

УДК 519.252

СТОХАСТИЧЕСКИЕ МОДЕЛИ ТРУДОЕМКОСТИ ВЫЧИСЛИТЕЛЬНЫХ ЗАДАЧ. I. ПРИНЦИПЫ ФОРМИРОВАНИЯ, СБОР СТАТИСТИЧЕСКИХ ДАННЫХ, ЗАДАЧИ ИДЕНТИФИКАЦИИ¹

© 2024 г. А.В. Борисов^{а,*}, А.В. Иванов^{а,**}

^а ФИЦ ИУ РАН, Москва, Россия

*e-mail: ABorisov@frccsc.ru

**e-mail: AIvanov@frccsc.ru

Поступила в редакцию 07.06.2023 г.

После доработки 16.08.2023 г.

Принято к публикации 02.10.2023 г.

Изложены результаты первой части исследования, посвященного построению математической модели времени выполнения пользовательских заданий на виртуальных вычислительных узлах. Предполагается, что данное время является случайной величиной, среднее значение и дисперсия которой зависят от ресурсов узла, характеристик пользовательских заданий и значений переменных аппаратно-программной среды. Представлены общие свойства функций, характерные для среднего значения и дисперсии времени выполнения заданий, а также предложены некоторые их частные примеры. Функции средних значений и дисперсий зависят от неизвестных параметров, и построение стохастической модели времени выполнения задания заключается в идентификации этих параметров — построении обобщений оценок максимального правдоподобия по разнородной статистической информации. Содержатся рекомендации по сбору и использованию этой информации: подготовке аппаратно-программной составляющей узла, планированию нагрузочного тестирования и обработке полученных в его процессе статистических данных. Конкретные прикладные примеры применения предложенной методологии будут представлены в следующих частях исследования.

Ключевые слова: стохастическая модель, параметрическое оценивание, М-оценка, задачи невыпуклой и негладкой оптимизации, нагрузочное тестирование

DOI: 10.31857/S0002338824010037, EDN: IXTIMV

STOCHASTIC MODELS FOR TIME COMPLEXITY OF COMPUTING TASKS: I. DEVELOPMENT PRINCIPLES, STATISTICAL DATA MINING, IDENTIFICATION PROBLEMS

© 2024 Andrey V. Borisov^{а,*}, Alexey V. Ivanov^{а,**}

^аFederal Research Center "Computer Science and Control" of Russian Academy of Sciences

*e-mail: ABorisov@frccsc.ru

**e-mail: AIvanov@frccsc.ru

The paper contains the first part of an investigation devoted to the design of the mathematical models for the execution time of user tasks carried out on the virtual calculating nodes. We suppose that the execution time is a random value with the mean and variance depending on the node resources, task parameters, and the current characteristics of the node state. We discover the key features of the mean and variance functions and specify some of their particular cases. Both the mean and variance functions depend on the unknown parameters, and the design of the stochastic model for the time complexity leads to the parameter identification in the form of the generalized maximum likelihood estimates under the heterogeneous statistical information. The paper also contains recommendations concerning the gathering and subsequent usage of this information: the node testbed preparation, stress test planning, and the obtained data processing. The specific illustrating examples of the proposed mathematical model will be presented in the subsequent parts of the investigation.

Keywords: stochastic model, parameter estimation, M-estimate, nonconvex and nonsmooth optimization problems, stress testing

¹ Работа выполнена с использованием инфраструктуры Центра коллективного пользования "Высокопроизводительные вычисления и большие данные" (ЦКП "Информатика") ФИЦ ИУ РАН (г. Москва).

Введение. Стремительное развитие средств вычислительной техники различной производительности в совокупности с повсеместным проникновением высокоскоростных сетей передачи данных привело к появлению таких новых отраслей услуг, как дополненная и виртуальная реальность [1], Индустриальный Интернет [2], Интернет вещей [3], Интернет военных вещей [4], Интернет транспортных средств [5] и пр. Широкое распространение в ближайшем будущем стандартов мобильной связи 5G/6G обеспечит пользователям возможность доступа к вычислительным мощностям любого уровня вне зависимости от их территориального расположения и без необходимости владения ими, с одной стороны, и даст владельцам компьютерных сетей возможность оперативного распределения нагрузки по территориально разнесенным вычислительным узлам — с другой. Подобные вычислительные сети функционируют в следующих условиях:

- высокая стоимость создания и эксплуатации,
- высокие требования пользователей сети к качеству предоставляемых услуг,
- высокая конкуренция между сетями — поставщиками услуг.

Эти жесткие требования приводят к необходимости оптимизации использования ресурсов компьютерных сетей.

Объектом выполненного исследования являются проблемы распределения вычислительной нагрузки по узлам различной производительности и выделения вычислительных ресурсов различным пользователям. Идеальный результат с точки зрения провайдера — предоставление каждому пользователю минимальных вычислительных ресурсов, достаточных для решения его задач, с учетом обязательного выполнения условий соглашения об уровне обслуживания (*service level agreement, SLA*). Однако достижению этой цели препятствует ряд фундаментальных проблем.

Во-первых, имеет место неопределенность запросов пользователей на выполнение вычислений. Несмотря на то что SLA, заключаемое между пользователем и провайдером, определяет перечень задач, решение которых обеспечивает вычислительная сеть, интенсивность поступления пользовательских заданий заранее неизвестна. Это означает, что количество выделенных ресурсов, необходимых для удовлетворения SLA, может варьироваться в сторону увеличения. Помимо этого трудоемкость отдельных заданий также содержит в себе неопределенность. Причинами этого являются не только различия в объемах заданий. Иногда время выполнения одного и того же задания на одной и той же аппаратно-программной платформе может отличаться в разы на различных входных данных. Примером этого служит численное решение оптимизационных задач [6, 7] при различных начальных условиях, использование алгоритмов машинного обучения [8], компиляция видеоландшафтов компьютерных игр [9] и пр.

Во-вторых, большинство используемого программного обеспечения (ПО) является проприетарным, что исключает возможность его детального алгоритмического анализа в целях определения трудоемкости выполнения той или иной задачи. Относительно процедуры выполнения заданий возможно делать только самые общие выводы, основываясь на знаниях функционирования аналогичных программ с открытым кодом, а также алгоритмов решения того или иного класса задач. В некоторых случаях о времени выполнения заданий существует прямая и косвенная статистическая информация в виде результатов нагрузочного тестирования.

В-третьих, между требованиями пользователей к качеству услуг, зафиксированными SLA, и объектами управления, доступными провайдерам, существует известный разрыв. Дело в том, что пользователи определяют качество своего обслуживания через максимальное время выполнения задач того или иного класса. В то же время для обеспечения выполнения пользовательских заданий провайдеры выделяют определенный объем вычислительных ресурсов: оперативной и дисковой памяти, процессорных ядер и пр. Количество выделенных ресурсов влияет на скорость выполнения заданий, однако не определяет время их выполнения из-за неизвестной трудоемкости.

