Федеральное государственное учреждение «Федеральный исследовательский центр «Информатика и управление» Российской академии наук»

На правах рукописи

Разумчик Ростислав Валерьевич

МЕТОДЫ АНАЛИЗА И АЛГОРИТМЫ УПРАВЛЕНИЯ ЧАСТИЧНО НАБЛЮДАЕМЫМИ СТОХАСТИЧЕСКИМИ СИСТЕМАМИ ОБСЛУЖИВАНИЯ

Специальность 2.3.1 —
«Системный анализ, управление и обработка информации, статистика»

Диссертация на соискание учёной степени
доктора физико–математических наук

Москва — 2022

Оглавление

Стр.

Стр.

Введение

Для современных суперкомпьютерных систем, систем распределенных вычислений, сетевых и производственных систем типичной является ситуация, когда взаимодействие или работу с ними необходимо огранизовывать в условиях неполного наблюдения (или, что то же, — частичного наблюдения, неполного информационного описания и т. п.). Как отмечено, например, в [1], неполнота эта может проявляться по-разному. Это и (частичное или полное) отсутствие априорной информации о системе, и ограниченная возможность наблюдения состояний системы. В подобных ситуациях для анализа и оптимизации системы первостепенное значение приобретает умение воспользоваться теми сведениями о ней, которые имеются в распоряжении.

Если от системы в процессе функционирования поступает какая–либо дополнительная информация, то для достижения цели обычно используются методы теории адаптации. Судя по публикациям в открытой печати (см. [1, Введение]), ее основополагающие идеи были заложены в середине прошлого века. Становление же теории и ее развитие до конца 80-х годов проходило во многом благодаря усилиям отечественных ученых [2;3]. С начала 90-х годов и по настоящее время адаптивное направление переживает большой подъем, что косвенно подтверждается неутихающим год от года потоком публикаций. Без сомнения, такой углубленный интерес вызван как новыми потребностями практики, так и прогрессом в области информационных технологий, который позволил поставить на реальную почву практическую реализацию адаптивных алгоритмов (см., например, [4–10]).

Если же дополнительная информация в ходе взаимодействия с системой не приобретается, то это делает фактически невозможным приспособление или, другими словами, применение адаптивных стратегий. Развиваемое в диссертационной работе направление связано с проблемами именно такого типа, т. е. лежит в русле фундаментальных исследований не адаптивного характера[1] в области стохастическим систем (см. [11]) с частичной наблюдаемостью. Сейчас эта проблематика является предметом постоянного внимания в научном сооб-

[1]Однако, в тех случаях, когда в диссертационной работе для получения решений приходится привлекать имитационные модели, некоторые приемы адаптивного управления все-таки используются.

ществе как в России, так и за рубежом[2] (см., например, [12–22] и ссылки в них). Ярким подтверждением этому является то обстоятельство, что в нее начали проникать идеи (см., например, [23;24]), тесно связанные с машинным обучением — сегодня одной из наиболее активно развивающихся научных областей [25–28]. В целом, круг нерешенных и не вполне решенных здесь проблем остается широким. Связано это, во-первых, с большим, диктуемым практикой разнообразием постановок. Во-вторых, зачастую к решениям не удается прийти исключительно математическими методами. Поэтому для повышения эффективности, надежности и качества систем приходится обращаться к методам статистического моделирования, искать эвристические идеи и разрабатывать инженерные подходы. Таким образом, тематика диссертационной работы находится в одной из **актуальных** областей современной науки, в которой необходим дальнейший прогресс.

Целью диссертационной работы является решение фундаментальной научной проблемы — разработка комплекса вероятностных моделей и создание на их основе методов анализа и алгоритмов управления для стохастических систем обслуживания с частичной наблюдаемостью.

Для достижения поставленной цели в диссертации решаются следующие **задачи**:

- разработка комплекса вероятностных моделей для анализа стационарных вероятностно-временных характеристик стохастических систем обслуживания, в которых не наблюдаются необходимые для управления очередями фактические времена обслуживания[3];
- разработка метода оценки значений стационарных вероятностно-временных характеристик частично наблюдаемых стохастических систем

[2]Из зарубежных научных и научно–практических центров можно отметить: исследовательский центр IBM T.J. Watson Research Center (США), национальный государственный исследовательский институт по информатике и автоматике INRIA (Франция), европейский институт исследования операций EURANDOM (Нидерланды), гренобльская лаборатория компьютерных наук LIG (Франция), департамент систем телекоммуникаций университета Аалто (Финляндия), департамент естественных наук и технологий университета Карнеги Меллон (США), центр математических разработок для ключевых технологий немецких университетов Matheon (Германия).

[3]То обстоятельство, что вместо точных значений времен обслуживания при планировании очередей могут быть доступны лишь некоторые оценки этих величин, хорошо известно как в практике эксплуатации современных информационных, вычислительных и телекоммуникационных систем, так и в научной литературе; например, суперкомпьютерные системы [29; 30], веб–серверы [31–34], пиринговые сети [35], MapReduce системы [36–39], базы данных [40].

обслуживания на основе доступной информации о прогнозных временах обслуживания и исследование границ его применимости ;

– разработка алгоритмов централизованного[4] квазиоптимального управления входящими потоками (диспетчеризации) в стохастических системах с параллельным обслуживанием при полной недоступности динамической информации об их состоянии[5];
– создание для частично наблюдаемых стохастических систем с параллельным обслуживанием простых и эффективных алгоритмов централизованной диспетчеризации, позволяющих решать задачи большой размерности.

Решаемые задачи **поставлены** в терминах[6] теории массового обслуживания (ТМО). Эта область математики, даже спустя 100 лет с момента зарождения, продолжает развиваться[7], и выделяется как разнообразием постановок задач, так и обилием применяемых математических методов исследования. На

[4]Т. е. решающая функция закреплена за одним узлом — т. н. диспетчером.

[5]Отметим, что это ограничение характерно для некоторых реально функционирующих систем и, в частности, систем добровольных вычислений (volunteer computing) [41, Section 2.3]. Типичная система представляют собой совокупность параллельно и независимо друг от друга работающих обслуживающих ресурсов, которые выполняют задания, направляемые на них диспетчером. При этом диспетчер, осуществляя выбор ресурса для выполнения очередного задания, не имеет возможности отложить решение. Ему также недоступна информация о состоянии ресурсов. Вопросы применения таких систем на практике и примеры новейших экспериментальных исследований обсуждаются, например, в работах [42–49].

[6]При этом, однако, в первой части удобно было придерживаться терминов “заявка”, “прибор”, “система”, а во второй части — терминов “задание”, “процессор”, “сервер” (характерных скорее для вычислительных систем [50]).

[7]ТМО была развита в фундаментальных работах Ф. Поллячека, К. Пальма, Д. Кендалла, Д. Линдли, П. Морана, Л. Такача, Дж. Ф.С. Кингмана, Д. Кокса, Т.Л. Саати, Л. Клейнрока, В.Е. Бенеша, Н.К. Джейсуола, С. Карлина, С. Асмуссена, М. Ньютса и др. за рубежом и А.Я. Хинчина, Б.В. Гнеденко, Б.А. Севастьянова, Ю.В. Прохорова, А.А. Боровкова, Г.П. Башарина, Г.П. Климова, А.Д. Соловьева, В.В. Калашникова, И.Н. Коваленко и многих других в нашей стране. Нет никакой возможности здесь хоть сколько–нибудь полноценно охватить современную литературу в области ТМО. Если сделанный в 1970 году достаточно полный обзор [51] содержит всего порядка тысячи наименований, то список литературы, например, диссертационного исследования [52] 2016 года, посвященного одной открытой проблеме в области ТМО (выработке нового неклассического подхода к моделированию конфликтных управляющих систем массового обслуживания (см. также [53–56])), содержит уже более 200 работ. Поэтому ограничимся ссылкой на спецвыкуск 1–2 тома 89 и том 100 журнала Queueing Systems [57; 58], которые могут дать некоторое представление о текущем состоянии исследований в области ТМО.

