расчеты, обработка последовательностей ДНК, визуализация научных данных, финансовое и химико-биологическое моделирование и др. Такие задачи часто запускаются как batch processing, которые пользователи отправляют на планировщик кластера. Планировщик распределяет эти задания по свободным узлам кластера для параллельного выполнения. Данные приложения требуют большего вычислительного времени и, как правило, имеют заранее известную структуру задачи (например, граф зависимостей) [20].

2. Веб-сервисы и микросервисы. Каждый HTTP-запрос или транзакция к веб-приложению интерпретируется системой как отдельная задача (процесс или поток), которую нужно быстро обработать. Примеры – это генерация веб-страниц, обращение к API, выполнение запросов к базам данных и т.п. Такие задачи обычно недолговечны (от нескольких миллисекунд до секунд) и имеют строгие требования по задержке (латентности). Планировщики распределяют их по серверам и виртуальным машинам в реальном времени, чтобы обеспечить высокую пропускную способность и быстрый отклик сервисов. В отличие от batch processing, здесь

оптимизация идет по минимизации задержек и соблюдению приоритетов каждого запроса [21].

Цель данной работы – улучшить предложенный ранее предсказательный метод распределения задач за счет многокритериального принятия решения при выборе узла.

Задачи статьи:

1. Формализовать многокритериальную модель принятия решений, определить показатели эффективности работы вычислительной системы (раздел 2).
2. Ввести в перечень ключевых критериев, кроме традиционно используемых (прогнозируемое время выполнения, текущее/ожидаемое значение загрузки узла), новый – метрику надежности/достоверности прогноза (раздел 3).
3. Применить подход для оптимального назначения задачи, сочетающий построение Парето-множества кандидатов и последующую процедуру арбитража/ранжирования с учетом заданных предпочтений (разделы 4 – 5).
4. Разработать алгоритм многокритериального принятия решений как основного этапа предсказательного метода распределения ресурсов в вычислительных системах (раздел 6).
5. Создать программу имитационного моделирования и провести эксперименты для сравнения предсказательного метода многокритериального принятия решений с предсказательным однокритериальным и динамическим (раздел 7 – 8).

2. Постановка задачи. В рамках рассматриваемой работы проблема распределения ресурсов формулируется как назначение вычислительной задачи на один из доступных узлов в условиях многокритериального выбора с использованием статистической информации.

Рассматривается система, состоящая из множества вычислительных узлов:

$$U = \{u_1, u_2, \dots, u_n\}.$$

Каждая вновь поступающая задача Z должна быть назначена на один из доступных узлов до начала ее выполнения.

Условия функционирования системы:

− узлы являются гетерогенными и различаются по производительности,

– система работает в условиях переменной нагрузки: количество поступающих задач и их характеристики (например, требуемая вычислительная сложность) со временем изменяются,

– в случае отсутствия свободных узлов задача будет поставлена в очередь или получит отказ в обслуживании при заполнении очереди.

Показателями, по которым оценивается эффективность системы, в данной работе выбраны быстродействие (время обработки задач) и равномерность распределения нагрузки между узлами системы.

Постановка задачи сводится к многокритериальной оптимизации: необходимо выбрать узел $u^* \in U$, для которого совокупность выбранных критериев будет оптимальной с учетом ограничений.

Пусть назначение задачи Z на узел u_j описывается кортежем критериев:

$$C(u_j, Z) = \langle c_1(u_j, Z), c_2(u_j, Z), \dots, c_k(u_j, Z), \dots, c_n(u_j, Z) \rangle,$$

где c_k – критерий качества, $k \in [1, n]$ (например: время выполнения, загрузка, вероятность отказа, достоверность прогноза и др.); n – количество критериев. Тогда задача распределения ресурсов формулируется как выбор узла, для которого кортеж $C(u_j, Z)$ будет иметь наилучшую совокупность критериев. Для решения поставленной задачи далее будут предложены критерии и подходы для нахождения их оптимальной совокупности.

3. Критерии оценки узлов. В общем виде, число критериев качества в задаче распределения ресурсов не фиксировано и может быть неограниченно большим. Увеличение числа критериев n влияет на несколько важных аспектов [22]:

– расширяются пространство компромиссов между критериями,

– растет объем данных для сравнения кандидатов (узлов),

– повышается вычислительная сложность процедур фильтрации и последующего ранжирования,

– снижается наглядность интерпретации результатов. Поэтому при практической реализации обычно выбирают ограниченное подмножество информативных и интерпретируемых показателей (критериев) – применяют методы отбора/редукции признаков [23], кластеризацию в пространстве критериев [24] или скаляризацию [25].

В данной работе, как наиболее типичные для распределенных вычислительных систем, используются три критерия, отражающих функционирование узлов:

1. Прогнозное время выполнения задачи – ожидаемая длительность выполнения задачи Z на узле u_j, рассчитанная по модели прогнозирования:

$$c_1 = T_{\text{п}}(u_j, Z).$$

Прогнозируемое значение параметра определяется в зависимости от расчетного значения и реальных (статистических) значений обработки аналогичного:

$$T_{\text{п}}(r) = \alpha T_{\text{р}}(r) + \sum_{s=1}^{N} \beta_s T_{\text{с}}(r-s),$$

где $T_{\text{п}}(r)$ – прогнозируемое значение на r-м шаге; $T_{\text{р}}(r)$ – расчетное значение на r-м шаге; $T_{\text{с}}(r–1)$ … $T_{\text{с}}(r–N)$ – статистические значения предыдущей обработки; N – глубина памяти (или максимальное значение количества имеющихся данных); α – весовой коэффициент расчетного значения, β_s – весовой коэффициент s-го статистического значения.

Этот критерий отражает быстродействие узла.

2. Текущая загрузка узла – время, которое необходимо узлу, чтобы обработать уже назначенные ему задачи.

$$c_2 = T_{\text{п}}(u_j, Z_{i-1}) + T_{\text{п}}(u_j, Z_{i-2}) + \cdots + T_{\text{п}}(u_j, Z_{i-m}),$$

где m – количество предыдущих задач.

Учет этого критерия позволяет распределять задачи равномерно и предотвращать перегрузку отдельных узлов.

3. Достоверность прогноза – показатель, отражающий статистические погрешности прогнозирования на конкретном узле.

Рассчитывается на основе двух рядов наблюдений:

- прогнозное время выполнения задачи: $T_{\text{п},i}(u_j, Z)$,
- фактическое время выполнения задачи: $T_{\text{ф},i}(u_j, Z)$.

Погрешность прогноза для каждого момента времени (шага) i определяется следующим образом:

$$\Delta T_i(u_j, Z) = T_{\text{ф},i}(u_j, Z) - T_{\text{п},i}(u_j, Z).$$

Для дальнейших расчетов предлагается использовать также значение абсолютной погрешности:

$$\Delta A_i = |\Delta T_i|.$$

Далее определяются необходимые характеристики погрешности. Для качественной оценки прогноза предлагается использовать характеристики погрешности на скользящем окне M. Это окно ограничено условием $M \leq i$, где i – число накопленных наблюдений, если статистики недостаточно ($i < M$), то характеристики рассчитываются по доступным данным ($M = i$).

Среднее значение погрешности прогнозирования:

$$\Delta T_{\text{ср}}(u_j, Z) = \frac{1}{M} \sum_{m=i-M}^{i} \Delta T_m(u_j, Z), i \geq M,$$

$$\Delta T_{\text{ср}}(u_j, Z) = \frac{1}{i} \sum_{m=0}^{i} \Delta T_m(u_j, Z), M > i \geq 0.$$

В случае, когда $\Delta T_{\text{ср}}(u_j, Z) > 0$ это означает, что на узле фактическое время достаточно часто больше прогнозного. А это значит – есть риск срыва сроков выполнения задач, которые важно обработать вовремя. Если $\Delta T_{\text{ср}}(u_j, Z) < 0$, то узел систематически обрабатывает задачу быстрее прогнозного времени – возможно, часть ресурсов не используется.

Стоит отметить, что возможна ситуация, когда $\Delta T_{\text{ср}}(u_j, Z) = 0$. В этом случае могли возникнуть следующие проблемы:

1. Наблюдаемый узел имеет одинаковую среднюю погрешность, как в большую сторону, так и в меньшую (например, {+2, -2, +2, -2}).
2. На узле наблюдалась существенная погрешность (+ или -), которая компенсировала величину суммы противоположных погрешностей (например, {+8, -2, -2, -2, -2}.

Для решения этих проблем используем расчет среднего значение абсолютной погрешности прогнозирования:

$$\Delta A_{\text{ср}}(u_j, Z) = \frac{1}{M} \sum_{m=i-M}^{i} \Delta A_m(u_j, Z), i \geq M,$$

$$\Delta A_{\text{ср}}(u_j, Z) = \frac{1}{i}\sum_{m=0}^{i} \Delta A_m(u_j, Z), M > i \geq 0.$$

$\Delta A_{\text{ср}}$ показывает на сколько в среднем прогнозируемое значение отличается от фактического. В отличие от простого среднего значения погрешности, абсолютное не обнуляется при чередовании (+) и (−). Чем меньше $\Delta A_{\text{ср}}$, тем точнее прогноз в среднем.