Цель данной работы — предложить стохастические модели времени выполнения пользовательских заданий на вычислительных узлах, а также методики идентификации параметров этих моделей.

Статья организована следующим образом. Раздел 1 содержит неформальное описание проблем оценивания вероятности нарушения SLA на определенном виртуальном вычислительном узле, заключающегося в превышении допустимого времени выполнения задания. При этом объем пользовательского задания либо известен, либо известен пользовательский профиль, т.е. вероятностное распределение заданий различного объема, характерное для данного пользователя.

В разд. 2 представлены основные принципы выбора моделей времени выполнения заданий. Приведена аргументация в пользу его вероятностного описания, а также предложены полезные свойства, которыми должны обладать выбранные модели. В разд. 3 время выполнения заданий предлагается описывать семейством случайных величин, чьи средние значения M и дисперсия D являются функциями ресурсов узла, характеристик выполняемого задания и переменных аппаратно-программной среды. Подобно производственным функциям [10], в разделе рассмотрены свойства функций M и D по разным переменным, а также предложены некоторые их частные случаи. Здесь же приведены математически корректные формулировки и решения задач оценивания вероятности нарушения SLA.

В разд. 4 предлагается использовать методологию взвешенного М-оценивания [11] для решения задачи идентификации параметров предложенной модели трудоемкости пользовательских заданий. При этом в качестве функции потерь предлагается применять одну из следующих выпуклых функций: квадратичную, модуль или выпуклую кусочно-линейную.

Решение задачи идентификации параметров предполагает наличие необходимой статистической информации. Раздел 5 содержит рекомендации по ее сбору и обработке. Накопление этой информации выполняется в процессе нагрузочного тестирования, проводимого именно на той аппаратно-программной платформе (аппаратное обеспечение плюс общесистемное программное обеспечение (ОПО)), на которой и будет впоследствии функционировать виртуальный вычислительный узел. В то же время реальное специальное программное обеспечение (СПО) может быть заменено некоторым тестовым, и его краткая классификация приведена в разделе. Помимо фиксируемого времени выполнения заданий с различными варьируемыми параметрами: предложен перечень регистрируемых данных, который может быть полезен для определения переменных аппаратно-программной среды. В Заключение содержатся краткие выводы по модели, а также перспективы дальнейших исследований.

1. Описание проблемы. Представлено неформальное описание проблемы характеристики времени выполнения пользовательских задач на вычислительных узлах. Предполагается, что узел является виртуальным и физически располагается на некотором средстве вычислительной техники (СВТ): рабочей станции, сервере, мейнфрейме и пр. Для обеспечения функционирования узла на СВТ выделено определенное количество процессорных ядер, оперативной памяти, дискового пространства и других ресурсов для монопольного использования. На узле развернуто ОПО и СПО. Именно СПО обеспечивает выполнение поступающих на узел пользовательских заданий.

Все задания, которые могут выполняться на исследуемом узле, подразделяются на типы. Задания относятся к одному типу, если они выполняются одним и тем же СПО с помощью одних и тех же алгоритмов. Например, к одному типу относятся задания по сжатию видеоданных, проводимому одной и той же программой-архиватором. Если тем же программным средством выполняется архивация текстовых данных, то это задание относится к другому типу, так как сжатие видео- и текстовой информации осуществляются с помощью разных алгоритмов. В то же время задания сжатия видеoinформации различными программными средствами так же относятся к разным типам, так как используют разное СПО.

Еще одним примером заданий различного типа являются задачи загрузки информации в некоторую базу данных (БД), с одной стороны, и выгрузки данных, полученных в результате некоторого запроса — с другой. В дальнейшем изложении предполагается, что изучаются задачи одного фиксированного типа.

Задания, принадлежащие одному типу, отличаются друг от друга. Эти отличия описываются некоторым вектором параметров. Самой простой характеристикой задания является объем данных, обрабатываемых в процессе его выполнения, например, объем видеофайлов, подлежащих сжатию, число снимков, подлежащих распознаванию, объем информации, подлежащих загрузке в БД, и пр. Тем не менее характеристики задания не исчерпываются объемом данных. Например, в задании обучения нейронной сети в качестве параметров могут выступать:

- число слоев в сети,
- максимальное число узлов в слое,
- объем обучающего банка,
- параметр, определяющий условие остановки процесса обучения.

Как уже было отмечено ранее, время выполнения индивидуальных заданий того или иного типа содержит в себе неопределенность, порожденную различными причинами. Первая из них — “закрытый” характер функционирования ПО. Неопределенность этого сорта проявляется в колебаниях времени даже при выполнении заданий с одинаковыми параметрами.

Вторая причина кроется в неопределенности выбора пользователем того или иного вектора параметров задания и входных данных. Эта неопределенность может быть частично сокращена с помощью формирования так называемого *пользовательского профиля* по исторической информации о запросах данного пользователя на выполнение заданий изучаемого типа. Например, задание на скачивание фильмов из некоторого хранилища в 80% случаев относится к полнометражным фильмам продолжительностью от 90 до 240 мин, в 15% случаев — к короткометражным фильмам продолжительностью от 10 до 30 мин, и в 5% случаев — к документальным фильмам продолжительностью от 40 до 90 мин. Таким образом, пользовательский профиль определяет условные доли задания с различными параметрами.

Для обеспечения оптимизации использования аппаратных ресурсов СВТ при наличии пользовательских SLA необходимо решить следующие практические проблемы.

Проблема 1. Для фиксированных аппаратных ресурсов узла, состояния ОПО и СПО, и пользовательского задания известного типа и объема оценить вероятность нарушения SLA — превышения максимального времени выполнения задания данного типа.

Проблема 2. Для известного пользовательского профиля при фиксированных аппаратных ресурсах узла, состояния ОПО и СПО оценить вероятность нарушения SLA — превышения максимального времени выполнения задания данного типа.

Формальное описание предлагаемых моделей, постановка задачи идентификации их параметров, а также математически корректная постановка и решение представленных выше проблем даны в следующих разделах.

2. Основные принципы выбора моделей. На выбор математических моделей времени выполнения заданий на вычислительных узлах фундаментальное влияние оказывает наличие неопределенности этого времени. Это означает, что повторяющееся выполнение одного и того же задания на одном и том же вычислительном узле будет требовать *разного* времени. В качестве математического аппарата, описывающего данное явление и позволяющего решать для него задачи системного анализа, можно использовать теорию нечетких множеств [12], игровой/гарантирующий подход [13, 14] и пр. Однако наиболее перспективным для этого представляется аппарат теории вероятностей. Таким образом, время выполнения задания будет считаться *случайной* величиной $\tau(\omega)$, а значит, модель, ее описывающая, относится к классу *стохастических*.

Стохастическую модель времени выполнения заданий следует выбирать, руководствуясь следующими принципами.

1. **Адекватность:** модель задает все те особенности действительного времени выполнения, которые необходимы для решения полного спектра последующих задач анализа, идентификации параметров модели и их оптимизации.