этот фундамент и опираются полученные в диссертации аналитические результаты[8].

Перейдем к обзору содержания диссертации. Она состоит из двух больших частей. К частично наблюдаемым стохастическим системам — системам массового обслуживания (СМО), — являющимся объектом внимания в первой части диссертации (главы 1 и 2), относится любая система, для которой выполнены, главным образом, два условия (подробнее см. стр. 106). Во-первых, для каждой поступающей заявки становится известным некоторое положительное число; оно считается ее остаточным прогнозным временем обслуживания, и имеет смысл работы, которую, как ожидается, необходимо совершить прибору для завершения обработки заявки. Во-вторых, та работа, которую в действительности необходимо совершить прибору для завершения ее обработки (т. е. фактическое время обслуживания заявки), хотя фиксируется в момент поступления заявки в систему, однако ненаблюдаема и не совпадает с указанным для заявки прогнозным временем обслуживания. Из этого следует, что, говоря о вероятностно–временных характеристиках частично наблюдаемых СМО, необходимо отличать их прогнозные значения, от фактических. Поскольку для задач практики значение имеют, вообще говоря, лишь последние, то возникает задача оценки[9] фактических значений только на основе доступной информа-

[8]Говоря более точно, аналитические результаты диссертации относятся к тому направлению (см. [59, Введение]) исследований в ТМО, которое связано с изучением “неклассических” постановок задач. Однако, если обычно исследования здесь мотивированы возможностью существенного улучшения качества работы системы с помощью применения специальных дисциплин, то в диссертации побудительным мотивом явилась обнаруженная экспериментально возможность уточнения с их помощью характеристик стохастических систем обслуживания с частичной наблюдаемостью.

[9]Или, строго говоря, уточнения тех оценок, которые всегда могут быть получены на основе имеющейся прогнозной информации без применения каких-либо специальных методов. Необходимо здесь добавить, что рассматриваемая задача примыкает к характеризационным задачам в теории массового обслуживания в том смысле, что речь здесь по сути идет о нахождении необходимых и/или достаточных условий для выполнения того или иного свойства (к примеру (2.5)). Однако к известным задачам характеризации свести ее не удается. Поясним это обстоятельство, воспользовавшись схемой характеризации, предложенной в [60] (это можно было также сделать, воспользовавшись и схемой устойчивости стохастических моделей В.М. Золотарева [61], но, с учетом специфики задачи, здесь удобнее схема [60]). Пусть стохастическая система трактуется (см. [60, с. 39]) как преобразование $\mathfrak{F}$ исходных данных $\mathbb{U} \in \mathfrak{U}$ в выходные данные $\mathbb{V} \in \mathfrak{V}$ т.е. $\mathfrak{F} : \mathfrak{U} \to \mathfrak{V}$. Вид преобрзования $\mathfrak{F}$ “диктуется” структурой системы. Назовем $\mathbb{Z} = (\mathbb{U},\mathbb{V})$ даными о системе, $\mathbb{Z} \in \mathfrak{Z} = \mathfrak{U} \times \mathfrak{V}$. Одним из главных предположений в (прямых и обратных) задачах характеризации является предположение о том, что в полном объеме данные $\mathbb{Z}$ неизвестны. Вместо этого известна некоторая априорная информация (т.е. что $\mathbb{Z}$ принадлежит некоторому фиксированному множеству $\mathfrak{Z}^* \subset \mathfrak{Z}$), и наблю-

ции о прогнозных временах обслуживания[10]. И в первой часте диссертации (см. главу 2) **впервые** предложен метод, позволяющий получать такие оценки для стационарного режима при определенных, продиктованных практикой ограничениях[11]. Выяснение условий, гарантирующих содержательность оценок, является одним из центральных результатов главы. Идея метода заключается в преобразовании остаточных прогнозных времен обслуживания заявок некоторым вероятностным механизмом, не сохраняющим работу, причем моменты преобразований синхронизированы с моментами поступления новых заявок в систему. Подмеченные в вычислительных экспериментах факты того, что, действуя подобным образом, можно получать содержательные результаты, а также отсутствие результатов в научной литературе, с помощью которых можно было бы объяснить наблюдаемые эффекты[12], послужили главным поводом для теоре-

дения — величины $\mathbb{W}(\mathbb{U},\mathbb{V})$, принимающие значения из известного подмножества $\mathfrak{W}^*$ пространства наблюдений $\mathfrak{W}$. Тогда, в зависимости от вида $\mathfrak{W}^*$ различают прямые и обратные задачи характеризации. В рассматриваемой задаче отсутствует необходимый компонент — наблюдения. Поэтому находжение точных распределений (задача, укладывающаяся в схему прямых задач характеризации) невозможно.

[10]И, разумеется, информации о структуре СМО, временах обслуживания заявок на приборах и дисциплины обслуживания. Важно отметить, что решение сформулированной задачи оценки фактических значений вероятностно–временных характеристик СМО предназначено для использования в задачах практики определенного рода. Воспользовавшись терминологией системного анализа систем связи, таковыми задачами являются (см., например, [62; 63]): определение необходимости разработки новых систем; выбор из нескольких систем, могущих решать одинаковые задачи, лучшей; выработка наилучших способов эксплуатации системы. Из сказанного следует, что искомые решения не предназначены для внедрения в систему и изменения ее функционирования. Этим они отличаются от тех (чаще всего оказывающихся предметом научных исследований), что разрабатываются для целей повышения производительности, выбора оптимальных решений и т. п.

[11]Примером одного из них (при поиске оценок сверху) является принадлежность прогнозных времен обслуживания классу случайных величин с убывающей функцией интенсивности. Отметим, что идеи использования (обычно выявляемых экспериментально) особенностей распределений времен обслуживания и распределений входящих потоков (для различных целей, в том числе и оптимизации работы систем) встречаются в научной литературе (см., например, [64–69]).

[12]Доступные из литературы результаты либо получены в отличных от рассматриваемых в диссертации предположениях, либо предназначены для, так сказать, исправления положения дел (см. сноску на предыдущей странице). Так в [70; 71], отталкиваясь от идей стратегий SRPT и PSJF, предложен класс правил (названный ε-SMART) для однолинейных систем, способных справляться с ошибками в прогнозных временах обслуживания. При этом предполагается, что в точности (!) известны максимальные погрешности. Подходящая дисциплина для некоторых многолинейных систем предложена в [72]. Дисциплины обслуживания, которые, опираются на (возможно неточные) данные о прошедших временах обслуживания, предлагаются в [73]. Задача оценки (условного) среднего времени пребывания поступающей заявки в системе $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, в зависимости от объема до-

тических исследований, результаты которых изложены в главе 1. Связаны они, главным образом, с развитием аналитического аппарата анализа стационарных характеристик **ранее не изучавшихся** классов СМО инверсионного типа[13].