Также, кроме описанных выше проблем, может возникнуть неопределенность, при которой на нескольких узлах абсолютная погрешность $\Delta A_{\text{ср}}$ будет одинакова. Для того чтобы избежать этого, предлагается использовать расчет среднеквадратического отклонения погрешностей σ:

$$c_3 = \sigma(u_j, Z) = \sqrt{\frac{1}{M-1}\sum_{m=i-M}^{i} \left(\Delta T_m(u_j, Z) - \Delta T_{\text{ср}}(u_j, Z)\right)^2}.$$

Этот расчет показывает, насколько отклоняется погрешность прогноза от среднего значения. Если σ достаточно велико, значит за последние M отсчетов фактическое значение сильно отличалось от прогнозного.

Далее рассмотрим числовой пример. Предположим, что имеется два узла (А и Б), среди которых необходимо выбрать один для обработки задачи.

1. Узел А: погрешности [+2, -2, +2, -2].

$$\Delta A_{\text{ср}} = 2$$
$$\Delta T_{\text{ср}} = 0$$
$$\sigma \approx 2{,}31$$

2. Узел Б: погрешности [+4, -2, -2, 0].

$$\Delta A_{\text{ср}} = 2$$
$$\Delta T_{\text{ср}} = 0$$
$$\sigma \approx 2{,}83$$

Оба узла имеют одинаковый $\Delta A_{\text{ср}} = 2$ и $\Delta T_{\text{ср}} = 0$, но среднеквадратическое отклонение у них разное. Это позволяет сделать

вывод: узел А: $\sigma \approx 2{,}31$ – прогноз более точен (разброс погрешности в меньшем диапазоне). Узел В: $\sigma \approx 2{,}83$ – прогноз менее точен (разброс погрешности в большем диапазоне).

Использование выбранных трех критериев обусловлено их практической релевантностью для задач распределения вычислительных задач: c_1 отражает производительность, c_2 – состояние загрузки, c_3 – точность прогнозов и устойчивость работы системы при неопределенности. В каждом конкретном случае перечень критериев может быть изменен – соответствующие последствия для методов отбора и вычислительной сложности обсуждаются в разделах 4 и 5, где также приводится используемый алгоритм предварительной фильтрации узлов и последующего выбора.

4. Концепция Парето-оптимальности. В многокритериальной постановке, как правило, сложно выделить один узел, который был бы наилучшим по всем показателям. Для этого используется понятие Парето-доминирования [26].

Парето-доминирование:

Пусть два узла u_i и u_j имеют соответствующие кортежи критериев $C(u_i)$ и $C(u_j)$.

Тогда u_i доминирует u_j, если:

- для всех $k \in [1, n]$: $c_k(u_i) \leq c_k(u_j)$,
- существует хотя бы один k': $c_{k'}(u_i) < c_{k'}(u_j)$.

Парето-множество:

Множество узлов, которые не доминируются ни одним другим, образует Парето-фронт или множество Парето-оптимальных решений:

$$P = \{ u_j \in U \mid \nexists u_i \in U : C(u_i) \prec C(u_j) \}.$$

Множество P содержит все такие элементы $u_j \in U$, для которых не существует другого элемента $u_i \in U$, доминирующего u_j. Парето-множество отражает набор компромиссных решений. Каждый элемент этого множества является обоснованным с какой-либо точки зрения, и не существует универсального предпочтения без дополнительной информации. Выбор из этого множества требует сужения на основе предпочтений системы или пользователя [27].

Построение Парето-фронта связано с процедурой перебора всех критериев для каждого узла. Поэтому вычислительная сложность формирования Парето-фронта зависит от количества узлов и количества критериев. Характер зависимостей и ограничения применения данного метода являются темой дальнейших исследований.

5. Роль арбитража и лица, принимающего решение (ЛПР). Построение множества Парето-оптимальных решений позволяет выделить и отбросить те варианты распределения задачи (или перечня задач), которые являются неудовлетворительными по сравнению с остальными. Однако отсутствие доминирования не означает равную предпочтительность – необходимо выбрать конкретный узел для назначения задачи. Этот выбор невозможно осуществить без учета предпочтений некоторого субъекта – лица, принимающего решение (ЛПР) [28]. В случае автоматического управления распределением задач, что и имеет место в вычислительных системах, эту функцию выполняет программный модуль принятия решения (планировщик).

Для того, чтобы принять решение, применяются методы, которые описаны ниже. Они не гарантируют, что останется одно из возможных решений, но позволяют существенно уменьшить их количество. После чего выбирается любой из оставшихся вариантов, так как они считаются равнозначными.

Метод агрегирования (линейная свертка).

При наличии кортежа весов $w = \langle w_1, w_2 \dots, w_m \rangle$, где $w_k \geq 0$ и $\sum w_k = 1$, вводится функция полезности:

$$S(u_j, Z) = \sum_{k=1}^{m} w_k \cdot c_k(u_j, Z).$$

Каждому критерию c_k сопоставляется коэффициент значимости w_k. После чего выбирается узел с минимальным (или максимальным – в зависимости от задачи) значением функции полезности. Преимущества: простота, гибкость. Недостатки: необходимость задания весов, возможность потери информации о компромиссах [29].

Иерархическая фильтрация.

В данном методе критерии ранжируются по приоритету. Сначала фильтруются узлы по главному критерию (например, прогнозируемое время обработки), затем – по следующему и т.д., пока не останется один или несколько вариантов. Метод подходит в случаях, когда некоторые критерии являются жесткими ограничениями, например, не назначать задачи на узел с высокой погрешностью прогнозирования. Преимущества: высокая управляемость, логичность. Недостатки: может отбрасывать близкие по значению, но в целом более выгодные решения [30].

Также существуют методы и модели для принятия решений, которые включают более сложные механизмы выбора. Примеры: логические правила (если… то…) [31], лексикографический порядок, нейросетевые функции предпочтений [32].

Интеграция механизма арбитража и моделирования ЛПР позволяет системе переходить от «чистой» оптимизации к интеллектуальному принятию решений, учитывающему контекст, цели, политику управления и пользовательские требования. Это является ключевым шагом на пути к созданию адаптивного планировщика, способного устойчиво функционировать в широком диапазоне условий

6. Алгоритм реализации многокритериального предсказательного метода распределения задач. Предложенная в разделах 2 – 5 математическая модель реализуется в виде алгоритма, который активируется при поступлении каждой новой задачи в систему. Алгоритм работает в реальном времени и состоит из следующих последовательных шагов.

Шаг 1. Инициализация и сбор актуального состояния системы.

Перед обработкой новой задачи система выполняет сбор информации о текущем состоянии вычислительных узлов $U = \{u_1, u_2, \ldots, u_n\}$. Формируются показатели:

- текущая загрузка каждого узла (количество активных задач, оставшееся время их выполнения, использование ресурсов),
- обновленная статистика по ранее выполненным задачам (прогнозные и фактические времена),
- доступность узла для приема новой задачи.

Эта информация формирует исходные данные для прогноза и оценки критериев.

Шаг 2. Прогнозирование времени выполнения новой задачи.

Для каждой пары «задача Z – узел u_j» рассчитывается прогнозное время $T_{\text{п},i}(u_j, Z)$ на основании модели прогнозирования, использующей накопленные статистические данные и параметры задачи (например: объем вычислений, тип задачи, требуемые ресурсы). Если статистика по данному типу задач или узлу ограничена, применяется адаптивная аппроксимация на основе близких классов задач. Результаты прогноза сохраняются в таблицу критериев.

Шаг 3. Формирование кортежа критериев и условий доминирования.

Для каждого узла формируется кортеж критериев:

$$C(u_j, Z) = \langle c_1(u_j, Z), c_2(u_j, Z), \ldots, c_k(u_j, Z), \ldots, c_n(u_j, Z) \rangle.$$

После чего эти кортежи сводятся в таблицу значений критериев для всех рассматриваемых узлов (таблица 1).

Таблица 1. Результирующая таблица критериев

	Критерии					
Номер узла	c_1	c_2	…	c_i	…	c_n
1	$c_1(u_1, Z)$	$c_2(u_1, Z)$	…	$c_k(u_1, Z)$	…	$c_n(u_1, Z)$
2	$c_1(u_2, Z)$	$c_2(u_2, Z)$	…	$c_k(u_2, Z)$	…	$c_n(u_2, Z)$
…	…	…	…	…	…	…
j	$c_1(u_j, Z)$	$c_2(u_j, Z)$	…	$c_k(u_j, Z)$	…	$c_n(u_j, Z)$
…	…	…	…	…	…	…
m	$c_1(u_m, Z)$	$c_2(u_m, Z)$	…	$c_k(u_m, Z)$	…	$c_n(u_m, Z)$

Ключевое требование этого шага – однозначно задать отношение «лучше/хуже» для каждой компонентной величины. Для каждого критерия $k = [1, n]$ фиксируется правило:

- либо «меньше – лучше» (минимизационный критерий),
- либо «больше – лучше» (максимизационный критерий).

На основе этих компонентных отношений задается отношение доминирования между двумя узлами. Эти данные служат входом для шага 4 – фильтрации (выделение недоминируемых решений) и последующей сортировки/отбора.