2. **Универсальность:** модель предоставляет возможность адекватно описывать время выполнения для широкого набора вычислительных узлов и пользовательских заданий.

3. **Простота:** число параметров модели должно быть, по возможности, минимальным при сохранении требуемой степени адекватности и универсальности.

4. **Гибкость по отношению к априорной информации:** модель настраивается в соответствии с дополнительной доступной информацией о характеристиках СВТ, ОПО, СПО, наличии новых или отсутствии каких-либо факторов, влияющих на время выполнения заданий того или иного типа.

5. **Развитость:** для выбранных моделей существует эффективный математический аппарат и алгоритмическое обеспечение решения всего спектра прикладных задач, связанных с анализом исследуемого времени, идентификацией параметров моделей, а также возможной последующей оптимизацией настройки виртуальных вычислительных узлов.

6. **Возможность применения имитаторов:** на предварительном этапе идентификации параметров модели при сборе статистических данных о времени выполнения заданий возможно использование тестового программного обеспечения (ТПО) (*benchmarks*) — имитаторов пользовательской нагрузки и имитаторов СПО.

3. Стохастические модели трудоемкости: формальное описание, постановка и решение задачи анализа. Согласно выбранному стохастическому подходу, время выполнения пользовательского задания некоторого фиксированного типа предлагается описывать случайной величиной $\tau(\omega)$, имеющей конечное математическое ожидание $M \triangleq E\{\tau\}$ и дисперсию $D \triangleq E\{(\tau(\omega) - M)^2\}$. Обе эти характеристики являются неизвестными функциями

$$M = M_z(x, y), \quad D = D_z(x, y),$$

где x — вектор ресурсов узла, y — вектор параметров задания, z — вектор параметров аппаратно-программной среды узла; развернутое описание данных векторов приведено ниже.

Вектор переменных $x = (x_1, \dots, x_N)$ задает аппаратно-программный состав виртуального вычислительного узла, например: x_1 — число процессорных ядер, x_2 — объем доступной оперативной памяти, x_3 — объем доступного дискового пространства.

Замечание 1. Предлагаемая модель позволяет варьировать набор x в зависимости от состава и возможности настроек виртуального узла, добавляя в него какие-то компоненты, или исключая неактуальные. Например, могут быть добавлены x_4 — объем дискового пространства, зарезервированного для подкачки страниц, x_5 — объем кэша и пр.

Вектор $y = (y_1, \dots, y_M)$ характеризует пользовательские задания фиксированного типа, поступающие на данный вычислительный узел. Компоненты y_m для разных видов заданий могут нести совершенно разный смысл, однако все они влияют на время выполнения. Например, если узел сконфигурирован для выполнения научных вычислений, скажем, для численного решения уравнений математической физики, то возможен следующий вариант: $M = 2$, y_1 — количество временных слоев в численном решении, y_2 — количество узлов пространственной сетки в численном решении.

Другой пример: на узле размещена база данных, и выполняемое на ней пользовательское задание предусматривает ввод/вывод данных и выполнение запросов на решение некоторой информационной задачи. В этом случае $M = 3$, y_1 — объем входных данных в задании, y_2 — объем выходных данных в задании, y_3 — число запросов к базе на выполнение информационных задач в задании.

Вектор $z = (z_1, \dots, z_K)$ содержит текущие параметры аппаратно-программной среды вычислительного узла. Например, если на данном вычислительном узле размещена база данных, то параметрами среды могут выступать следующие характеристики: z_1 — максимальный объем оперативной памяти, доступный на узле, z_2 — текущий объем “полезных” данных, хранящийся в базе, z_3 — общий текущий объем служебной информации, хранящейся в базе данных: журнал транзакций и пр., z_4 — средний объем оперативной памяти вычислительного узла, занимаемый ОПО, и т. д.

Замечание 2. Между векторами x , y и z существуют следующие смысловые различия. Вектор x представляет собой “управление”, доступное провайдеру. Вектор y — “управление”, доступное пользователю. И x , и y могут независимо друг от друга варьироваться в процессе нагрузочного тестирования и сбора статистической информации для последующей идентификации параметров модели. Следует также отметить, что из-за того, что на одном и том же узле могут выполняться задания различных типов, размерность вектора x и множество его возможных значений для данного узла остаются неизменными, в отличие от вектора y , характеристики которого для заданий разного типа различны.

В отличие от x и y параметры z аппаратно-программной среды не управляются напрямую ни провайдером, ни пользователем. Тем не менее его компоненты доступны прямому или косвенному наблюдению в процессе нагрузочного тестирования: информацию о нем можно получить либо путем выполнения служебных команд/запросов, либо путем анализа системных журналов.

Множество $\mathcal{A}$ допустимых значений аргументов (x, y) является ограниченным и, более того, конечным: компоненты x_n , $n = 1, N$ и y_m , $m = 1, M$ могут принимать значения из некоторых конечных множеств. Без ограничения общности предполагается, что множество допустимых значений векторов (x, y) содержится в параллелепипеде $\mathcal{U} = [\underline{x}, \bar{x}] \times [\underline{y}, \bar{y}]$, на котором функции $\mathcal{M}_z(x, y)$ и $\mathcal{D}_z(x, y)$ обладают следующими свойствами.

1. *Неотрицательность*: для любого (x, y, z) верны неравенства

$$\mathcal{M}_z(x, y) > 0, \quad \mathcal{D}_z(x, y) > 0.$$

Данные неравенства очевидны: математическое ожидание и дисперсия любой положительной случайной величины с конечными первыми двумя моментами всегда положительны.

2. *Локальное невозрастание $\mathcal{M}_z(\cdot)$ по x* : для любого фиксированного z существует такое подмножество $\mathcal{U} \subseteq \mathcal{U}$, что для любых $(x', y^*), (x'', y^*) \in \mathcal{U}$ таких, что $x' \leq x''$ покомпонентно, выполняется

$$\mathcal{M}_z(x', y^*) \geq \mathcal{M}_z(x'', y^*).$$

Данное неравенство означает, что в некоторой подобласти $\mathcal{U}$ рост используемых ресурсов ведет к снижению среднего времени выполнения заданий, т.е. наращивание ресурсов вычислительного узла имеет смысл.

3. *Неубывание $\mathcal{M}_z(\cdot)$ по y* : для любого фиксированного z компоненты вектора y могут быть определены так, что для любых $(x^*, y'), (x^*, y'') \in \mathcal{A}$ таких, что $y' \leq y''$ покомпонентно, верно неравенство

$$\mathcal{M}_z(x^*, y') \leq \mathcal{M}_z(x^*, y'').$$

Согласно данному свойству вектор y может быть определен таким образом, что все его компоненты будут нести смысл объема задания, и с ростом их значений время выполнения задания в среднем будет увеличиваться.