Упомянутый выше вероятностный механизм является разновидностью предложенной в параграфе 1.1 специальной дисциплины обслуживания — инверсионный порядок обслуживания с обобщенным вероятностным приоритетом (далее — LIFO GPP) — и его содержание посвящено выводу основных стационарных характеристик СМО[14] $M_k \,|\, GI \,|\, 1 \,|\, n$ с этой дисциплиной. Не останавливаясь здесь на ее описании ввиду его громоздкости (см. стр. 37), отметим, что отличительной особенностью СМО этого класса является их возможная неконсервативность. С одной стороны она приводит к неприятным последствиям:

ступной информации, решается в [74]. Однако случай отсутствия наблюдений не рассматривается: минимум, известный всегда — это общее число заявок в системе и остаточное время обслуживания поступившей заявки. Необходимо здесь отметить [75, С. 72], где говорится о классе задач, возникающих, когда функция распределения времени обслуживания неизвестна. Однако, в отличие от диссертационной проблематики, здесь для того, чтобы воспользоваться аналитическими результатами, необходимо иметь оценки функции распределения по результатам наблюдений за функционированием системы (такие, как, например, в [76]). Тот же способ моделирования прогнозных времен обслуживания, что рассмотрен в диссертации (см. сноску на стр. 113), взят за основу в [77]; здесь предложена новая дисциплина обслуживания (модифицирующая известное правило FSP [78–80]), реализация которой в однолинейной системе позволяет обеспечить (в том числе и) справедливость обслуживания (см. [81–83], [84, Figure 1] и [73, Figure 4]). В [23], в предположении, что в однолинейной системе реализован алгоритм (машинного обучения), предсказывающий (по наблюдениям!) для поступающей заявки ее фактическое время обслуживания, дан вероятностный анализ ошибок предсказания при дисциплинах SRPT и SJF. В [85] предложен метод определения границ изменения основных характеристик систем $GI \,|\, GI \,|\, 1 \,|\, \infty \,|\,$ FIFO, в условиях недостаточной информации о распределениях входящего потока и времен обслуживания. Специальный случай ненаблюдаемых систем рассмотрен в [86]: здесь информация о (некоторых) заявках становится известной в моменты их (предполагаемого) начала обслуживания. Наконец, отметим работы [87; 88], посвященные игровым постановкам для ненаблюдаемых СМО.

[13]Несмотря на то, что эта тематика (входящая в направление исследований, связанное с изучением "неклассических" постановок задач в теории массового обслуживания, см. [59, С. 8–9]) является предметом исследований уже на протяжении более полувека, интерес к ней не ослабевает (см., например, [89–93]). Связано это, в частности, с тем, что разновидности дисциплины LIFO позволяют изучать как системы со сложными зависимостями (общие многолинейные СМО [94]), так и системы, функционирующие в особых условиях (например, когда времена обслуживания зависят от размера очереди [95], когда входящий поток — нерекуррентный (и не фазового типа) [96], когда система в переходном режиме [97]). Однако общий вариант инверсионного обслуживания, что изучается в диссертации, в научной литературе, по-видимому, ранее не освещался.

[14]Здесь обозначение M_k указывает на тот факт, что параметр входящего пуассоновского потока зависит от числа k заявок в системе.

например, нельзя сформулировать критерий существования стационарного режима (см. стр. 40). С другой стороны она (в некоторым смысле) вбирает в себя ту неопределенность, которая характерна для рассматриваемых частично наблюдаемых СМО. Опираясь на известную, развитую в ряде работ других авторов (см. [59; 98–109]) теорию систем со специальными дисциплинами обслуживания, в параграфе 1.1 доказаны теоремы, решающие в общем вопросы расчета стационарного распределения очереди, а также нахождения (в терминах преобразований [110]) стационарных распределений основных временных характеристик поступающих в систему заявок. Параграф 1.2 посвящен более подробному изучению важнейшего частного случая дисциплины LIFO GPP — инверсионный порядок обслуживания без прерывания и обслуживанием заново с новой реализацией длительности обслуживания (далее — LIFO Re). Привлекая развитый аналитический аппарат, а также другие известные приемы анализа, удалось существенно продвинуться в понимании работы СМО $M \,|\, GI \,|\, 1 \,|\, \infty$ с таким не сохраняющим работу обслуживанием (см. стр. 51), и выявить ряд ее неожиданных свойств[15]. Например, в этой СМО для любого распределения длины[16] заявки при достаточно малой интенсивности входного потока существует стационарный режим; стационарное распределение общего числа заявок в системе является геометрическим[17]; справедлив закон Литтла. В параграфе 1.3, используя аппарат матрично–аналитических методов, доказано, что аналогичным образом обстоит дело и в случае поступления в систему $r > 1$ пуассоновских потоков заявок различных типов. Отличительной

[15]Эта система относится к однолинейным СМО с прерываниями, которые исследованы в научной литературе очень хорошо; с подробным обзором можно ознакомиться по [111]. Поэтому неудивительно, что часть из представленных для нее результатов могут получаться как следствия уже известных. В частности, ПЛС (1.17) распределения времени пребывания заявки на приборе встречается в [112, Section 4.4]. В [93] исследована (отличными от использованными в диссертации методами) система $M \,|\, GI \,|\, 1 \,|\, \infty \,|\,$ LIFO PRD, критерий стационарности которой, как видно из [93, Theorem 5], совпадает с критерием в *Теореме 3*. В [93, Theorem 6] авторами получен более общий результат: критерий существования стационарного режима при рекуррентном входном потоке. Методика, использованная при изучении выходящего потока, следует [113].

[16]Следуя устоявшейся традиции (см. [59]), всюду в первой главе диссертации понятие “время обслуживания” заменено понятием “длина заявки”. Напомним, что связано это с тем, что в нестандартных СМО общее время обслуживания заявки из-за прерываний, обслуживания с отличной от единицы скоростью и т.п. может не совпадать с собственно временем ее обслуживания.

[17]Но не является нечувствительным к виду функции распределения длины заявки; в связи с этим вопросом см., например, [114; 115].

особенностью СМО[18] $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$ является отсутствие в ней приоритетов для входящих потоков (см. стр. 69), т.е. она не относится к хорошо известному классу приоритетных СМО [117–123]. Однако предположение о том, что в системе реализована дисциплина LIFO Re оказывается настолько сильным, что позволяет в итоге прийти и к совместному стационарному распределению общего числа заявок в системе, остаточной длины (и типа) каждой заявки, в том числе и заявки на приборе[19]. Удивительным на этом фоне выглядит то небольшое[20] число результатов, которое удается получить для аналогичных систем, но с несколькими приборами. Например, стационарное распределение общего числа заявок в системе (с двумя приборами) остается геометрическим, но невыясненным остается даже критерий его существования. Обобщениям полученных теоретических результатов на случаи более общих входящих потоков, посвящен параграф 1.4. Здесь развит аналитический аппарат [132] расчета стационарных характеристик (в терминах преобразований) однолинейных СМО с произвольным обслуживанием и либо неординарным пуассоновским потоком разнородных заявок[21], либо двумя конкурирующими потоками — основным

[18] Здесь и всюду в первой главе используется классификация СМО, принятая в книге [116, С. 25].

[19] Несмотря на огромный объем уже накопленных знаний по приоритетным СМО, представленные результаты стационарного анализа СМО $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$ в научной литературе не отмечались и, по-видимому, являются новыми. В связи с этим важно сделать два замечания. Эта система доставляет новый пример СМО, которая может быть исследована (в стационарном режиме) без теоремы Руше [124]. Примечательно и условие ее стационарности (см. *Теорему 9* на стр. 77). Оно подсказывает, что эта СМО относится к типу развиваемому в работах [125], и, судя по всему, связана со специальный понятием баланса, рассмотренным в [126; 127], или понятием позиционного баланса в [75, С. 85].

[20] И это косвенно служит еще одним подтверждением известному факту, что СМО $M \,|\, GI \,|\, 2 \,|\, \infty$ [128–131] являются чрезвычайно сложными для вероятностного анализа.

[21] Как было продемонстрировано еще в [108], для однолинейных систем с инверсионным порядком обслуживания и вероятностным приоритетом, возможны содержательные обобщения на случай потоков фазового типа, которые не являются рекуррентными и считаются более привлекательными при моделировании процессов в реальных технических системах [133; 134]. Несмотря на свою общность, модель потока фазового типа не подразумевает, что процесс поступления заявок в систему зависит от состояния самой системы. Не останавливаясь на возможных практических интерпретациях связей между входящим потоком и состоянием системы (см. [135]), отметим лишь, что исследованию СМО с такими зависимостями посвящено достаточно много работ (см., например, [136–138] и ссылки в них). Обычно предполагается, что в систему поступает пуассоновский поток второго рода (т. е. интенсивность потока зависит от общего числа заявок, находящихся в системе). Если же допускается поступление групп заявок, то обычно предполагается, что размеры (остаточные времена обслуживания) заявок в группе являются независимыми случайными величинами (не зависящими также и от размера группы). В исследованной в параграфе 1.4 СМО эти

групповым пуассоновским и потоком насыщения[22]. В обоих случаях предполагается, что в системе реализован частный случай дисциплины LIFO GPP — инверсионный порядок обслуживания с вероятностным приоритетом.