Шаг 4. Построение множества Парето-оптимальных решений.

После определения критериев и построения кортежей производится их анализ для всех узлов. Для каждой пары (u_i, u_j) выполняется проверка на Парето-доминирование. В множество P включаются только те узлы, которые недоминируются никаким другим.

Таким образом, множество P определяет набор альтернатив, являющихся компромиссно оптимальными по совокупности критериев. Построение множества Парето-оптимальных решений реализовано методом полного перебора, что является эффективным при небольшом числе узлов и обеспечивает точное выделение недоминируемых решений.

Шаг 5. Принятие окончательного решения (Арбитраж).

Если множество P содержит более одного узла, выполняется процедура арбитража, реализующая предпочтения лица, принимающего решение (ЛПР), или заданную политику планировщика (возможные стратегии были описаны в разделе 5). Результатом арбитража является окончательное решение о назначении задачи Z на узел $u^* \in P$.

Шаг 6. Выполнение задачи и обновление статистики.

После назначения задачи происходит ее выполнение на выбранном узле u^*. По завершении фиксируются фактические показатели: время выполнения, использованные ресурсы, возникшие отклонения от прогноза. Эти данные добавляются в статистическую базу и используются для пересчета параметров прогнозной модели. Таким образом, система обладает свойством самообучения и повышает точность прогнозов при длительной эксплуатации.

7. Программное моделирование. Для экспериментальных исследований авторами ранее использовался AnyLogic [33]. Для более гибкой настройки разработанная многокритериальная модель была реализована в виде программного комплекса на языке Python, включающего графический интерфейс пользователя и средства визуального анализа результатов. Реализация выполнена в модульной архитектуре и позволяет проводить серию вычислительных экспериментов для сравнения различных методов распределения задач в вычислительных системах.

Эксперименты направлены на сравнительный анализ трех методов распределения:

1. Динамического (Д) – это существующий метод, в котором задачи назначаются на узел с минимальной текущей загрузкой.
2. Предсказательного однокритериального (П) – это авторский метод, согласно которому выбор узла с минимальным прогнозным временем выполнения.
3. Многокритериального предсказательного (М) – этот метод является улучшенной версией предсказательного однокритериального метода (П), в нем выбор узла происходит на основе совокупности критериев (например, прогнозного времени, загрузки узла, достоверность прогноза).

Серия экспериментов выполнена поэтапно, с постепенным усложнением исходных условий, что позволяет проследить изменение поведения алгоритмов при переходе от идеальных к реальным сценариям.

Для моделирования рассматривалась система, имеющая следующие входные данные:

- количество задач,
- группы сложности задач,
- вычислительная сложность каждой группы задач, в условных единицах (у.е.),
- пропорция распределения задач по группам сложности,
- производительность, в у.е./е.в. (единица времени).

Все входные данные являются синтетическими.

7.1. Сценарий 1. Идеальные условия – равномерные и стабильные узлы. Цель: проверить корректность работы модели в условиях, когда отсутствуют различия между узлами и статистика прогноза полностью совпадает с фактическими значениями. В таких условиях любой рациональный метод назначения задач должен приводить к одинаковым результатам по времени выполнения и распределению нагрузки.

Условия эксперимента:

- все узлы имеют одинаковую производительность,
- задачи однородные, с одинаковой вычислительной сложностью,
- исторические данные не содержат выбросов, то есть статистика стабильна,
- количество задач и длина временного ряда выбраны так, чтобы прогноз полностью совпадал с фактическим временем обработки задач.

Ожидаемые результаты:

- методы Д, П и М обеспечивают одинаковое время выполнения и равномерную загрузку узлов,
- Парето-множество в методе М будет содержать все узлы, так как критерии идентичны,
- достоверность прогноза максимальна, а арбитраж не влияет на результат.

На рисунке 1 продемонстрирована конфигурация первого сценария.

В этих условиях все три метода – Д, П, М должны демонстрировать одинаковое поведение. Результаты моделирования этого сценария показаны на рисунке 2.

Рисунок 2 иллюстрирует результаты экспериментов для сценария, в котором нагрузка распределяется равномерно и отсутствуют существенные колебания входных параметров. Как видно из таблицы, при увеличении количества задач от 1000 до 5000 все методы: динамический, однокритериальный предсказательный и многокритериальный предсказательный – демонстрируют одинаковые значения как расчетного, так и фактического времени обработки. Это отражено в одинаковых значениях столбцов «Динамический (расчетное)», «Динамический (фактическое)», «Предсказательно однокритериальный (р/ф)» и «Предск. многокритериальный (р/ф)», где наблюдаются идентичные величины (300, 600, 900, 1200, 1500 мс). Кроме того, показатели баланса загрузки на узлы во всех случаях равны нулю (показывают дельту загрузки между самым нагруженным узлом и самым ненагруженным), что подтверждает отсутствие различий между методами.

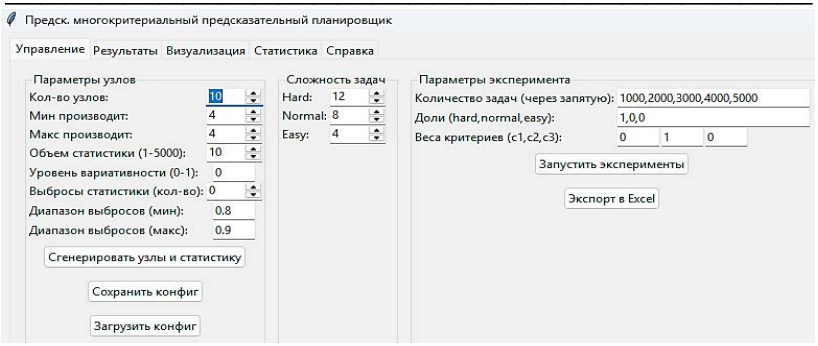
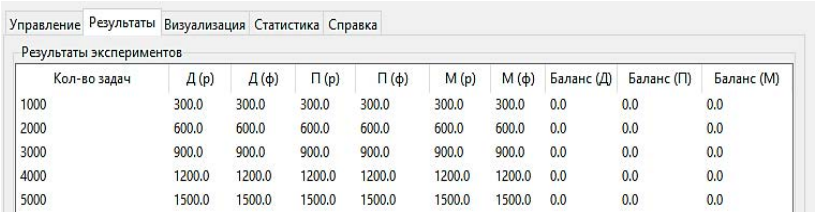


Рис. 1. Окно конфигурации сценария 1



Кол-во задач	Д (р)	Д (ф)	П (р)	П (ф)	М (р)	М (ф)	Баланс (Д)	Баланс (П)	Баланс (М)
1000	300.0	300.0	300.0	300.0	300.0	300.0	0.0	0.0	0.0
2000	600.0	600.0	600.0	600.0	600.0	600.0	0.0	0.0	0.0
3000	900.0	900.0	900.0	900.0	900.0	900.0	0.0	0.0	0.0
4000	1200.0	1200.0	1200.0	1200.0	1200.0	1200.0	0.0	0.0	0.0
5000	1500.0	1500.0	1500.0	1500.0	1500.0	1500.0	0.0	0.0	0.0

Рис. 2. Окно результатов сценария 1

Такой результат объясняется тем, что в данном сценарии выбор узла не влияет на итоговое время выполнения: все узлы находятся в одинаковых условиях, и даже сложные методы (П или М) не получают преимущества. Таким образом, рисунок 2 демонстрирует ситуацию, в которой эффективность всех подходов выравнивается из-за идеальной однородности вычислительной среды и отсутствии погрешности в статистической выборке.

Интерпретация: данный сценарий служит базовой точкой проверки корректности модели – полученные результаты подтверждают корректность реализации алгоритма и отсутствие искусственных искажений: модель в идеальных условиях не дает преимуществ ни одному из методов, что соответствует теоретическим ожиданиям.

7.2. Сценарий 2. Неоднородные узлы, разнотипные задачи и статистические данные – преимущество предсказательных методов. Цель: проверить корректность работы модели и показать, что при различной производительности узлов и сложности задач, а также при добавлении статистических данных предсказательные методы обеспечивают более рациональное распределение по сравнению с динамическим распределением. Это объясняется тем, что введение

статистики позволяет оценивать не только расчетные показатели производительности, но и учитывать их фактические значения.

Для доказательства адекватности разработанной модели многокритериальный метод принятия решения (Парето-оптимизация на первом этапе и линейная свертка на втором) используется в однокритериальном режиме.

Условия эксперимента:

- узлы имеют различную производительность, что отражает гетерогенность вычислительных ресурсов (например, различие в тактовой частоте),
- в поток задач включаются три класса по сложности: простые задачи низкой трудоемкости, задачи средней сложности, ресурсоемкие задачи, требующие максимальной производительности,
- соотношение типов задач предлагается принять как соотношение 34 % : 33 % : 33 % (один из возможных вариантов),
- для каждой пары (тип задачи, узел) накоплена статистика выполнения, позволяющая построить корректный прогноз времени выполнения для приближения к соответствующим фактическим (реальным) значениям,
- система не моделирует текущую загрузку – предполагается, что все узлы доступны и готовы к приему новой задачи (равные начальные условия),
- весовые коэффициенты многокритериального метода заданы так, чтобы принятие решения осуществлялось по одному критерию, как и в однокритериальном методе.