4. *Непрерывность по y* : для любых фиксированных x и z функции $\mathcal{M}_z(x, y)$ и $\mathcal{D}_z(x, y)$ непрерывны по переменной y . Данное свойство означает, что при фиксированных ресурсах малые вариации объема пользовательского задания приводят к малым вариациям характеристик времени его выполнения.

5. *Выпуклость $\mathcal{M}_z(\cdot)$ по y* : для любого фиксированного z и для любых $(x^*, y'), (x^*, y'') \in \mathcal{U}$ и $\lambda \in [0, 1]$ справедливо неравенство

$$\mathcal{M}_z(x^*, \lambda y' + (1 - \lambda)y'') \leq \lambda \mathcal{M}_z(x^*, y') + (1 - \lambda) \mathcal{M}_z(x^*, y'').$$

Согласно неравенству при фиксированных ресурсах среднее время выполнения задания как функция объема пользовательского задания растет как линейная, или быстрее.

Замечание 3. Локальность невозрастания среднего времени выполнения задач по ресурсным переменным (свойство 2) выглядит экзотично лишь на первый взгляд. Во-первых, подобный характер поведения среднего времени выполнения наблюдается при анализе функционирования реальных приложений, например баз данных. Конкретный пример данного явления будет приведен во второй части представляемого исследования. Во-вторых, неумение времени выполнения задания с ростом объема предоставленных ресурсов может быть отчасти объяснено косвенными ограничениями и различиями в интенсивности использования ресурсов различного вида. Если представить работу всего приложения как совокупности некоторых конвейеров обработки данных разной интенсивности, эксплуатирующих единые ресурсы узла, то вполне возможна следующая ситуация. Среди упомянутых конвейеров существует один наименее производительный — “бутылочное горло”, определяющий в итоге интенсивность работы всего узла. Наращивание некоторых ресурсов выше определенного порога может вести к ситуации, когда “бутылочное горло” перестанет справляться с обработкой возросшего потока данных, и эти данные будут помещаться в очередь или попросту бесполезно теряться, требуя повторного их создания. Более того, из-за совместного использования конвейерами ресурсов часть их будет отнята у “бутылочного горла” более быстрыми конвейерами, дополнительно снижая его производительность.

Замечание 4. Для иллюстрации свойства 3 вновь рассмотрим пользовательское задание, заключающееся в численном решении некоторого уравнения математической физики. Упрощенно говоря, время выполнения задания является возрастающей функцией от числа узлов сетки, в которых требуется вычислить решение уравнения. Параметры задачи можно описать следующим образом: y_1 — шаг сетки по пространственной переменной, y_2 — шаг сетки по времени, y_3 — область интегрирования пространственной переменной, y_4 — область интегрирования по времени.

Однако такая параметризация не будет удовлетворять свойству 3: среднее время выполнения задания не будет расти с ростом шагов сетки (параметров y_1 и y_2). Параметризацию можно модифицировать подходящим образом так, чтобы свойство 3 выполнялось: корректный вариант приведен в данном разделе выше, сразу после Замечания 1.

Перечисленные свойства функций $\mathcal{M}_z(x, y)$ и $\mathcal{D}_z(x, y)$ накладывают на них определенные ограничения. В работе предлагается использовать следующие зависимости.

Первая модель предполагает экспоненциальную зависимость $\mathcal{M}_z$ и $\mathcal{D}_z$ от пары (x, y) “ресурсы — объем задания” и независимость от параметров z аппаратно-программной среды:

$$f_\varrho^1(x, y) = \alpha + \exp \left[\beta + \sum_{n=1}^N \gamma_n x_n + \sum_{m=1}^M \varepsilon_m y_m \right], \quad (3.1)$$

где $\varrho \triangleq \text{vec}(\alpha, \beta, \{\gamma_n\}_1^N, \{\varepsilon_m\}_1^M)$ — вектор неизвестных параметров, подлежащих последующей идентификации отдельно для функций $\mathcal{M}_z$ и $\mathcal{D}_z$. Свойства 1–5 функций $\mathcal{M}_z$ и $\mathcal{D}_z$ диктуют наличие следующих ограничений на векторы параметров:

- 1) $\min_{\{(x_n, y_m)\}} \left(\sum_{n=1}^N \gamma_n x_n + \sum_{m=1}^M \varepsilon_m y_m \right) + \beta - \ln \alpha \geq 0$ для функций $\mathcal{M}_z$ и $\mathcal{D}_z$ — условие неотрицательности,
- 2) $\frac{\gamma_n}{1, N} \leq 0$, $\overline{1, N}$ для функции $\mathcal{M}_z$ — условие локального невозрастания по переменным x_n , $\overline{1, N}$,
- 3) $\frac{\varepsilon_m}{1, M} \geq 0$, $\overline{1, M}$ для функции $\mathcal{M}_z$ — условие локального неубывания по переменным y_m , $\overline{1, M}$.

Экспоненциальная модель может быть использована для первоначального, так называемого разведочного статистического анализа [16], необходимого, например, для определения зоны эффективной работы виртуального узла.

Вторая модель предполагает степенную зависимость от компонентов (x, y) и независимость от параметров z аппаратно-программной среды. Таким образом, функции $\mathcal{M}_z$ и $\mathcal{D}_z$ второго типа имеют вид

$$f_{\varrho}^2(x, y) = \alpha + \beta \prod_{n=1}^N x_n^{-\gamma_n} \prod_{m=1}^M y_m^{\varepsilon_m}, \quad (3.2)$$

где $\varrho \triangleq \text{vec}(\alpha, \beta, \{\gamma_n\}_1^N, \{\varepsilon_m\}_1^M)$ — вектор неизвестных параметров, подлежащих последующей идентификации отдельно для функций $\mathcal{M}_z$ и $\mathcal{D}_z$. Свойства 1–5 функций $\mathcal{M}_z$ и $\mathcal{D}_z$ диктуют наличие следующих ограничений на векторы параметров, часть из которых можно задать явно, а часть зависит от каждой конкретной задачи идентификации:

- 4) $\alpha \geq 0, \beta \geq 0$ для функций $\mathcal{M}_z$ и $\mathcal{D}_z$ — условие неотрицательности,
- 5) $\gamma_n \geq 0, n = \overline{1, N}$ для функции $\mathcal{M}_z$ — условие невозрастания по переменной $x, n = \overline{1, N}$,
- 6) $\varepsilon_m \geq 0, m = \overline{1, M}$ для функции $\mathcal{M}_z$ — условие неубывания по переменной $y, m = \overline{1, M}$.