Имея теперь некоторое представление об основных теоретических результатах первой главы, вернемся к методу получения оценок для частично наблюдаемых СМО, речь о котором шла выше. Он заключается в следующем. Для имеющейся частично наблюдаемой стохастической системы обслуживания сначала фиксируется интересующая характеристика, стационарное распределение которой существует, и вычисляется ее значение. Затем, исходя из имеющейся о системе информации, выбирается СМО с некоторой разновидностью дисциплины[23] LIFO GPP, в которой значение искомой (или, возможно, другой) характеристики лучше рассчитанного прогнозного значения и близко к (неизвестному!) фактическому. Совершенно ясно, что приблизиться к фактическому значению, не зная его, можно не всегда. В главе 2, имея в виду получение оценок сверху[24], формулируется соответствующее достаточное условие (см. стр. 109) в виде принадлежности частично наблюдаемой СМО некоторому множеству (оно обозначается $\mathfrak{M}^*$), и показывается, что оно непусто (одним из его элементов является система $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$). Вопрос исчерпывающего описания $\mathfrak{M}^*$ остается открытым. Однако не приходится рассчитывать на то, что его мощность велика. В оставшейся части главы 2 (см. стр. 121 и далее) показывается, что расширение области применения предложенного метода возможно (по крайней мере) в том случае, когда интересующей характеристикой частично наблюдаемой СМО является стационарное среднее время

предположения ослаблены принципиально новым образом: рассмотрен неординарный пуассоновский поток, интенсивность которого может зависеть от общего числа заявок, находящихся в системе в момент поступления группы, причем размер поступающей группы и размеры заявок в ней имеют совместное произвольное распределение.

[22]Т. н. поток фоновых заявок (см., например, [139; 140]). Заметим, что такая СМО может быть трактована как система с прогулками (отключением прибора) при опустошении системы от заявок основного потока (см., например, [141; 142]). Таким образом, полученные результаты обобщают некоторые из тех, что известны для этого типа систем в научной литературе и, в частности, позволяют находить стационарное распределение вероятностей состояний при прямом порядке обслуживания.

[23]В том, что СМО со специальными дисциплинами обслуживания могут быть полезны при анализе СМО с классическими дисциплинами нет ничего нового. Такие случаи известны в научной литературе; см., например, [143; 144].

[24]Если же для частично наблюдаемых систем выполняются сформулированные условия, в которых неравенства заменены на противоположные (но, по-прежнему, нестрогие), то получаемые по методу оценки являются оценками снизу.

пребывания заявки в системе (или стационарное среднее число заявок в ней). Достаточное условие формулируется прежним способом и предъявляется еще один элемент из $\mathfrak{M}^*$ (система $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{PS}$). Полученные аналитические результаты подсказывают условия[25], при которых метод пригоден, по-видимому, для весьма широкого класса частично наблюдаемых СМО (включающего, например, даже неконсервативные системы). Обсуждению, в связи с этим обстоятельством, наиболее интересных случаев посвящен параграф 2.3 (см. стр. 127).

Основным объектом исследований во второй части диссертации (главы 3 и 4) является следующая математическая модель. В функционирующую в непрерывном времени частично наблюдаемую систему из $M \geqslant 2$ параллельно работающих серверов поступает рекуррентный поток заданий. Задания поступают по одному и имеют случайные объемы (размеры), причем размеры заданий являются независимыми одинаково распределенными случайными величинами (сл. в.). Каждое поступившее задание должно быть немедленно[26] направлено на один из серверов. Серверы работают независимо, без обмена заданиями и являются абсолютно надежными. В каждом сервере имеется очередь неограниченной емкости для хранения заданий и один процессор для обработки. Производительность по крайней мере одного процессора из M отличается от остальных. Наконец, выбор на обслуживание в каждом сервере происходит в соответствии с некоторой консервативной дисциплиной обслуживания.

Частичная наблюдаемость подразумевает, что при принятии очередного решения диспетчеру доступна только[27]

- априорная информация о системе (функция распределения интервала между поступлениями заданий, функция распределения их объема, производительности серверов), включая исчерпывающую информацию о состоянии серверов в момент начала функционирования, и
- информация о предыдущих моментах поступлений[28] заданий в систему и принятых им решениях.

[25] Это условия на загрузку системы и на распределение(я) прогнозных времен обслуживания; см. *Теорему 17* и *Теорему 18*, начиная со стр. 123.

[26] Т. е. диспетчер, осуществляющий этот выбор в автоматическом или ручном режиме в реальной системе, не имеет очереди для хранения заданий.

[27] Другими словами, недоступна динамическая информация о состоянии системы (например, о числе заданий на серверах, об остаточных временах обслуживания, о размерах заданий и т. п.).

[28] Поскольку момент принятия решения совпадает с моментом поступления очередного задания, то этот момент также считается известным диспетчеру.

Решаемая задача — построение (квази)оптимальной процедуры выбора сервера (диспетчеризации) для выполнения очередного задания. Во многих системах, включая распределенные компьютерные системы передачи и обработки данных, наиболее популярным критерием оптимальности является минимум стационарного среднего времени пребывания задания в системе (или, по-другому, стационарное среднее время отклика). В диссертации этот критерий взят за основной[29].

Более строго рассматриваемая задача диспетчеризации может быть сформулирована так. Обозначим через $F(x)$ функцию распределения (ф. р.) длины интервала между последовательными поступлениями заданий. Через $B(x) = \mathsf{P}\{S < x\}$ обозначим ф. р. размера задания. Индексируя серверы числами, начиная с единицы, будем обозначать производительность сервера m через $v^{(m)}$, предполагая ее конечной. Кроме того, по крайней мере, для одного m значение $v^{(m)}$ отличается от остальных[30]. Относительно дисциплины обслуживания будем считать, что она выбрана из множества[31] $\{\mathsf{FIFO}, \mathsf{LIFO}, \mathsf{RANDOM}, \mathsf{SJF}, \mathsf{PS}, \mathsf{PLIFO}, \mathsf{FB}, \mathsf{PSJF}, \mathsf{SPRT}\}$. Условимся обозначать последовательность моментов поступления заданий в систему через t_n (причем t_1 — момент поступления первого задания и $0 \leqslant t_1 < \cdots < t_n < \ldots$), а решение (действие, правило), принимаемое в момент t_n относительно вновь поступившего задания — через y_n. Пусть задание, поступившее в момент t_n и обслуженное согласно правилу y_n, проведет в системе время, равное V_n. Требуется найти

[29] Из научной литературы и практики хорошо известны и другие функционалы: минимум стационарного среднего времени ожидания начала обслуживания, минимум стационарного среднего значения slowdown (см. сноску на стр. 46), минимум квантилей стационарных распределений времен ожидания или пребывания и др. Для некоторых из них полученные в диссертации результаты (как будет видно из численных примеров) также приводят к близким к оптимальным правилам диспетчеризации. Принцип функционирования систем подобных описанной и алгоритмы управления потоками в них служат единственной цели — обеспечению заданных характеристик обслуживания заданий. Это и определяет характер целевых функционалов.

[30] И это предположение является существенным. Интуиция подсказывает, что в однородной системе (т. е. при $v^{(1)} = \cdots = v^{(M)}$) оптимальной (в рассматриваемых условиях частичной наблюдаемости) диспетчеризацией является та, что максимизирует время между двумя последовательными поступлениями в каждый сервер. Следовательно, по крайней мере, при рекуррентном входном потоке и рекуррентном обслуживании, оптимальной является циклическая стратегия (Round Robin), предписывающая направлять задание с номером n на сервер с номером $(n \bmod M) + 1$. Обсуждение этого факта (и доказательства) можно найти, например, в [145–147] и [148, Theorem 4.1]. Поэтому однородные системы из рассмотрения исключены.