Ожидаемые результаты:

- динамический метод начнет показывать худший результат, так как ориентируется только на текущую загрузку и фиксированные расчетные значения производительности, игнорируя изменчивость реальной производительности устройств,
- однокритериальный предсказательный метод и многокритериальный покажут близкие результаты по времени выполнения и балансировке, так как веса критериев метода (М) заданы таким, образом, чтобы он соответствовал методу (П).

Конфигурация программы для второго сценария представлена на рисунке 3.

На рисунке 4 представлено сравнение среднего времени выполнения задач для трех рассматриваемых методов распределения: Д, П и М. По оси вертикали откладывается время выполнения задач, где меньшие значения соответствуют более эффективной работе системы.

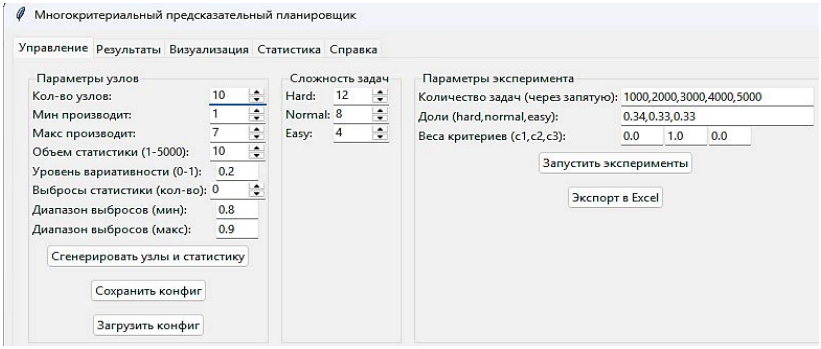


Рис. 3. Окно конфигурации сценария 2

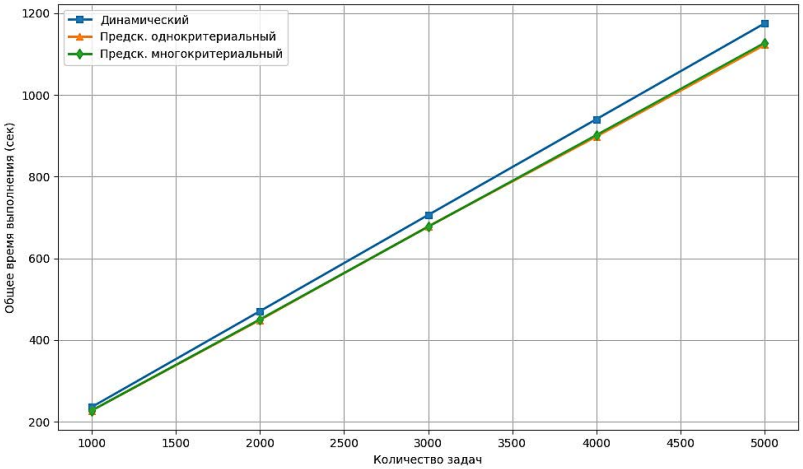


Рис. 4. График сравнения времени выполнения задач для сценария 2 (меньше – лучше)

На рисунке 5 приведено сравнение степени равномерности загрузки вычислительных узлов при использовании трех методов распределения задач. По вертикальной оси отложена разница между максимальной и минимальной загрузкой узлов, характеризующая степень дисбаланса системы: чем меньше значение, тем более равномерно распределена нагрузка.

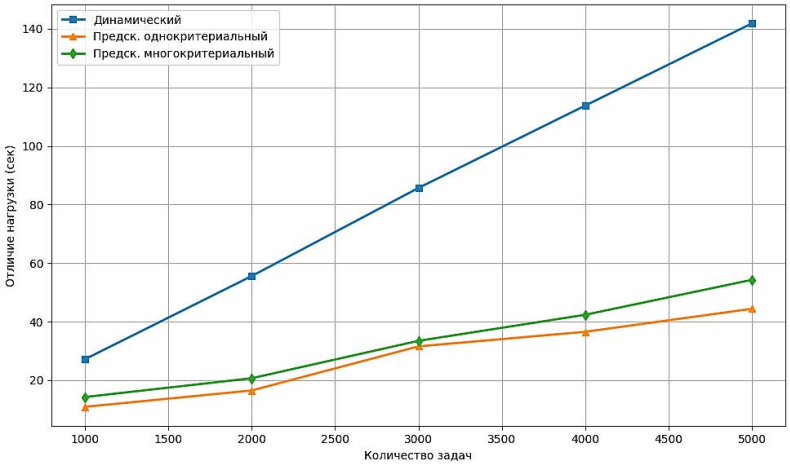


Рис. 5. График сравнения балансировки нагрузки узлов для сценария 2 (меньше – лучше)

Так как на рисунках 4, 5 представлены результаты в абсолютных значениях, для лучшей наглядности предлагается оценивать методы между собой в относительных значениях. Эти значения получаются следующим образом:

$$r_{\text{отн}} = 100 - \frac{r_{\text{абс}}(\text{П}) \cdot 100}{r_{\text{абс}}(\text{Д})},$$

где $r_{\text{отн}}$ – относительное отличие методов (%), $r_{\text{абс}}$ – абсолютное значение показателя (время обработки задач, равномерность загрузки) для рассматриваемых методов.

На рисунках 6, 7 показано относительное отличие предсказательных методов от динамического по двум вышеупомянутым показателям: время выполнения и равномерность.

Из представленных данных видно, что динамический метод демонстрирует наибольшее время выполнения и обеспечивает наихудшую балансировку среди рассматриваемых методов. Это обусловлено тем, что данный метод принимает решение исключительно на основе текущей загрузки узлов и не использует накопленную статистическую информацию о фактической производительности. Предсказательные методы (П и М), наоборот, показывают меньшие значения времени выполнения задач и более сбалансированное распределение.

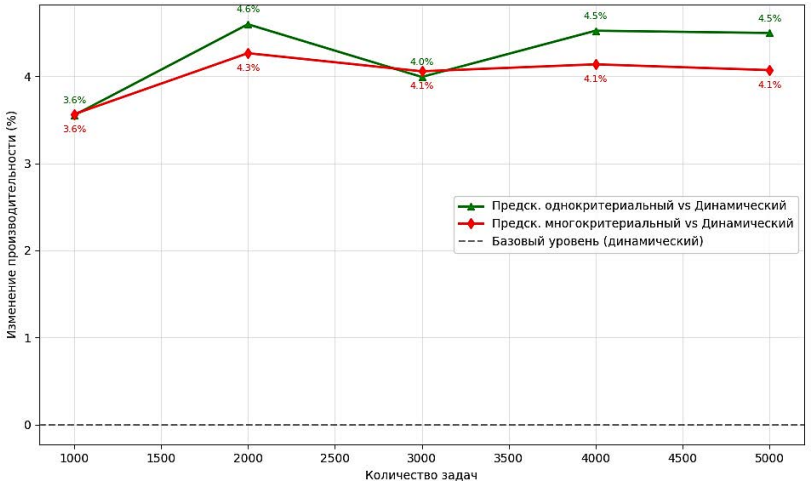


Рис. 6. График относительного изменения времени выполнения для сценария 2 (больше – лучше)

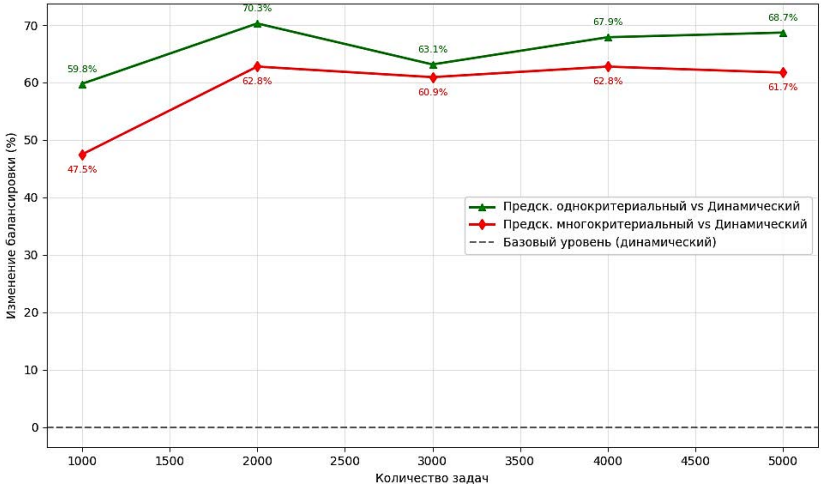


Рис. 7. График относительного изменения балансировки нагрузки для сценария 2 (больше – лучше)

Стоит отметить, что различия между предсказательным однокритериальным и многокритериальным методами в данном сценарии минимальны. Это связано с выбором весовых коэффициентов критериев в функции агрегированной оценки в методе (M). В рассматриваемой конфигурации использовано значение

кортежа весов $w = \langle 0, 1, 0 \rangle$, при котором единственным значимым критерием становится прогнозное время выполнения задачи. В результате оба предсказательных метода (П и М) выбирают одни и те же узлы или узлы близкие по характеристикам для размещения задач и обеспечивают сопоставимый уровень эффективности.