Третья модель предполагает кусочно-линейную зависимость по переменной y . При этом подмножества $\{U_j\}_{j=\overline{1, J}}$ множества $\mathcal{U}$, на которых функции $\mathcal{M}_z(x, y)$ и $\mathcal{D}_z(x, y)$ являются линейными по переменным y , образуют разбиение $\mathcal{U}$: $U_i \cap U_j = \emptyset$ для $\forall i \neq j$ и $\bigcup_{j=1}^J U_j = \mathcal{U}$. Введем в рассмотрение индикаторные функции $\mathbf{I}_{U_j}(x, y)$ множеств $\{U_j\}_{j=\overline{1, J}}$:

$$\mathbf{I}_{U_j}(x, y) \triangleq \begin{cases} 1, & \text{если } (x, y) \in U_j, \\ 0, & \text{если } (x, y) \notin U_j. \end{cases}$$

Для выполнения заданий различного объема виртуальный вычислительный узел использует различное количество ресурсов. При достижении заданиями определенных объемов узел скачкообразно меняет процедуру их обработки, при этом эти изменения недоступны прямому наблюдению. О них можно судить только косвенно, анализируя информацию, хранящуюся в системных журналах. Данный скачок и является причиной смены параметров линейной зависимости времени обработки от объема задания. Наиболее “выпукло” подобное явление можно наблюдать при варьировании объема заданий и скачкообразной смене использования оперативной памяти. При малых объемах все задание умещается в кэше — наиболее быстродействующей области оперативной памяти. Затем с ростом объема задание начинает заполнять свободную часть оперативной памяти, и скорость обработки несколько снижается. Как только объема оперативной памяти не хватает для полного размещения задания, включается механизм подкачки страниц, что резко увеличивает время обработки.

Итак, функции $\mathcal{M}_z$ и $\mathcal{D}_z$ третьего типа имеют вид

$$f_{\varrho}^3(x, y) = \alpha + \prod_{n=1}^N x_n^{-\beta_n} \sum_{j=1}^J \mathbf{I}_{U_j}(x, y) \left(\sum_{m=1}^M \gamma_{jm} y_m + \varepsilon_j \right), \quad (3.3)$$

где $\varrho \triangleq \text{vec}(\alpha, \{\beta_n\}_1^N, \{\gamma_{jm}\}_{1,1}^{J,M}, \{\varepsilon_j\}_1^J)$ — вектор неизвестных параметров, подлежащих последующей идентификации отдельно для функций $\mathcal{M}_z$ и $\mathcal{D}_z$. Следует уточнить, что γ_{jm} — это коэффициенты линейной зависимости по переменной $y_m, m = \overline{1, M}$ в области $U_j, j = \overline{1, J}$.

Свойства 1–5 функций $\mathcal{M}_z$ и $\mathcal{D}_z$ определяют следующие ограничения на оцениваемые параметры:

- 7) $\alpha \geq 0$ для функций $\mathcal{M}_z$ и $\mathcal{D}_z$ — условие неотрицательности,
- 8) $\beta_n \geq 0, n = \overline{1, N}$, для функций $\mathcal{M}_z$ — локальное невозрастание по функции x ,
- 9) $\gamma_{jm}^M \geq 0, m = \overline{1, M}, j = \overline{1, J}$, — неубывание $\mathcal{M}_z$ по переменной y ,
- 10) совместные ограничения на $\{\gamma_{jm}\}_{1,1}^{J,M}$ и $\{\varepsilon_j\}_1^J$, обеспечивающие непрерывность функций $\mathcal{M}_z$ и $\mathcal{D}_z$ по переменной y ,
- 11) совместные ограничения на $\{\gamma_{jm}\}_{1,1}^{J,M}$ и $\{\varepsilon_j\}_1^J$, обеспечивающие выпуклость $\mathcal{M}_z$ по переменной y .

Замечание 5. Модели среднего времени выполнения заданий (3.2) и (3.3) допускают следующую экономическую интерпретацию. Предположим, что алгоритм, реализующий решение пользовательского задания, относится к классу сложности P . Это означает, что число операций, обеспечивающих выполнение задания, ограничено сверху некоторой степенной, в частном случае линейной функцией компонентов вектора Y . Из-за использования механизма подкачки страниц при недостатке оперативной памяти число операций может увеличиваться, что описывается с помощью степенной или выпуклой кусочно-линейной функции параметров задания. Интенсивность выполнения задания, определяющая число операций алгоритма, выполняемого на вычислительном узле в единицу времени, близка по смыслу к производственной функции [10], в которой в качестве ресурсов выступают компоненты вектора x . Одной из популярных является функция Кобба—Дугласа, представляющая собой произведение компонент x с некоторыми *положительными* показателями. Зная общее число операций и интенсивность их выполнения, время выполнения задания определяется их частным. Модели (3.2) и (3.3) как раз и представляют собой отношение оценок сверху числа потребных операций к интенсивности их выполнения, описываемой функцией Кобба—Дугласа.

Предложенные стохастические модели времени выполнения позволяют корректно сформулировать проблемы 1 и 2 из предыдущего раздела как задачи анализа и решить их.

Задача анализа 1. На виртуальном узле выполняются задания фиксированного типа, функции среднего $\mathcal{M}_z(x, y)$ и дисперсии $\mathcal{D}_z(x, y)$ времени $\tau(\omega)$ выполнения задания известны. Помимо этого тройка “конфигурация узла — параметры задания — параметры аппаратно-программной среды” (x, y, z) фиксированы и известны. Пусть $\bar{T}$ — параметр SLA, определяющий максимально допустимое время выполнения пользовательского задания Y : оценить сверху вероятность превышения временем $\tau(\omega)$ порога $\bar{T}$.

Решение задачи может быть получено с помощью неравенства Чебышева [15]:

$$\begin{aligned} P\{\tau(\omega) > \bar{T}\} &= P\{\tau(\omega) - \mathcal{M}_z(x, y) > \bar{T} - \mathcal{M}_z(x, y)\} \leq \\ &\leq \frac{\mathcal{D}_z(x, y)}{\mathcal{D}_z(x, y) + (\bar{T} - \mathcal{M}_z(x, y))^2}. \end{aligned} \quad (3.4)$$

Задача анализа 2. Параметры конфигурации узла x фиксированы и известны, а параметры пользовательского задания $Y(\omega)$ являются случайными. Тем не менее пользовательский профиль задания известен и задан в форме вероятностного распределения $P(dy)$ вектора $Y(\omega)$. Помимо этого известна зависимость, связывающая значение параметра z от пары (x, y) : $z = z(x, y)$. Пусть $\bar{T}$ — параметр SLA, определяющий максимальное время выполнения пользовательского задания с параметрами Y : оценить сверху вероятность превышения временем $\tau(\omega)$ порога $\bar{T}$.