[31] Конкретное множество выбрано здесь лишь для определенности и, в принципе, допускается любая консервативная дисцпилина обслуживания.

такую стратегию[32] $y = \{y_1, y_2, \dots\}$, которая минимизировала бы предельное среднее время пребывания задания в системе[33], определяемое как

$$\mathsf{E}V = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathsf{E}_y V_n, \tag{1}$$

где E_y — интегрирование по мере, порождаемой последовательностью y. В таком виде формулировка задачи остается неполной, поскольку не указано множество допустимых диспетчеризаций на котором осуществляется минимизация. В свою очередь задание множества допустимых диспетчеризаций можно интерпретировать с точки зрения возможностей наблюдения за системой. Поэтому его можно задать (и далее это будет сделано; см. (4)) из того условия, что рассматриваемые в диссертации стохастические системы с параллельным обслуживанием являются частично наблюдаемыми. Однако не будем пока этого делать и посмотрим на задачу с разных точек зрения, а именно — в зависимости от того, на каком множестве допустимых диспетчеризаций осуществляется минимизация. С такой позиции сразу будет видно и место, которое занимают результаты диссертационного исследования на общей картине.

Обозначим через h_n совокупность наблюдаемых параметров системы до момента принятия решения t_n. Тогда допустимое правило диспетчеризации имеет вид $y_n = f(h_n)$, где f — рандомизированная или детерминированная функция со значениями в множестве $\{1,2,\dots,M\}$, а h_n принимает значения из некоторого множества наблюдений (далее — H_n).

Вариант 1. Крайний случай, приводящий к наиболее бедному множеству допустимых стратегий — это отсутствие вообще каких-либо наблюдений, $H_n = \varnothing$. В этом случае допустимая стратегия описывается $(M-1)$ параметрами — вероятностями p_m выбора для очередного задания сервера m, т. е.

$$y_n = \begin{cases} 1, & \text{если } U < p_1, \\ \dots & \\ j, & \text{если } \sum\limits_{m=1}^{j-1} p_m \leqslant U < \sum\limits_{m=1}^{j} p_m, \\ \dots & \\ M, & \text{если } U \geqslant \sum\limits_{m=1}^{M-1} p_m, \end{cases} \tag{2}$$

[32] Всюду в диссертации слова стратегия, диспетчеризация, правило и алгоритм используются как синонимы.

[33] Предполагая, что оно существует.

где U — равномерно распределенная на [0,1] сл.в. Известен ряд результатов, касающихся оптимальности рандомизированных стратегий, используя которые можно численно находить значения вероятностей p_m (см., например, [149–155]). Наиболее полно задача решается для полностью марковских систем и систем с входящим пуассоновским потоком заданий и серверов типа $M\,|\,GI\,|\,1\,|\,\infty$. В общих случаях (например, когда система состоит из серверов типа $GI\,|\,GI\,|\,n\,|\,\infty$ или когда входящий поток — коррелированный (см. [133;134])) известны различные приближенные (и эвристические) решения, которые чаще всего получаются методами математического программирования. Далее всюду семейство таких стратегий обозначается RND[34]. Важным обстоятельством, которое позволяет упростить решение оптимизационной задачи в случае рандомизированной стратегии является то, что если поток поступающих в систему заданий является рекуррентным, то и "прореженный" поток на каждый сервер также является рекуррентным. Тогда система из нескольких серверов распадается на независимые системы из одного сервера, для которых можно использовать известные точные или приближенные формулы.

Вариант 2. Большее по сравнению с *Вариантом 1* разнообразие в выборе диспетчеризации получается, если допустить возможность наблюдения за траекторией принятых решений, что приводит к допустимым правилам вида

$$y_n = f(y_{n-k_n},\ldots,y_{n-1}),\ 1 \leqslant k_n \leqslant n-1, \tag{3}$$

В этом случае предыстория к моменту t_n определяется значением из множества $H_n = \{1,2,\ldots,M\}^{k_n}$, где число k_n характеризует глубину предыстории, используемую в момент t_n. Центральное место здесь занимают программные стратегии (далее всюду — PROG), т. е. стратегии, параметризуемые бесконечными последовательностями $\{a_1,a_2,\ldots,a_{n-1},a_n,\ldots\}$, в которых a_n означает, что n-е задание направляется на сервер с номером a_n. Внимание к ним связано с интуитивным представлением о том, что входящий поток на каждый сервер при программной стратегии является более регулярным (т. е. "менее" случайным), чем при вероятностной, что (как доказано для ряда случаев) приводит к уменьшению значения стационарного среднего времени пребывания задания в системе [160–163]. Для произвольного числа серверов нахождение

[34] От англ. random. В литературе, однако, встречаются и другие обозначения: PAP (Probabilistic Allocation Policy), BS (Bernoulli Splitting), RS (Random Splitting). Заметим, что подобные стратегии "появляются" не только в проблемах диспетчеризации; см., например, [156–158] и [159, Глава 5].

оптимальной программной стратегии является сложной задачей, решение которой за приемлемое время обычно найти не удается. Если стратегия PROG предписывает направлять на каждый из M серверов задания в соответствии с вероятностным распределением $\{d_1,\ldots,d_M\}$, то из [148; 164] известно, что оптимальными являются так называемые сбалансированные последовательности[35]. Однако и сбалансированные последовательности для произвольного вероятностного распределения $\{d_1,\ldots,d_M\}$ существуют лишь в редких случаях. Такими случаями являются система из произвольного числа одинаковых серверов[36] (здесь оптимальной является упоминавшаяся выше циклическая стратегия[37]) и случай двух серверов (здесь при рациональном значении d_1 оптимальной является так называемая последовательность Битти[38]). Заметим, что в последнем случае оптимальность стратегии зависит от значения d_1 и способа нахождения точного значения до сих пор не предложено (см., например, [168; 169]). Тем не менее, простой эвристический подход к нахождению значения d_1 приводит к значениям целевой функции, которые не удается уменьшить, не привлекая при диспетчеризации дополнительную информацию о системе. При $M \geqslant 3$ сбалансированную последовательность удается построить лишь в частных случаях (см., подробнее в [148; 170]). Поэтому действуют по-другому: для заданного вероятностного распределения $\{d_1,\ldots,d_M\}$ ищут детерминированную последовательность, расстояние[39] которой от сбалансированной последовательности было бы минимальным. Эта задача является комбинаторной, и для нее известно несколько численных алгоритмов решения (см. [166; 171; 172]). В наиболее важном случае — случае рациональных значений d_m — результаты работы этих алгоритмов приводят к периодическим последовательностям и последовательностям специального вида — так называ-

[35]Например, если последовательность состоит только из нулей и единиц, то она называется сбалансированной, если число единиц в любых двух произвольно выделенных подпоследовательностях фиксированной длины отличается не более, чем на единицу. Вообще говоря, понятие сбалансированной последовательности было введено еще в середине прошлого века (см. [165]), но не в контексте задач управления.

[36]Например, серверов типа $\cdot\,|\,G\,|\,1$.

[37]Т. е. n-ое задание направляется на сервер с номером $(n \bmod M) + 1$.

[38]В литературе встречаются и другие названия: последовательность Штурма, бильярдная последовательность (см. подробнее в [166; 167]). Заметим, что для реализации такой диспетчеризации вообще не требуются наблюдения за траекторией принятых решений, а для определения нужного сервера необходимо знать лишь порядковый номер поступающего задания

[39]Подробнее о том, как задается расстояние между последовательностями см. [166].

емым бильярдным последовательностям (обладающих хорошими свойствами, например, минимальным дисбалансом; см. [173]).