Статистика необходима для оценки реальной производительности, она может показать, что производительность узла изменилась по тем или иным причинам (например, из-за повышения фоновой нагрузки выделенные вычислительные ресурсы (оперативная память, файл подкачки) уменьшились, или из-за повышения температуры могла снизиться тактовая частота процессора).

7.3. Сценарий 3. Нестабильные условия – влияние достоверности прогноза.

Цель: выявить различия между предсказательными методами при ухудшении качества прогноза и наличии выбросов в статистических данных.

Условия эксперимента:

- в статистику выполнения задач добавляются случайные выбросы,
- узлы сохраняют различную производительность,
- для многокритериального метода добавляется критерий достоверности прогноза с весом $c_3 > 0$.

Ожидаемые результаты:

- однокритериальный предсказательный метод (П) начнет назначать задачи на узлы с недостоверной статистикой, что ведет росту общего времени выполнения и может способствовать перегрузке отдельных узлов;
- многокритериальный метод (М) учитывает достоверность прогноза и частично скомпенсирует выбросы, сохранив сбалансированное распределение;
- отличие в эффективности между П и М становится значимой: многокритериальный метод обеспечит наименьшее время выполнения, однако «пожертвует» равномерной загрузкой узлов, что объясняется весовыми коэффициентами.

Конфигурация программы для третьего сценария представлена на рисунке 8.

Ниже представлены результаты сравнения предлагаемого и известных методов для сценария 3 по нескольким критериям - время выполнения задач (рисунок 9), балансировка нагрузки (рисунок 10), относительное изменение времени выполнения задач (рисунок 11), относительное изменение балансировки нагрузки (рисунок 12).

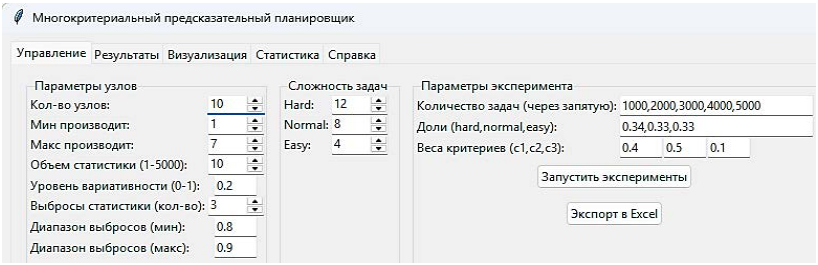


Рис. 8. Окно конфигурации сценария 3

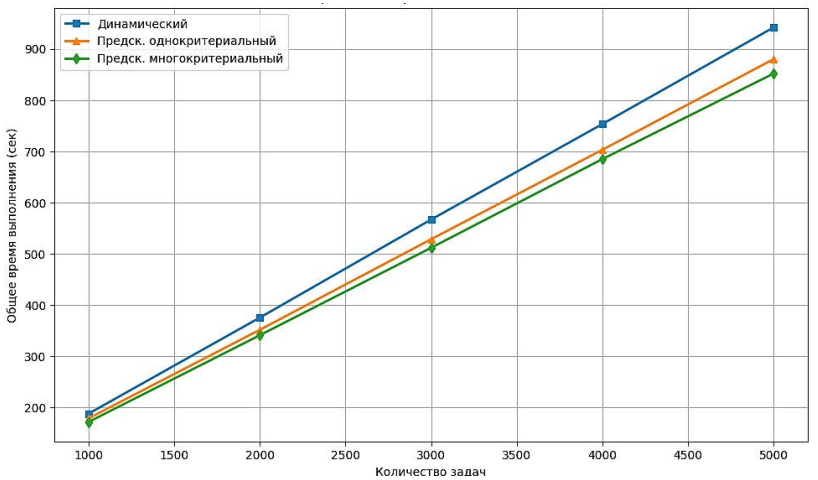


Рис. 9. График сравнения времени выполнения задач для сценария 3 (меньше – лучше)

Переход от однокритериального к многокритериальному предсказательному подходу обусловлен необходимостью учитывать не только текущую загруженность узла, но и контекстные характеристики системы: прогнозное время выполнения и достоверность прогноза, или др. параметры (например, энергопотребление узла, приоритет задач и т. д).

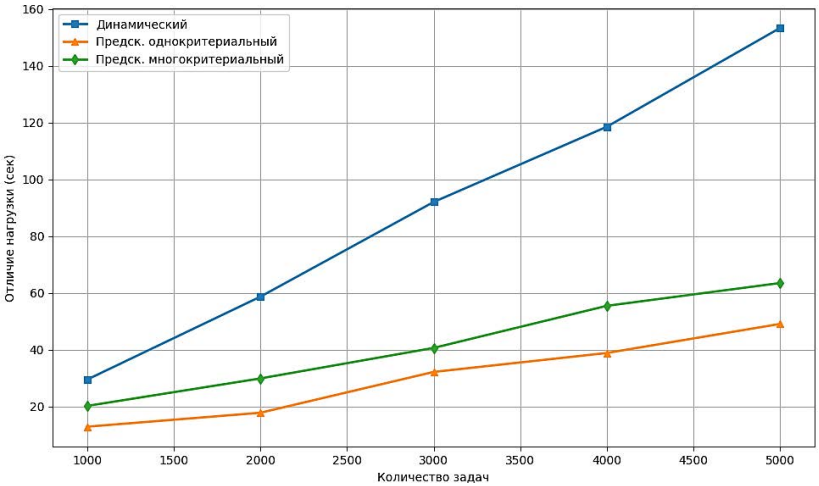


Рис. 10. График сравнения балансировки нагрузки узлов для сценария 3 (меньше – лучше)

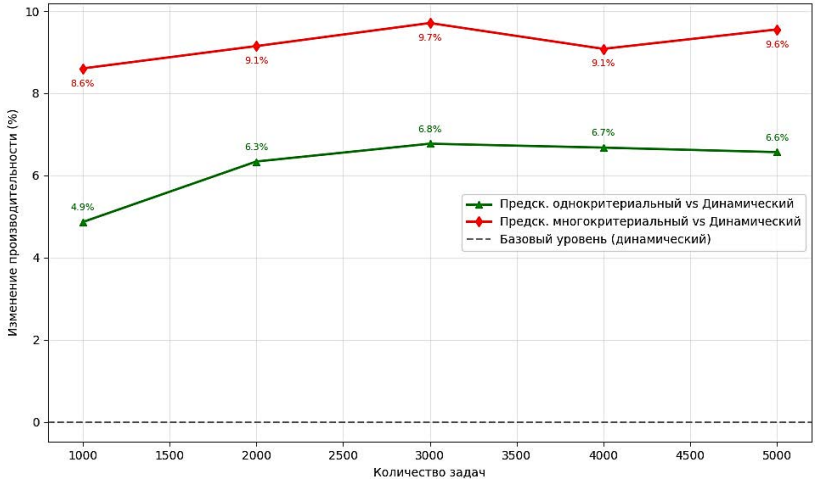


Рис. 11. График относительного изменения времени выполнения для сценария 3 (больше – лучше)

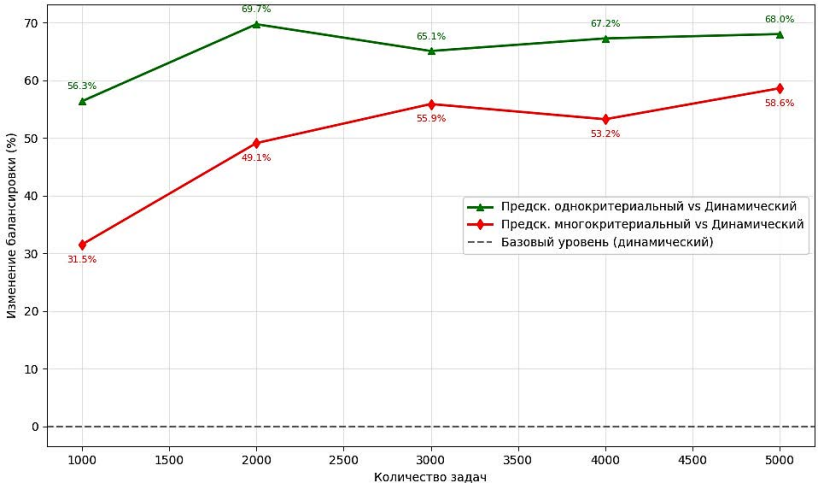


Рис. 12. График относительного изменения балансировки нагрузки для сценария 3 (больше – лучше)

8. Анализ результатов моделирования. Проведенные вычислительные эксперименты подтвердили корректность и эффективность предложенной многокритериальной модели принятия решений в предсказательном методе распределения ресурсов. Результаты сценариев демонстрируют последовательную эволюцию поведения алгоритмов при усложнении условий функционирования системы.