Среднее время выполнения пользовательского задания $\bar{\mathcal{M}}$ вычисляется с помощью формулы полной вероятности [15]:

$$\bar{\mathcal{M}} = E\{\tau(\omega)\} = E\{E\{\tau(\omega) | Y\}\} = E\{\mathcal{M}_{z(x, Y)}(x, Y)\} = \int \mathcal{M}_{z(x, y)}(x, y) P(dy). \quad (3.5)$$

Аналогичным образом можно рассчитать и дисперсию $\bar{D}$:

$$\begin{aligned}
 \bar{D} &= \mathbb{E}\left\{\left(\tau(\omega) - \bar{M}\right)^2\right\} = \\
 &= \mathbb{E}\left\{\mathbb{E}\left\{\left(\left(\tau(\omega) - \mathcal{M}_z(x, Y)\right) + \left(\mathcal{M}_z(x, Y) - \bar{M}\right)^2 \mid Y\right)\right\}\right\} = \\
 &= \mathbb{E}\left\{\mathbb{E}\left\{\left(\tau(\omega) - \mathcal{M}_z(x, Y)\right)^2 \mid Y\right\}\right\} + \mathbb{E}\left\{\mathbb{E}\left\{\left(\mathcal{M}_z(x, Y) - \bar{M}\right)^2 \mid Y\right\}\right\} + \\
 &\quad + 2\mathbb{E}\left\{\mathbb{E}\left\{\left(\tau(\omega) - \mathcal{M}_z(x, Y)\right)\left(\mathcal{M}_z(x, Y) - \bar{M}\right) \mid Y\right\}\right\} = \\
 &= \int \mathcal{D}_z(x, y) P(dy) + \int \left(\mathcal{M}_z(x, y) - \bar{M}\right)^2 P(dy) = \\
 &= \int \mathcal{D}_z(x, y) P(dy) + \int \mathcal{M}_z^2(x, y) P(dy) - (\bar{M})^2.
 \end{aligned} \tag{3.6}$$

Решение задачи вновь может быть получено с помощью неравенства Чебышева:

$$\mathbb{P}\{\tau(\omega) > \bar{T}\} \leq \frac{\bar{D}}{\bar{D} + (\bar{T} - \bar{M})^2}. \tag{3.7}$$

Таким образом, предложенная стохастическая модель времени выполнения пользовательских заданий позволяет эффективно решать важную задачу анализа — оценивать сверху вероятность нарушения SLA.

Оценки (3.4) и (3.7), основанные на неравенствах Чебышева, являются очень консервативными. Если о распределении времени $\tau(\omega)$ имеется дополнительная информация, то предложенные оценки могут быть существенно уточнены (см. работы [17, 18] и ссылки внутри них). Отметим, что данные предположения не являются чрезмерно обременительными и почти всегда выполняются на практике.

Первое предположение заключается в том, что функция распределения $\tau(\omega)$ — вогнутая на положительной полуоси. В этом случае решение задачи анализа 1 может быть получено из неравенства Гаусса:

$$\mathbb{P}\{\tau(\omega) > \bar{T}\} \leq \begin{cases} 1 - \frac{\bar{T}}{\sqrt{3(\mathcal{M}_z^2(x, y) + \mathcal{D}_z(x, y))}}, & \text{если } 0 \leq \bar{T} \leq \frac{2}{\sqrt{3}} \sqrt{\mathcal{M}_z^2(x, y) + \mathcal{D}_z(x, y)}, \\ \frac{4(\mathcal{M}_z^2(x, y) + \mathcal{D}_z(x, y))}{9\bar{T}^2}, & \text{если } \frac{2\sqrt{\mathcal{M}_z^2(x, y) + \mathcal{D}_z(x, y)}}{\sqrt{3}} < \bar{T}. \end{cases} \tag{3.8}$$

Второе предположение состоит в том, что у $\tau(\omega)$ имеется плотность распределения, являющаяся унимодальной функцией на положительной полуоси. В этом случае решение задачи анализа 1 может быть получено из неравенства Кантелли:

$$\begin{aligned}
 &\mathbb{P}\{\tau(\omega) > \bar{T}\} \leq \\
 &\leq \begin{cases} \frac{3\mathcal{D}_z(x, y) - (\bar{T} - \mathcal{M}_z(x, y))^2}{3\mathcal{D}_z(x, y) + 3(\bar{T} - \mathcal{M}_z(x, y))^2}, & \text{если } \mathcal{M}_z(x, y) \leq \bar{T} \leq \mathcal{M}_z(x, y) + \sqrt{\frac{5}{3}\mathcal{D}_z(x, y)}, \\ \frac{4}{9} \frac{\mathcal{D}_z(x, y)}{\mathcal{D}_z(x, y) + (\bar{T} - \mathcal{M}_z(x, y))^2}, & \text{если } \mathcal{M}_z(x, y) + \sqrt{\frac{5}{3}\mathcal{D}_z(x, y)} < \bar{T}. \end{cases} \tag{3.9}
 \end{aligned}$$

Замечание 6. Вид функций $f_\varrho(\cdot)$, используемых для описания моментных характеристик времени $\tau(\omega)$ выполнения задания, не исчерпывается функциями (3.1) — (3.3). Причины выбора этих моделей в представляемом исследовании следующие. Экспоненциальная зависимость (3.1) может быть использована для разведочного анализа поведения описываемых моментных характеристик и определения области (x, y) , в которой узел эффективно выполняет задания указанного типа. Степенная модель (3.2) имеет удачную экономическую интерпретацию. Модель (3.3), представляющая собой степенную зависимость от ресурсных параметров и кусочно-линейную — от объема выполняемого задания, также имеет свои сильные стороны. Во-первых, она проста и экономична, так как включает небольшое число параметров. Во-вторых, эта модель позволяет описывать скачкообразную смену дисциплины обработки заданий в зависимости от их объема.

4. Задачи идентификации параметров. Пусть дан некоторый фиксированный набор $\{(x^r, y^r)\}_{r=\overline{1, R}}$ переменных (x, y) , для которых проведен комплекс нагрузочного тестирования исследуемого вычислительного узла. Результатом тестирования является набор векторов $\{\tilde{M}^r, \tilde{D}^r, \tilde{Z}^r\}_{r=\overline{1, R}}$, полученных при обработке статистической информации о работе узла при фиксированных значениях пар (x^r, y^r) . Первая компонента $\tilde{M}^r$ представляет собой выборочное среднее времени $\tau(\omega)$, вторая — его дисперсию, а третья компонента, возможно блочная — некоторые известные выборочные характеристики аппаратно-программной среды, соответствующие паре (x^r, y^r) . Также задан набор положительных весов $\{w_r\}_{r=\overline{1, R}}$, определяющих индивидуальную значимость результат каждого эксперимента.

Выбор моделей, описывающих математическое ожидание и дисперсию времени выполнения задания среди возможных функций (3.1)–(3.3), а также идентификация их параметров, проводятся независимо. Поэтому подробно опишем постановку задачи идентификации только для функции математического ожидания: задача идентификации параметров функции дисперсии будет выглядеть абсолютно аналогично. Величина

$$\Delta_r^{M,i}(\varrho) \triangleq \tilde{M}_r - f_\varrho^i(X_r, Y_r), i = 1, 2, 3,$$

представляет собой ошибку оценки среднего значения времени выполнения задания при описании конфигурации (x^r, y^r) функцией $f_\varrho^i(\cdot)$, вычисленной со значениями параметров ϱ .