Вариант 3. Появление возможности наблюдения за состоянием серверов или очередей в них, позволяет пополнить множество допустимых диспетчеризаций, описанное в *Варианте 2*. В новых условиях обычно рассматриваются правила вида

$$y_n = f\left(N_n^{(1)}, \ldots, N_n^{(M)}\right)$$

где $N_n^{(m)}$ — число заданий в сервере m к моменту принятия решения в момент t_n. Таким образом, $H_n = \{0,1,2,\ldots\}$. Наиболее известным примером здесь является диспетчеризация по наикратчайшей очереди (далее JSQ), предписывающая направлять поступающее задание на сервер с минимальной очередью. Она является оптимальной в случае пуассоновского входящего потока, экспоненциальных времен обслуживания с одинаковыми параметрами и с дисциплинами FIFO в серверах (подробнее см., например, обзоры [174; 175]).

Вариант 4. Наиболее полный вариант наблюдений предполагает возможность использования при выборе сервера в момент t_n значений незаконченной работы (по каждому заданию) в каждом сервере (пусть $\vec{W}_n^{(1)}, \ldots, \vec{W}_n^{(M)}$) и размера S_n нового задания. В этом случае оптимальная диспетчеризация находится в множестве стратегий вида

$$y_n = f\left(N_n^{(1)}, \ldots, N_n^{(M)}, \vec{W}_n^{(1)}, \ldots, \vec{W}_n^{(M)}, S_n\right).$$

Среди диспетчеризаций такого вида простым и в то же время достаточно эффективным решением является стратегия[40], известная в зарубежной литературе как Myopic, когда вновь поступающее задание посылается на тот сервер, который минимизирует время, необходимое для освобождения от заданий всей системы целиком, в предположении, что в дальнейшем задания в систему не поступают. Известны более сложные и менее универсальные алгоритмы (например, Deep из [191]), основанные на теории марковских процессов принятия решений. Упомянутая стратегия[41] Deep, возможно, вообще является квазиоптимальной в том смысле, что она дает значение целевой функции, близкое

[40] Упомянутая здесь по той единственной причине, что она еще встретится в диссертации. Вообще число работ, посвященных диспетчеризациям такого вида, огромно. Поскольку для диссертации они не представляют интереса, ограничимся упоминанием лишь нескольких: [176–188]. Некоторый обзор методов динамического распределения нагрузки до 2014 г. можно найти в [189]. См. также [190].

[41] Требующая, однако, чтобы входящие потоки заданий были простейшими [192].

к (неизвестному) оптимальному значению. Однако, несмотря на достигнутые успехи формального сведения задачи диспетчеризации при полном наблюдении к марковскому процессу принятия решений, до сих пор не известно, как находить оптимальное значение функционала (1).

Разумеется описанные выше четыре варианта наблюдений системы не исчерпывают все возможные постановки. Однако обычно встречающиеся задачи "укладываются", по крайней мере, в один из них. Не является исключением и рассматриваемая в диссертации задача диспетчеризации в частично наблюдаемых стохастических системах с параллельным обслуживанием (см. стр. 13). В ней недоступны наблюдения, отражающие состояния серверов (число заданий в очередях, объем незаконченной работы, моменты начала и окончания непосредственного выполнения заданий и т. п.). Неизвестным также считается размер поступающего задания. Однако информация о совершенных действиях понимается несколько более широко, чем выше в *Варианте 2*. Побудительный мотив можно выразить такими словами: зная, "что было сделано", естественно допустить, что известно также, "когда было сделано". Точнее говоря, помимо самих решений y_n, считается известной информация о моментах времени t_n, в которые эти решения принимались. Таким образом, допустимыми являются диспетчеризации, правила которых основываются на предыстории принятых решений и моментов поступления заданий или, другими словами, правила, представимые (детерминированной или рандомизированной) функцией вида

$$y_n = f\left(y_1, \ldots, y_{n-1}, t_1, \ldots, t_n\right), \tag{4}$$

а множество доступных наблюдений к моменту поступления n-го задания есть $H_n = \{1,2,\ldots,M\}^{n-1} \times (0,\infty)^n$.

Итак, в рассматриваемых частично наблюдаемых стохастических системах с параллельным обслуживанием диспетчеризация осуществляется в условиях, когда не наблюдаемы традиционно важные для решения задач оптимизации характеристики[42]. Более того, не наблюдается даже показатель, подлежащий минимизации. Поэтому большинство как диспетчеризаций[43], так и приемов решения, известных из научной литературы, неприменимы для достижения цели

[42] В частности, это не позволяет пользоваться мощным правилом — индексом Гиттинса (см., например, [193, Глава 9] и [194]): не наблюдаются прошедшие времена обслуживания!

[43] Отметим наиболее известные из научной литературы стратегии: JSQ, HJSQ(d), MEST, MERL, LWL, Myopic, SITA-E, SITA-V, VITA, C-MU, LAVA, TDP, FPI, TAGS, TAPTF. Более подробный список можно составить, например, по обзорам [174; 175].

— минимизации (1) на множестве стратегий (4). Строго говоря, известных на данный момент решений всего два[44]: использовать либо раномизированную стратегию (см. *Вариант 1*), либо программную стратегию (см. *Вариант 2*). Остановимся на них подробнее[45].

Проблема нахождения оптимального[46] набора $(p_1, \ldots, p_M)$ для стратегии RND хорошо известна (см., например, обзор в [149, Section 1, 2] или книгу [152]). Например, при пуассоновском потоке заданий (пусть со средним λ^{-1}) значение (1) совпадает со значением суммы

$$\sum_{m=1}^{M} p_m \mathsf{E}V(m), \tag{5}$$

где $\mathsf{E}V(m)$ — стационарное среднее время пребывания задания в сервере m, который теперь представляет собой классическую СМО $M\,|\,GI\,|\,1\,|\,\infty$ с интенсивностью входящего потока λp_m и распределением времени обслуживания $B(xv^{(m)})$. Поэтому искомые вероятности p_m суть решения задачи минимизации (5) при котором $\sum_{m=1}^{M} p_m = 1$, а также загрузка каждого сервера

[44] Ср. (2) и (3) с (4).

[45] Отметим доступные в литературе работы, близко примыкающие к рассматриваемой задаче. В [195] минимизируется стационарное среднее время пребывания задания в системе, однако предполагается, что диспетчеру известны моменты окончания выполнения заданий. В работе [196], хотя и не предполагается наличия очереди для хранения заданий, но считается, что диспетчеру известен их размер. Экспоненциальная система из двух однопроцессорных серверов рассмотрена в [197]: аналитически исследуются свойства пороговой стратегии в предположении, что диспетчер наблюдает точное состояние одного из серверов. Свойства программной стратегии в двухсерверной системе с произвольным входным потоком (но экспоненциальным обслуживанием) изучаются в [198]. Многосерверная частично наблюдаемая система рассмотрена в [199]: здесь предполагается, что имеется несколько независимых диспетчеров, каждый из которых использует рандомизированную стратегию. Результаты асимптотической оптимальности некоторых программных стратегий для рассматриваемых частично наблюдаемых систем получены в [200]. Вопрос построения периодической стратегии для заданий фиксированного размера, поступающих через одинаковые промежутки времени изучается в [201] (однако, например, полученно решение неприменимо в случаях, когда все серверы имеют различные производительности). В [202] сделано предположение, что точное состояние системы становится ивестным через некоторое, фиксированное заранее число поступлений (пусть N). Тогда исходная задача сводится к нахождению алгоритма диспетчеризации N заявок по M серверам при известном начальном состоянии последних. В предположениях экспоненциального обслуживания и дисцилипны FIFO, предложен алгоритм градиентного типа (основанный на принципе максимума Понтрягина), который дает близкое к оптимальному решение задачи.

[46] С разных точек зрения, в том числе и с точки зрения минимизации (1), и, конечно, не только для систем с одним пуассоновским потоком заданий (см., например, работы [203; 204] и ссылки в них).