Сценарий 1 (идеальные условия) показал, что при равномерной производительности узлов и стабильной статистике все три метода (Д, П, М) обеспечивают одинаковые результаты. Это указывает на корректность реализации алгоритмов и отсутствие искажений при базовых условиях. Парето-множество в этом случае включало все узлы, что соответствует теоретическому ожиданию – ни один из узлов не имеет преимущества при одинаковых параметрах. Таким образом, данный сценарий подтвердил адекватность исходных предпосылок и корректность механизма нормализации критериев.

Сценарий 2 (неоднородные узлы и разнотипные задачи) выявил различия между динамическим и предсказательными методами. В условиях гетерогенной производительности узлов динамический метод показал наихудшие результаты по среднему времени выполнения задач. Это объясняется тем, что он ориентируется только на моментную загрузку и не использует историческую информацию. Оба предсказательных метода (П и М) обеспечили более

сбалансированные показатели времени выполнения и загрузки узлов. Их преимущество обусловлено использованием прогнозных данных, позволяющих учитывать фактическую производительность каждого узла. При этом различия между П и М были минимальны, так как в данном сценарии достоверность прогноза оставалась высокой и влияние дополнительного критерия было незначительным.

Сценарий 3 (нестабильные условия) стал ключевым для оценки преимуществ многокритериального подхода. Введение статистических выбросов и ухудшение качества прогнозов показали, что метод (П) теряет эффективность – наблюдается рост суммарного времени выполнения. Метод (М), наоборот, демонстрирует адаптивность за счет учета достоверности прогноза: задачи перераспределяются в пользу узлов с более надежной статистикой, что снижает риск перегрузки и повышает точность прогноза. При этом наблюдается небольшое снижение равномерности загрузки, что является допустимой «платой» за повышение точности и предсказуемости выполнения задач.

Таким образом, сравнительный анализ показал, что:

1. В простых условиях все методы эквивалентны по эффективности.
2. В условиях гетерогенности преимущество имеют предсказательные методы, использующие накопленную статистику.
3. При наличии нестабильных данных только многокритериальный подход сохраняет сбалансированные показатели производительности.

Количественные результаты, представленные на графиках, подтверждают, что метод (М) обеспечивает снижение среднего времени выполнения задач на 8-10% и улучшение равномерности распределения нагрузки на 60-70% по сравнению с динамическим методом при ухудшении достоверности статистики. Это свидетельствует о практической значимости введения критерия достоверности прогноза и подтверждает его роль в обеспечении устойчивого функционирования вычислительной системы при изменяющихся условиях.

Итогом моделирования является доказательство того, что интеграция многокритериального выбора в предсказательный метод распределения ресурсов позволяет повысить качество решений и адаптивность планировщика без существенного ухудшения других показателей (равномерности нагрузки на узлы и вычислительной сложности алгоритма).

9. Заключение. В работе предложен усовершенствованный предсказательный метод распределения вычислительных ресурсов, основанный на многокритериальной модели принятия решений, который учитывает не только прогнозируемое время выполнения и загрузку узла, но и достоверность прогноза, отражающую соответствие прогнозируемых и фактических данных.

1. Построена многокритериальная модель принятия решений, определены показатели эффективности работы вычислительной системы: время обработки задач и равномерность распределения нагрузки.
2. Заданы критерии принятия решений, которые предложено расширить за счет нового, характеризующего достоверность прогнозирования.
3. Предложен подход к решению задачи многокритериальной оптимизации: *1 этап* – Парето-оптимизации; *2 этап* – выбор единственного решения на основе метода агрегирования (линейной свертки).
4. Разработан подробный пошаговый алгоритм многокритериального принятия решений, который является основным этапом предсказательного метода распределения ресурсов в вычислительных системах.
5. Проведены экспериментальные исследования показателей эффективности рассматриваемых методов распределения ресурсов. Результаты моделирования подтвердили, что при усложнении условий функционирования системы предсказательный метод с многокритериальным принятием решений обеспечивает более эффективное распределение задач, позволяя компенсировать влияние выбросов в статистике. Введение критерия достоверности позволяет улучшить выбранный показатель эффективности. Созданный инструментарий моделирования дает возможность гибкой настройки параметров для определения области целесообразного применения предложенного метода.

Предложенный предсказательный метод распределения ресурсов в вычислительных системах на основе многокритериальной модели принятия решений может быть использован как основа для создания интеллектуальных систем управления вычислительными кластерами, способных адаптироваться к изменяющимся условиям в реальном времени.

Дальнейшие исследования предполагается проводить для определения области целесообразного применения предложенного метода распределения. В частности, планируется исследовать

показатели эффективности метода в зависимости от процентного соотношения типов задач, характера поступления задач, вариативности и выбросов статистики и т.д., а также ограничения по его использованию.

Литература

1. Hussain H., Malik S.-U.-R., Hameed A., Khan S.-U., Bickler G., Min-Allah N., Qureshi M., et al. A survey on resource allocation in high performance distributed computing systems // Parallel Computing. 2013. vol. 39. no. 11. pp. 709–736. DOI: 10.1016/j.parco.2013.09.009.
2. Coleman J., Bhaskar K. PISA: An Adversarial Approach to Comparing Task Graph Scheduling Algorithms // IEEE International Parallel and Distributed Processing Symposium. 2025. pp. 54–66. DOI: 10.1109/IPDPS64566.2025.00014.
3. Topcuoglu H., Hariri S., Wu M-Y. Performance-effective and low-complexity task scheduling for heterogeneous computing // IEEE Transactions on Parallel and Distributed Systems. 2002. vol. 13. no. 3. pp. 260–274. DOI: 10.1109/71.993206.
4. Shim H. A demand-centered scheduling framework for shared supercomputing resources: modeling, metrics, and case insights // Scientific Reports. 2025. vol. 15. DOI: 10.1038/s41598-025-02353-9.
5. Свиридкин Д.О., Скороходов В.А. О задаче размещения потребителей в сетях с распределением потока. I. NP-полнота // Известия высших учебных заведений. Северо-Кавказский регион. Естественные науки. 2017. № 3-1(195-1). С. 36–41. DOI: 10.23683/0321-3005-2017-3-1-36-41.
6. Moly M.-I., Hossain Md.-A., Lecturer S., Roy O. Load Balancing Approach and Algorithm in Cloud Computing Environment // American Journal of Engineering Research (AJER). 2019. vol. 8. no. 04. pp. 99–105.
7. Клименко А.Б. Математическая модель и эвристические методы организации распределенных вычислений в системах интернета вещей // Компьютерные исследования и моделирование. 2025. Т. 17. № 5. С. 851–870. DOI: 10.20537/2076-7633-2025-17-5-851-870.
8. Wijaya C., Wiryasaputra R., Huang C.-Y., Tanato J., Yang C.-T. Load Balancing Algorithm in a Software-Defined Network Environment with Round Robin and Least Connections. Smart Grid and Internet of Things // 7th EAI International Conference. SGIoT. 2023. vol. 557. pp. 148–157. DOI: 10.1007/978-3-031-55976-1_15.
9. Albalawi N.-S. Dynamic scheduling strategies for cloud-based load balancing in parallel and distributed systems // J Cloud Comp. 2025. vol. 14. DOI: 10.1186/s13677-025-00757-6.
10. Sana M.-U., Li. Z. Efficiency aware scheduling techniques in cloud computing: a descriptive literature review // PeerJ Computer Science. 2021. vol. 7. DOI: 10.7717/peerj-cs.509.
11. Alfahid A., Lhioui C., Asklany S., et al. Peer-driven task scheduling and resource allocation for enhanced performance in industrial IoT systems // Scientific Reports. 2025. vol. 15. DOI: 10.1038/s41598-025-98910-3.
12. Li Y., Zhang Z., Ding Z. Optimization scheduling of microgrid cluster based on improved moth-flame algorithm // Energy Inform. 2024. vol. 7. DOI: 10.1186/s42162-024-00418-z.
13. Грибова В.В., Шалфеева Е.А., Филаретов В.Ф., Зуев А.В., Юхимец Д.А. Метод интеллектуального планирования миссий автономных подводных аппаратов // Информатика и автоматизация. 2025. Т. 24. № 5. С. 1257–1283. DOI: 10.15622/ia.24.5.1.