Для сравнения качества моделей с различными параметрами воспользуемся функцией потерь $\pi(u) : \mathbb{R}^2 \rightarrow \mathbb{R}$, удовлетворяющей следующим свойствам [11]: $\pi(0) = 0$, $\pi(u) \geq 0$ для всех $u \in \mathbb{R}^2$, функция $\pi(u)$ является невозрастающей при $u < 0$ и неубывающей при $u > 0$.

Задача оптимальной идентификации модели среднего времени выполнения задания заключается в нахождении

$$(i^*, \varrho^*) \in \operatorname{Argmin}_{i, \varrho} \sum_{r=1}^R w_r \pi(\Delta_r^{M,i}(\varrho)). \quad (4.1)$$

Задача оптимальной идентификации модели дисперсии времени выполнения задания состоит в определении

$$(i^*, \varrho^*) \in \operatorname{Argmin}_{i, \varrho} \sum_{r=1}^R w_r \pi(\Delta_r^{D,i}(\varrho)). \quad (4.2)$$

Если в качестве функции потерь выступает квадратичная функция, т.е. $\pi(u) = u^2$, то (i^*, ϱ^*) называется оценкой наименьших квадратов, если используется абсолютное значение, т.е. $\pi(u) = |u|$, то (i^*, ϱ^*) — оценка наименьших модулей [19, 20]. В более общем случае, когда $\pi(u)$ является кусочно-линейной функцией, удовлетворяющей представленным выше свойствам, (i^*, ϱ^*) называется квантильной оценкой [21]. Очевидно, что (3.10) и (3.11) представляют собой взвешенный вариант задач построения М-оценок [19].

Замечание 7. Использование различных весов $\{w_r\}_{r=\overline{1,R}}$ имеет практический смысл. Во-первых, точность индивидуальных значений выборочных средних и дисперсий $(\tilde{M}^r, \tilde{D}^r)$ для разных параметров (x^r, y^r) может быть разной, и это необходимо учитывать при последующей идентификации. Такая ситуация может складываться в том случае, когда при некотором аппаратно-программном составе узла x^r время выполнения заданий y^r достаточно велико и нет возможности найти тестовую выборку достаточного размера для последующего осреднения и получения $(\tilde{M}^r, \tilde{D}^r)$ приемлемой точности. Во-вторых, на практике наблюдается следующее явление: в случае “скудных” ресурсов x наблюдается большой разброс времени $\tau(\omega)$, что ведет к низкой точности выборочных моментов $\tilde{M}^r$ и $\tilde{D}^r$. Так как модели $\mathcal{M}(\cdot)$ и $\mathcal{D}(\cdot)$ строятся едиными как для всех значений x , то неравноточность наблюдений $(\tilde{M}_r, \tilde{D}_r)$ можно учитывать путем выбора соответствующих весов w_r . В-третьих, владелец вычислительного узла может иметь некоторую дополнительную априорную информацию о предпочтении пользователя выполнять задания с определенными параметрами чаще остальных. В связи с этим может возникнуть задача идентифицировать параметры таким образом, чтобы модель точнее описывала время выполнения именно предпочитаемых пользователем заданий, возможно, в ущерб описанию остальных.

5. Статистическая информация для идентификации моделей: принципы сбора и обработки. Задача идентификации параметров модели трудоемкости заданий при их выполнении на некотором виртуальном узле является абсолютно практической, и поэтому сбор статистических данных для ее решения должен проводиться на той же аппаратно-программной платформе, которая и будет затем использоваться в реальности. Затруднение может возникнуть только при оснащении узла планируемым СПО: по разным причинам оно может быть недоступно до момента реальной эксплуатации. В этом случае данное СПО следует заменить ТПО, близким по целевой направленности к планируемому СПО.

В настоящее время существует достаточно большой выбор доступного ТПО, отличающегося друг от друга:

- предметом тестирования (общая производительность компьютера, производительность процессора, графических сопроцессоров, дисковой системы и пр.),
- набором симулируемых приложений (офисные программы, базы данных, научные вычисления и пр.),
- уровнем кроссплатформенности и готовности к нагрузочному тестированию (готовность “из коробки” / необходимость предварительной компиляции кода для тестируемой аппаратно-программной платформы),
- уровнем сервиса сбора статистической информации в процессе нагрузочного тестирования (есть/нет встроенная возможность журналирования используемых ресурсов),
- видом лицензии (платная/бесплатная).

В исследуемой задаче идентификации первые два признака являются ключевыми при выборе того или иного ТПО. Следующие виды ТПО позволяют имитировать нагрузку, порождаемую:

- офисными приложениями [22, 23],
- компиляторами, интерпретаторами [24],
- сжатием данных, в том числе видео [24],
- научными вычислениями (гидродинамика, моделирование атмосферы, молекулярная динамика и пр.) [24],
- 3D-рендерингом [24],
- транзакциями баз данных [25].

Использование ТПО вместо СПО допустимо на начальном этапе идентификации, однако окончательное уточнение параметров модели возможно только на основе статистической информации, полученной при эксплуатации реального СПО. Далее, вне зависимости от применения СПО/ТПО следует учитывать, что для решения задач оптимальной идентификации среднего времени (3.10) и дисперсии (3.11) выполнения задания необходимо многократное повторение тестов с одними и теми же значениями пар (x^r, y^r) с последующим вычислением по ним соответствующих выборочных моментов $(\tilde{M}^r, \tilde{D}^r)$.

Помимо измерения времени τ во время нагрузочного тестирования следует с некоторой периодичностью регистрировать данные, относящиеся к состоянию аппаратно-программной

среды, т.е. ту информацию, обработка которой позволит определить компоненты вектора $\mathcal{Z}$. Обычно фиксируется следующая информация:

- процент использования процессорного времени,
- объем оперативной памяти,
- объем записанных данных,
- объем считанных данных,
- объем страничной памяти,
- объем данных, записанных во вторичное хранилище страничной памяти (своп),
- объем данных, считанных из свопа и пр.

Обычно периодичность фиксации данных составляет 1 с. Для дальнейшей обработки полученных “сырых” данных к ним обычно применяются стандартные операции осреднения, нахождения минимальных/максимальных/медианных значений и пр. Дополнительная информация в форме вектора $\mathcal{Z}$ позволяет построить разбиение $\mathcal{U}$ для последующего применения кусочно-линейной модели (3.3). Например, с помощью подобной обработки данных можно определить минимальный объем задания, с которого начинает использоваться механизм страничной подкачки памяти, кардинально замедляющий выполнение заданий.

В заключение следует отметить, что практическая реализация идентификации параметров описанных моделей связана с решением следующих проблем. Первая из них имеет более теоретический характер и связана с корректным планированием эксперимента по нагрузочному тестированию [26, 27], обеспечивающим идентифицируемость параметров [28], построением их состоятельных оценок, а также характеристикой их скорости сходимости к истинным значениям [29]. Вторая проблема является по большей части алгоритмической. Дело в том, что задачи идентификации (3.10) и (3.11) относятся к оптимизационным, критерий в которых может быть не только невыпуклым по оптимизируемым переменным, но и негладким. Набор строгих утверждений и теоретически обоснованных алгоритмов решения подобных оптимизационных проблем достаточно скуден, что ведет к необходимости использования эвристических и нейросетевых алгоритмов. Проблема заключается в рациональном выборе алгоритма оптимизации в соответствии со спецификой критерия и вида ограничений или разработке собственной версии алгоритма. Достаточно полный обзор и описание подобных алгоритмов оптимизации приведен в монографии [30].