меньше единицы[47]. Если же входящий в систему поток заданий — рекуррентный (пусть со средним λ^{-1} и коэффициентом вариации C_F), то сервер m (при прочих равных) представляет собой СМО $GI\,|\,GI\,|\,1\,|\,\infty$ со средним временем между поступлениями $(\lambda p_m)^{-1}$ и коэффициентом вариации $\sqrt{1+(C_F^2-1)p_m}$. В этом случае остается надежда только на приближенные методы нахождения близкого к оптимальному набора $(p_1,\ldots,p_M)$. Сформулированная задача насколько хорошо известна в научной литературе[48], что вне сомнений для большинства классических дисциплин обслуживания[49] уже известны приемы ее решения. Для целей диссертации интерес представляют, главным образом, две[50] из них: обслуживание в порядке поступления (FIFO) и обслуживание при справедливом разделении процессора (PS). При пуассоновском входящем потоке и дисциплине FIFO оптимальные с точки зрения минимума (1) вероятности p_m находятся численно, как решение модифицированной задачи PA1 из[51] [149]; например, при дисциплине PS решение может быть выписано в явном виде[52]:

$$p_m = \begin{cases} 0, & \text{если } 0 < \lambda \leqslant r_m, \\ \frac{1}{\lambda}\left(\frac{v^{(m)}}{\mathsf{E}S} - \frac{\sqrt{\frac{v^{(m)}}{\mathsf{E}S}}}{\sum\limits_{i=1}^{M^*}\sqrt{\frac{v^{(i)}}{\mathsf{E}S}}}\left(\sum\limits_{i=1}^{M^*}\frac{v^{(i)}}{\mathsf{E}S} - \lambda\right)\right), & \text{иначе,} \end{cases} \tag{6}$$

где

$$r_m = \sum_{i=1}^{m}\left(\frac{v^{(i)}}{\mathsf{E}S} - \sqrt{\frac{v^{(i)}}{\mathsf{E}S}\frac{v^{(m)}}{\mathsf{E}S}}\right), \quad r_{M+1} = \sum_{i=1}^{M}\frac{v^{(i)}}{\mathsf{E}S},$$

и $M^* = \operatorname{argmin}_{1\leqslant m\leqslant M}(r_m < \lambda \leqslant r_{m+1})$. При рекуррентном потоке, как уже было сказано выше, приходится пользоваться аппроксимациями (см., например, обсуждение на стр. 216). В диссертации, впрочем, они использовались

[47] Т. е. $0 \leqslant p_m\lambda\mathsf{E}(S/v^{(m)}) < 1$ для каждого m. Чтобы решение "получилось" и $\mathsf{E}V(m)$ должны существовать. Дополнительные ограничения зависят от принятой в сервере m дисциплины обслуживания.

[48] Под разными именами, например: проблема распределения потоков в [205, Раздел 5.8] и [206], проблема динамической маршрутизации в [207].

[49] И, по крайней мере, простейших входящих потоках.

[50] Хотя речь зайдет и о более экзотической, но хорошо известной дисциплине — дисциплине преимущественного обслуживания наикратчайшего задания с прерыванием обслуживания (SRPT); см. стр. 207.

[51] Можно было бы сослаться и на другие источники (например, [208, Раздел 4.3.2]), но здесь удобен этот т. к. в нем указан явный вид вероятностей p_m, минимизирующих родственный (1) функционал — стационарное среднее время ожидания заданием начала обслуживания (см. соотношение (2.8) в [149]).

[52] По соотношению (28) из [150]; см. также [184; 209; 210].

лишь для контроля результатов, так или иначе получаемых с помощью метода Монте–Карло, и во всех вычислительных экспериментах уступали последним.

Проблема нахождения оптимальной программной стратегии[53] т. е. бесконечной последовательности $\{a_1,a_2,\ldots,a_{n-1},a_n,\ldots\}$, $a_n \in \{1,\ldots,M\}$, известна в научной литературе почти так же хорошо[54], как и проблема поиска оптимальных параметров диспетчеризации RND. Решают ее обычно в два этапа[55]. Сначала находится наилучшее (с точки зрения выбранного критерия) вероятностное распределение $\{d_1,\ldots,d_M\}$, где, напомним, d_m — доля заданий направляемых на сервер m. Затем ищется детерминированная последовательность, сохраняющая доли d_m и обеспечивающая максимальное расщепление потока по серверам. Известно (см. [218–220]), что в случае двух серверов (каждый из которых представляет собой СМО[56] $GI\,|\,GI\,|\,1\,|\,\infty$ с дисциплиной FIFO) при рациональном d_1 оптимальной в классе стратегий (3) является последовательность Битти:

$$a_n = \lfloor (n+1)d_1 + \varphi \rfloor - \lfloor nd_1 + \varphi \rfloor. \tag{7}$$

Здесь $\varphi \in (-\infty,\infty)$ — произвольное число, обусловливающее только сдвиг детерминированной последовательности, и в рассматриваемых проблемах не влияет на значение целевого функционала. По-другому обстоит дело со значением d_1. Здесь имеет место сильная зависимость и, как уже упоминалось выше, универсального способа нахождения его точного значения до сих пор не предложено. В общем случае при $M \geqslant 3$ оптимальных правил, подобных (7), в научной литературе нет. Однако имеется ряд процедур для порождения хороших последовательностей $\{a_1,a_2,\ldots,a_{n-1},a_n,\ldots\}$. Судя по вычислительным экспериментам, из доступных в научной литературе программных стратегий[57],

[53] Отметим, например, что в [207, С. 29–31] она названа лучшим детерминированным алгоритмом. Там же дается обзор работы [211], в которой его предлагается искать путем максимизации некоторой функции (энтропии); см. также [212].

[54] Помимо упомянутых ранее работ, см., например, [198; 213–217] и ссылки в них.

[55] См., например, [207, С. 30].

[56] На самом деле допускается любая модель сервера, чье поведение (например, динамика процесса незаконченной работы) линейно в терминах $(\max,+)$ алгребры (см. [219; 221] и [148, Section 3.1]).

[57] См. алгоритм в [211], алгоритмы GRR, CGRR, and mBS в [171; 172], GR в [222], GG в [166]. Отметим, что некоторые нижние границы для значений целевых функционалов при использовании программных стратегий даны в [160; 166; 223].

имеющих широкую область применения, наилучшие результаты[58] удается достичь с помощью так называемого "жадного" алгоритма из [166, С. 184]:

$$a_n = \mathrm{argmin}_{1 \leqslant m \leqslant M} \left(\frac{x_m + \kappa^m(n-1)}{d_m} \right), \tag{8}$$

где $\kappa^m(n-1)$ обозначает суммарное число заданий (из первых $n-1$), направленных на сервер m, $x_1, \ldots, x_M$ — произвольные[59] числа из [0,1], а неоднозначность при нахождении минимума (здесь и всюду далее) разрешается в пользу самого быстрого сервера и, если их несколько, — равновероятным выбором. Еще раз отметим, что нахождение оптимальной программной диспетчеризации является трудной оптимизационной задачей, пока не имеющей решения[60]. И известные приемы нахождения оптимального набора для стратегии RND не облегчают положение, поскольку оптимальные с точки зрения одного и того же критерия наборы $\{p_1, \ldots, p_M\}$ и $\{d_1, \ldots, d_M\}$ могут отличаться (см., например, таблицу 4 на стр. 143.). Вместе с тем, выбор стратегии PROG в классе стратегий (3), даже при условии использования эвристических приемов для нахождения значений неизвестных параметров, приводит к таким значениям целевых функций, которые не удается уменьшить, не привлекая при диспетчеризации дополнительную информацию о системе[61].