14. Писковский В.О., Лычева Е.О., Могиленец В.М. Прогнозирование временных характеристик прикладных сетевых сервисов // Вестник Московского университета. Серия № 15. Вычислительная математика и кибернетика. 2025. № 3. С. 62–73. DOI: 10.55959/MSU/0137–0782–15–2025–49–3–62–73.
15. Ламановский М.Н., Лавров Д.Н. Балансировка нагрузки облачных вычислений // Математические структуры и моделирование. 2024. № 2(70). С. 87–99. DOI: 10.24147/2222-8772.2024.2.87-99.
16. Клименко А.Б. Ресурсно-ориентированная технология организации информационного процесса распределения вычислительных ресурсов при интеграции концепций Интернета вещей и краевых вычислений // Моделирование, оптимизация и информационные технологии. 2025. № 13(3). URL: https://moitvivt.ru/ru/journal/pdf?id=2038. DOI: 10.26102/2310-6018/2025.50.3.045.
17. Sanjalawe Y., Al-Emari S., Fraihat S., et al. AI-driven job scheduling in cloud computing: a comprehensive review // Artificial Intelligence Review. 2025. vol. 58. DOI: 10.1007/s10462-025-11208-8.
18. Klimenko A., Elmekeev M. Case study of Lamarckian and Baldwin Evolution Principles Application to the Computations Planning in Resource-Constrained Ad-Hoc Networks // 18th International Conference on Management of Large-Scale System Development (MLSD). 2025. pp. 1–6. DOI: 10.1109/MLSD65526.2025.11220680.
19. Трушкин Е.С., Фрейман В.И. Предсказательный метод распределения ресурсов в вычислительных системах // Информационные технологии и вычислительные системы. 2026. № 1. С. 133–144. DOI: 10.14357/20718632260112.
20. Кулешов С.В., Зайцева А.А., Шальнев И.О. Подход к реализации распределенной системы виртуальных машин для самоорганизующихся сетей // Информационно-управляющие системы. 2019. № 5. С. 30–37. DOI: 10.31799/1684-8853- 2019-5-30-37.
21. Дубинин Р.С., Темников Д.О. Методы балансировки нагрузки между микросервисами в облачной среде // Международный научно-исследовательский журнал. 2025. № 8(158). DOI: 10.60797/IRJ.2025.158.27.
22. Клейман Л.А. Методика принятия решений в задаче диагностики элементов информационно-управляющих систем // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2021. № 38. С. 90–109. DOI: 10.15593/2224-9397/2021.2.05.
23. Barrera-Garcia J., Cisternas-Caneo F., et al. Feature Selection Problem and Metaheuristics: A Systematic Literature Review about Its Formulation, Evaluation and Applications // MDPI Biomimetics. 2024. vol. 9. no. 1. DOI: 10.3390/biomimetics9010009.
24. Idrissi A., Alaoui A. A Multi-Criteria Decision Method in the DBSCAN Algorithm for Better Clustering // International Journal of Advanced Computer Science and Applications (IJACSA). 2016. vol. 7 no. 2. pp. 377–384. DOI: 10.14569/IJACSA.2016.070252.
25. Helfrich S., Herzel A., Ruzika S., et al. Using scalarizations for the approximation of multiobjective optimization problems: towards a general theory // Math Meth Oper Res. 2023. vol. 100. pp. 27–63. DOI: 10.1007/s00186-023-00823-2.
26. Dehnad P., Bidgoli A.-A., Rahnamayan S. Beyond the Pareto Front: Utilizing the Entire Population for Decision-Making in Evolutionary Machine Learning // Mathematics. 2025. vol. 13. no. 16. DOI: 10.3390/math13162579.
27. Колесников В.Л., Бракович А.И., Жук Я.А. Решение многокритериальных задач, оптимальных по Парето // Труды БГТУ. Физико-математические науки

и информатика. 2014. № 6. URL: https://cyberleninka.ru/article/n/reshenie-mnogokriterialnyh-zadach-optimalnyh-po-pareto (дата обращения: 29.10.2025).
28. Подиновский В.В., Ногин В.Д. Парето-оптимальные решения многокритериальных задач: 2-е изд., испр. и доп. М.: ФИЗМАТЛИТ. 2007. 256 с.
29. Menzies T., Saint-Hilary G., Mozgunov P. A comparison of various aggregation functions in multi-criteria decision analysis for drug benefit–risk assessment // Statistical Methods in Medical Research. 2022. vol. 31. no. 5. pp. 899–916. DOI: 10.1177/09622802211072512.
30. Yang F., Li X., Liu Q., Li X., Li Z. Learning-Based Hierarchical Decision-Making Framework for Automatic Driving in Incompletely Connected Traffic Scenarios // MDPI Sensors. 2024. vol. 24. no. 8. DOI: 10.3390/s24082592.
31. Chen S., Liu J., et al. A linguistic multi-criteria decision making approach based on logical reasoning // Information Sciences. 2014. vol. 258. pp. 266–276. DOI: 10.1016/j.ins.2013.08.040.
32. Hullermeier E., Słowinski R. Preference learning and multiple criteria decision aiding: differences, commonalities, and synergies – part II // 4OR. 2024. vol. 22. pp. 313–349. DOI: 10.1007/s10288-023-00561-5.
33. Трушкин Е.С., Гаврилов А.В., Фрейман В.И. Математическое и имитационное моделирование для оценки производительности коммуникационных устройств вычислительных, информационно-управляющих и телекоммуникационных систем // Вестник Пермского национального исследовательского политехнического университета. Электротехника, информационные технологии, системы управления. 2025. № 53. С. 129–156. DOI: 10.15593/2224-9397/2025.1.07.

Трушкин Егор Сергеевич — аспирант, кафедра «Автоматика и Телемеханика», Пермский национальный исследовательский политехнический университет (ПНИПУ). Область научных интересов: вычислительные системы и информационные технологии. Число научных публикаций — 6. egor.s.trushkin@gmail.com; Комсомольский проспект, 29, 614990, Пермь, Россия; р.т.: +7(902)797-8390.

Фрейман Владимир Исаакович — д-р техн. наук, профессор, почетный работник сферы образования Российской Федерации, профессор кафедры, кафедра «Автоматика и Телемеханика», Пермский национальный исследовательский политехнический университет (ПНИПУ). Область научных интересов: вычислительные системы, информационные технологии, техническая диагностика, проектирование, управление и мониторинг телекоммуникационных систем и сетей, помехоустойчивое кодирование, цифровая обработка сигналов, теория принятия решений. Число научных публикаций — 230. vifrejman@pstu.ru; Комсомольский проспект, 29, 614990, Пермь, Россия; р.т.: +7(342)239-1816.

DOI 10.15622/ia.25.2.10

E. TRUSHKIN, V. FREYMAN

A PREDICTIVE METHOD OF RESOURCE ALLOCATION IN COMPUTING SYSTEMS BASED ON A MULTICRITERIA DECISION-MAKING MODEL

Trushkin E., Freyman V. **A Predictive Method of Resource Allocation in Computing Systems Based on a Multicriteria Decision-Making Model.**

Abstract. Modern computing systems typically operate under heterogeneous and variable load conditions. Efficient distribution of computing resources is an important tool for ensuring high performance (e.g., productivity, reliability, stability). Therefore, developing task distribution methods that can simultaneously improve several metrics is a pressing issue. The approach proposed in this paper is an extension of the predictive resource allocation method. This is achieved through the introduction of a multi-criteria decision-making model that includes the predicted execution time, the current node load, and the reliability of the forecast, assessed by the statistics of discrepancies between actual and predicted values. The object of this study is computing systems with heterogeneous nodes processing task flows of variable complexity. The subject of this study is the models and algorithms for predictive distribution of computing resources based on multi-criteria decision making. The objective of the study is to improve the efficiency and stability of heterogeneous computing systems by using a more substantiated mechanism for node selection based on a set of criteria. A review of existing dynamic and predictive distribution methods is provided, and their advantages, disadvantages, and limitations for effective application are identified. A multi-criteria decision-making model was developed that implements the construction of a set of Pareto-optimal solutions and an arbitration procedure. Software simulations were conducted under various system operating scenarios, including conditions with reduced statistical reliability. The results of the study showed that the proposed predictive method based on multi-criteria decision-making reduces the average task execution time and improves node load uniformity compared to known approaches. These results are proposed for use in the construction of heterogeneous computing systems with adaptive resource management systems.

Keywords: computing system, resource allocation methods, Pareto optimality, multi-criteria decision making, load forecasting, statistical data.

References

1. Hussain H., Malik S.-U.-R., Hameed A., Khan S.-U., Bickler G., Min-Allah N., Qureshi M., et al. A survey on resource allocation in high performance distributed computing systems. Parallel Computing. 2013. vol. 39. no. 11. pp. 709–736. DOI: 10.1016/j.parco.2013.09.009.
2. Coleman J., Bhaskar K. PISA: An Adversarial Approach to Comparing Task Graph Scheduling Algorithms. IEEE International Parallel and Distributed Processing Symposium. 2025. pp. 54–66. DOI: 10.1109/IPDPS64566.2025.00014.
3. Topcuoglu H., Hariri S., Wu M-Y. Performance-effective and low-complexity task scheduling for heterogeneous computing. IEEE Transactions on Parallel and Distributed Systems. 2002. vol. 13. no. 3. pp. 260–274. DOI: 10.1109/71.993206.
4. Shim H. A demand-centered scheduling framework for shared supercomputing resources: modeling, metrics, and case insights. Scientific Reports. 2025. vol. 15. DOI: 10.1038/s41598-025-02353-9.