Заключение. Первая часть исследования посвящена теоретическому аспекту построения математической модели времени выполнения пользовательских заданий на виртуальном вычислительном узле. Приведена аргументация в пользу использования вероятностного аппарата для описания фундаментального свойства вариативности, присущего этому времени. Исходя из необходимости последующей идентификации параметров модели вероятностного распределения времени, предложено ограничиться двумя его первыми моментами. Аргументы функций, описывающих выбранные моменты, включают в себя характеристики ресурсов, пользовательских заданий и аппаратно-программной среды. В работе представлены свойства этих функций, а также некоторые их варианты. Очевидно, что функции содержат неизвестные параметры, подлежащие идентификации для каждого вычислительного узла и вида заданий по статистическим данным, полученным в результате нагрузочных испытаний. Среди различных видов оценок параметров предлагается использовать взвешенные М-оценки. Представлены принципы сбора и обработки статистической информации по результатам нагрузочного тестирования, проводимого в целях идентификации параметров предложенной модели.

Исследования в данной области нельзя считать законченными. Перспективными предполагаются следующие направления. Во-первых, параметрическая идентификация модели является достаточно ресурсозатратной задачей, требующей продолжительного и разнообразного нагрузочного тестирования. В связи с этим вопросы оптимизации плана тестирования и адаптивного уточнения параметров модели в процессе эксплуатации узла представляются весьма актуальными. К этой же области следует отнести возможное расширение и уточнение класса функций, описывающих моментные характеристики времени выполнения заданий. Во-вторых, важной является детализация модели, позволяющая учитывать некоторые дополнительные особенности функционирования виртуальных узлов: совместное использование аппаратных ресурсов различными виртуальными узлами, одновременное обращение различных пользователей к одному виртуальному узлу и пр.

Однако первостепенное значение имеет проверка возможности применения предложенной методологии для построения моделей трудоемкости вычислительных заданий различного типа: реализации обработки информации в базах данных, научных вычислений, обработки изображений, сжатия данных и пр. Именно эту цель преследуют последующие части данного исследования.

СПИСОК ЛИТЕРАТУРЫ

1. *Sirohi P., Agarwal A., Maheshwari P.* A Survey on Augmented Virtual Reality: Applications and Future Directions // Seventh Intern. Conf. on Information Technology Trends (ITT). Abu Dhabi, United Arab Emirates, 2020, P. 99–106.
2. *Gupta P., Krishna C., Rajesh R. et al.* Industrial Internet of Things in Intelligent Manufacturing: a Review, Approaches, Opportunities, Open Challenges, and Future Directions // Int. J. Interact. Des. Manuf. 2022.
3. *Parveen I., C.A., Anjali, O., Sunder, R.* Internet of Things: A Review on Its Applications // Information and Communication Technology for Competitive Strategies (ICTCS 2021). Lecture Notes in Networks and Systems. 2021. V. 400. P. 123–134.
4. *Fraga-Lamas P., Fernández-Caramés T., Suárez-Albela M., Castedo L., González-López M.* A Review on Internet of Things for Defense and Public Safety // Sensors. 2016. V. 16. Iss. 10. P. 1644.
5. *Abu Talib M., Abbas S., Nasir Q., Mowakeh M.* Systematic literature review on Internet-of-Vehicles Communication Security // Intern. J. of Distributed Sensor Networks. 2018. V. 14. P. 12.
6. *Nocedal J., Wright S.* Numerical Optimization. N.Y.: Springer, 2006. 686 p.
7. *Bertsekas D.* Convex Optimization Algorithms. Belmont: Athena Scientific, 2015. 576 p.
8. *Kearns M.* The Computational Complexity of Machine Learning. Cambridge: MIT Press, 1990. 182 p.
9. *Teller S., Séquin C.* Visibility preprocessing for interactive walkthroughs // Proc. 18th Annual Conf. on Computer Graphics and Interactive Techniques. 1991. P. 61–70.
10. *Интрилигатор М.* Математические методы оптимизации и экономическая теория. М.: Прогресс, 1975. 600 с.
11. *Хьюбер П.* Робастность в статистике. М.: Мир, 1984. 304 с.
12. *Броневич А., Ленский А.* Нечеткие модели анализа данных и принятия решений. М.: ВШЭ, 2022. 264 с.
13. *Pankov A., Siemenikhin K.* Minimax Estimation for Singular Linear Multivariate Models with Mixed Uncertainty // J Multivariate Analysis. 2007. V. 98. Iss. 1. P. 145–176.
14. *Borisov A.* Minimax Estimation in Regression under Sample Conformity Constraints // Mathematics. 2021. V. 9. P. 1080.
15. *Ширяев А.* Вероятность. М.: Физматлит, 1989. 644 с.
16. *Tukey J.* Exploratory Data Analysis. Boston: Addison-Wesley, 1977. 712 p.
17. *Семенухин К.* Двусторонняя вероятностная граница для симметричной унимодальной случайной величины // АиТ. 2019. № 3. С. 103–122.
18. *Ion R., Klaassen C., van der Heuvel E.* Sharp Inequalities of Bienaymé-Chebyshev and Gauß Type for Possibly Asymmetric Intervals Around the Mean // TEST. 2023. <https://doi.org/10.1007/s11749-022-00844-9>
19. *Демиденко Е.* Линейная и нелинейная регрессии. М.: Финансы и статистика, 1981. 304 с.
20. *Мудров В., Кушко В.* Метод наименьших модулей. М.: URSS, 2022. 64 с.
21. *Koenker R.* Quantile Regression. Cambridge: Cambridge University Press, 2005. 368 p.
22. <https://bapco.com/products/sysmark-25/>
23. <https://benchmarks.ul.com/pcmark10>
24. <https://www.spec.org/cpu2017/>
25. <https://www.tpc.org/tpce/>
26. *Ермаков С., Козлов В., Жигляевский А.* Математическая теория планирования эксперимента. М.: Наука, 1983. 392 с.
27. *Black R.* Managing the Testing Process, 3rd Edition: Practical Tools and Techniques for Managing Hardware and Software Testing. Indianapolis: Wiley, 2009. 672 p.
28. *Льюнг Л.* Идентификация систем: Теория для пользователя. М.: Наука, 1991. 432 с.
29. *Ивченко Г., Медведев Ю.* Математическая статистика. М.: URSS, 2014. 608 с.
30. *Пантелеев А., Метлицкая Д., Алешина Е.* Методы глобальной оптимизации. Метаэвристические стратегии и алгоритмы. М.: Вузовская книга, 2013. 244 с.