5. Sviridkin D., Skorokhodov V. [On the task of placing consumers in flow-distributed networks. I. NP-completeness]. Izvestiya vysshikh uchebnykh zavedeniy. Severo-Kavkazskiy region. Estestvennye nauki – News of higher educational institutions. The North Caucasus region. Natural sciences. 2017. no. 3-1(195-1). pp. 36–41. DOI: 10.23683/0321-3005-2017-3-1-36-41. (In Russ.).
6. Moly M.-I., Hossain Md.-A., Lecturer S., Roy O. Load Balancing Approach and Algorithm in Cloud Computing Environment. American Journal of Engineering Research (AJER). 2019. vol. 8. no. 04. pp. 99–105.
7. Klimenko A. [Mathematical model and heuristic methods for organizing distributed computing in Internet of Things systems]. Kompyuternye issledovaniya i modelirovanie – Computer research and modeling. 2025. vol. 17. no. 5. pp. 851–870. DOI: 10.20537/2076-7633-2025-17-5-851-870. (In Russ.).
8. Wijaya C., Wiryasaputra R., Huang C.-Y., Tanato J., Yang C.-T. Load Balancing Algorithm in a Software-Defined Network Environment with Round Robin and Least Connections. Smart Grid and Internet of Things. 7th EAI International Conference. SGIoT. 2023. vol. 557. pp. 148–157. DOI: 10.1007/978-3-031-55976-1_15.
9. Albalawi N.-S. Dynamic scheduling strategies for cloud-based load balancing in parallel and distributed systems. J Cloud Comp. 2025. vol. 14. DOI: 10.1186/s13677-025-00757-6.
10. Sana M.-U., Li. Z. Efficiency aware scheduling techniques in cloud computing: a descriptive literature review. PeerJ Computer Science. 2021. vol. 7. DOI: 10.7717/peerj-cs.509.
11. Alfahid A., Lhioui C., Asklany S., et al. Peer-driven task scheduling and resource allocation for enhanced performance in industrial IoT systems. Scientific Reports. 2025. vol. 15. DOI: 10.1038/s41598-025-98910-3.
12. Li Y., Zhang Z., Ding Z. Optimization scheduling of microgrid cluster based on improved moth-flame algorithm. Energy Inform. 2024. vol. 7. DOI: 10.1186/s42162-024-00418-z.
13. Gribova V., Shalfeeva E., Filaretov V., Zuev A., Yukhimets D. [Intelligent mission planning method for autonomous underwater vehicles]. Informatika i avtomatizatsiya – Informatics and Automation. 2025. vol. 24. no. 5. pp. 1257–1283. DOI: 10.15622/ia.24.5.1. (In Russ.).
14. Piskovsky V., Lycheva E., Mogilenets V. [Forecasting the time characteristics of applied network services]. Vestnik Moskovskogo universiteta. Seriya no. 15. Vychislitel'naya matematika i kibernetika – Bulletin of the Moscow University. Series no. 15. Computational mathematics and cybernetics. 2025. no. 3. pp. 62–73. DOI: 10.55959/MSU/0137–0782–15–2025–49–3–62–73. (In Russ.).
15. Lamanovsky M., Lavrov D. [Load balancing of cloud computing]. Matematicheskie struktury i modelirovanie – Mathematical structures and modeling. 2024. no. 2(70). pp. 87–99. DOI: 10.24147/2222-8772.2024.2.87-99. (In Russ.).
16. Klimenko A. [Resource-oriented technology for organizing the information process of computing resource allocation when integrating the concepts of the Internet of Things and edge computing]. Modelirovanie, optimizatsiya i informatsionnye tekhnologii – Modeling, optimization, and information technology. 2025. no. 13(3). DOI: 10.26102/2310-6018/2025.50.3.045. (In Russ.).
17. Sanjalawe Y., Al-Emari S., Fraihat S., et al. AI-driven job scheduling in cloud computing: a comprehensive review. Artificial Intelligence Review. 2025. vol. 58. DOI: 10.1007/s10462-025-11208-8.
18. Klimenko A., Elmekeev M. Case study of Lamarckian and Baldwin Evolution Principles Application to the Computations Planning in Resource-Constrained Ad-Hoc Networks. 18th International Conference on Management of Large-Scale System Development (MLSD). 2025. pp. 1–6. DOI: 10.1109/MLSD65526.2025.11220680.

19. Trushkin E., Freyman V. [Predictive method of resource allocation in computing systems]. Informacionnye tehnologii i vychislitel'nye sistemy – Information technologies and computing systems. 2026. no 1. pp. 133–144. DOI: 10.14357/20718632260112. (In Russ.).
20. Kuleshov S., Zaitseva A., Shalnev I. [An approach to the implementation of a distributed virtual machine system for self-organizing networks]. Informatsionno-upravlyayushchie sistemy – Information management systems. 2019. no. 5. pp. 30–37. DOI: 10.31799/1684-8853- 2019-5-30-37. (In Russ.).
21. Dubinin R., Temnikov D. [Load balancing methods between microservices in a cloud environment]. Mezhdunarodnyy nauchno-issledovatel'skiy zhurnal – International Scientific Research Journal. 2025. no. 8(158). DOI: 10.60797/IRJ.2025.158.27. (In Russ.).
22. Kleiman L. [Decision-making methodology in the task of diagnosing elements of information management systems]. Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotekhnika, informatsionnye tekhnologii, sistemy upravleniya – Bulletin of the Perm National Research Polytechnic University. Electrical engineering, information technology, control systems. 2021. no. 38. pp. 90–109. DOI: 10.15593/2224-9397/2021.2.05. (In Russ.).
23. Barrera-Garcia J., Cisternas-Caneo F., et al. Feature Selection Problem and Metaheuristics: A Systematic Literature Review about Its Formulation, Evaluation and Applications. MDPI Biomimetics. 2024. vol. 9. no. 1. DOI: 10.3390/biomimetics9010009.
24. Idrissi A., Alaoui A. A Multi-Criteria Decision Method in the DBSCAN Algorithm for Better Clustering. International Journal of Advanced Computer Science and Applications (IJACSA). 2016. vol. 7 no. 2. pp. 377–384. DOI: 10.14569/IJACSA.2016.070252.
25. Helfrich S., Herzel A., Ruzika S., et al. Using scalarizations for the approximation of multiobjective optimization problems: towards a general theory. Math Meth Oper Res. 2023. vol. 100. pp. 27–63. DOI: 10.1007/s00186-023-00823-2.
26. Dehnad P., Bidgoli A.-A., Rahnamayan S. Beyond the Pareto Front: Utilizing the Entire Population for Decision-Making in Evolutionary Machine Learning. Mathematics. 2025. vol. 13. no. 16. DOI: 10.3390/math13162579.
27. Kolesnikov V., Brakovich A., Zhuk Ya. [Solution of multi-criteria Pareto optimal problems]. Trudy BGTU. Fiziko-matematicheskie nauki i informatika – Proceedings of BSTU. Physical and mathematical sciences and computer science. 2014. no. 6. (In Russ.).
28. Podinovskiy V., Nogin V. Pareto-optimarnye resheniya mnogokriterial'nykh zadach: 2-e izd., ispr. i dop [Pareto-optimal solutions of multi-criteria problems: 2nd ed., ispr. and add.]. Moscow: FIZMALIT, 2007. 256 p. (In Russ.).
29. Menzies T., Saint-Hilary G., Mozgunov P. A comparison of various aggregation functions in multi-criteria decision analysis for drug benefit–risk assessment. Statistical Methods in Medical Research. 2022. vol. 31. no. 5. pp. 899–916. DOI: 10.1177/09622802211072512.
30. Yang F., Li X., Liu Q., Li X., Li Z. Learning-Based Hierarchical Decision-Making Framework for Automatic Driving in Incompletely Connected Traffic Scenarios. MDPI Sensors. 2024. vol. 24. no. 8. DOI: 10.3390/s24082592.
31. Chen S., Liu J., et al. A linguistic multi-criteria decision making approach based on logical reasoning. Information Sciences. 2014. vol. 258. pp. 266–276. DOI: 10.1016/j.ins.2013.08.040.
32. Hullermeier E., Słowinski R. Preference learning and multiple criteria decision aiding: differences, commonalities, and synergies – part II. 4OR. 2024. vol. 22. pp. 313–349. DOI: 10.1007/s10288-023-00561-5.

33. Trushkin E., Gavrilov A., Freiman V. [Mathematical and simulation modeling for evaluating the performance of communication devices of computing, information management and telecommunication systems]. Vestnik Permskogo natsional'nogo issledovatel'skogo politekhnicheskogo universiteta. Elektrotekhnika, informatsionnye tekhnologii, sistemy upravleniya – Bulletin of the Perm National Research Polytechnic University. Electrical engineering, information technology, control systems. 2025. no. 53. pp. 129–156. DOI: 10.15593/2224-9397/2025.1.07. (In Russ.).

Trushkin Egor — Graduate student, Department of Automation and Telemechanics, Perm National Research Polytechnic University (PNRPU). Research interests: computing systems and information technology. The number of publications — 6. egor.s.trushkin@gmail.com; 29, Komsomolsky Ave., 614990, Perm, Russia; office phone: +7(902)797-8390.

Freyman Vladimir — Ph.D., Dr.Sci., Professor, Honored worker of education of the Russian Federation, Professor of the Department, Department of Automation and Telemechanics, Perm National Research Polytechnic University (PNRPU). Research interests: computing systems; information technologies; technical diagnostics; design, control and monitoring of networks; error-correcting coding; digital signal processing; decision-making. The number of publications — 230. vifrejman@pstu.ru; 29, Komsomolsky Ave., 614990, Perm, Russia; office phone: +7(342)239-1816.