Итак частичная наблюдаемость является весьма жестким ограничением на допустимые стратегии диспетчеризации, которое влечет существенный проигрыш в целевой функции по сравнению со стратегиями, использующими максимальную информацию. Оба описанных выше подхода к диспетчеризации в частично наблюдаемых системах с параллельным обслуживанием являются плодотворными, однако обладают и рядом недостатков. Во-первых, качество предоставляемых ими решений сильно зависит от предположений о характерах потоков и процессах обслуживания, и избираемых приемов для преодоления

[58] Что именно так, по-видимому, обстоит дело подчеркивается и в других исследованиях; см., например, [201, С. 189–190].

[59] Как показано в [224], большего можно добиться, наведя здесь порядок, а именно — положив $x_m = 1$, если сервер m является самым производительным, и $x_m = 0$ иначе.

[60] И трудность, главным образом, связана с тем, что малое изменение значений в наборе $\{d_1, \ldots, d_M\}$ может привести к большому изменению периода последовательности; см. пример на стр. 9 в [225]. Там же предложен алгоритм эволюционной оптимизации для преодоления этой трудности. В диссертации, однако, он рассматривается т. к. (по крайней мере, в рамках рассматриваемой задачи) получаемые с его помощью результаты либо совпадают, либо уступают тем, что получаются с помощью более простого и универсального правила (8).

[61] Этот эффект хорошо известен в научной литературе (см., например, [161]).

возникающих трудностей. Во-вторых, не ясно приводят ли они к оптимальным в классе стратегий (4) результатам. Наконец, обе стратегии RND и PROG, ввиду своей универсальности, не дают глубокого понимания того, как должна управляться именно частично наблюдаемая система. Главная цель второй части диссертации — сформировать такое понимание. Идея, благодаря которой это стало возможным, состоит в использовании при диспетчеризации всей доступной предыстории наблюдаемых компонент (ср. (3) с (4)). Однако не является ни очевидным, ни интуитивно понятным дает ли такая незначительная дополнительная информация (учет моментов поступления заданий) возможность улучшить (по сравнению с RND и PROG) значения целевых функционалов по типу (1). Оказывается, что такая возможность принципиально есть. В главах 3 и 4 **впервые** показано, что соответствующие стратегии существуют (далее они всюду обозначается AA от англ. Arrival Aware) и предложено несколько конструктивных способов для их порождения. Интуитивная предпосылка для положительного ответа заключается в следующем соображении: большие промежутки времени между поступлениями заданий повышают вероятность того, что серверы находятся в состояниях с меньшей незаконченной работой и наоборот. Чтобы воспользоваться этим довольно расплывчатым соображением, необходимо уметь на основании доступной информации в момент поступления очередного задания получать хотя бы приближенную оценку либо целевого функционала, либо связанных с ним величин (в случае (1) — это, например, $\mathsf{E}V_n$). Эта необходимость определила и структуру второй части диссертации. В главе 3 сначала описывается аналитический подход к воплощению идеи диспетчеризации по предыстории (параграфы 3.1 и 3.2). Затем излагается более универсальный, аналитико–имитационный подход, значительно расширяющий тот круг систем, который очерчен результатами предыдущих параграфов. Наконец, в главе 4 речь идет о принципиально другом, простом и универсальном подходе к диспетчеризации по предыстории, свободном от тех вычислительных недостатков, которые присущи предыдущим двум подходам. Если судить только по значениям целевых функционалов, то с увеличением номера параграфа убывает эффективность предложенных в нем решений. Алгоритмы параграфа 3.1 являются наилучшими в сравнении с ранее известными в научной литературе алгоритмами[62] для частично наблюдаемых стохастических систем с параллельным обслуживанием во всем диапазоне изменения

[62] Т. е., как следует из данного выше обзора, в сравнении с RND и PROG.

значений исходных параметров, а алгоритмы главы 4 — главным образом в области низкой загрузки. С точки же зрения вычислительной сложности, дело обстоит с точностью до наоборот: результаты главы 4 являются самыми простыми и универсальными. Обычно новые диспетчеризации получаются параметрическими[63]. Из-за предположения о ненаблюдаемости целевого функционала, для оценки их параметров (и параметров стратегий RND и PROG), необходимо привлекать компьютерную модель исходной системы. Будучи основанными на принципиально иной идее, новые диспетчеризации применимы при общих предположениях о распределениях входящих потоков и размерах заданий, в случае наличия нескольких потоков, многопроцессорных серверов и т.п. Кроме того, в наиболее типичных условиях они гарантируют выигрыш, а в наихудших — паритет с наилучшей из ранее известных в научной литературе стратегией (PROG). Забегая вперед отметим, что такое преимущество дается не бесплатно: теряется простота реализации, свойственная стратегиям RND и PROG. Тем не менее, результаты глав 3 и 4 диссертации дают основание утверждать, что диспетчеризация по предыстории — это именно то, как должны управляться рассматриваемые частично наблюдаемые стохастические системы с параллельным обслуживанием.

Посмотрим подробнее на результаты второй части диссертации. В первом параграфе главы 3 (параграф 3.1) изложено решение задачи диспетчеризации в стохастических системах с параллельным обслуживанием, в которых для величин, связанных с целевым функционалом, можно получить вычислительно реализуемые точные или хорошие приближенные формулы расчета. Наиболее простой, но вместе с тем и наиболее часто встречающейся в научных исследованиях и задачах практики, является система с дисциплиной FIFO на серверах.

[63] Однако диспетчеризацию практически без параметров можно предложить. Например, см. (3.2) и *Алгоритм I* на стр. 137. Задавшись малым значением Δ, *Алгоритм I* можно реализовать в диспетчере и (без предварительных экспериментов на компьютерной модели!) быть вполне уверенным в эффективности принимаемых решений (в данном случае, с точки зрения минимизации (1)). Применить ранее известные стратегии (RND и PROG) без предварительной оценки параметров тоже можно. Для этого достаточно установить им такие значения, чтобы нагрузка балансировалась между серверами пропорционально их производительности т. е. $p_m = d_m = v^{(m)} / \sum_{m=1}^{M} v^{(m)}$ (см. также обсуждение на стр. 215). Однако здесь уже нет никаких оснований расчитывать на хорошее качество решений, поскольку, например, значения параметров не зависят от целевого функционала (см. таблицу 24 на стр. 205).

Она и была выбрана[64] полигоном[65] для демонстрации возможностей нового подхода к управлению в частично наблюдаемых системах с параллельным обслуживанием. Одним из вариантов диспетчеризации по полной предыстории[66] здесь является правило (см. стр. 135): отправить задание, поступившее в момент t_{n+1}, на сервер с номером y_{n+1}, где

$$y_{n+1} = \operatorname{argmin}_{1\leqslant m\leqslant M}\left(\mathsf{E}W_{n+1}^{(m)} + \frac{\mathsf{E}S}{v^{(m)}}\right),\ n \geqslant 0,$$

а сл. в. $W_{n+1}^{(m)}$ обозначает время, необходимое для выполнения всех заданий, имеющихся на сервере m в момент t_{n+1}, без учета задания, поступившего в этот момент[67]. Будучи избранными для удобства, обозначения скрывают тот факт, что математические ожидания сл. в. $W_{n+1}^{(m)}$ являются условными и зависят от распределений размеров первых n заданий и моментов их поступлений. Хотя изучению процессов незаконченной работы в СМО и СеМО посвящено огромное число работ, проблема вычисления[68] моментов сл. в. $W_{n+1}^{(m)}$ практически не освещена. Известные в научной литературе результаты (см. [226–249], [250, § 7] и ссылки в них) не позволяют прийти к ее решению, поскольку задающие $W_{n+1}^{(m)}$ сл. в. являются зависимыми. Бесплодными оказываются и попытки воспользоваться известными неравенствами для функций от сл. в. (см., например, [251–262]): получаемые на их основе оценки математических ожиданий оказываются настолько грубыми, что стратегия $\{y_1,y_2,\dots\}$ перестает различать серверы. Фактически единственным выходом из положения является использование основного рекуррентного соотношения, связывающего величины $W_{n-1}^{(m)}$ и $W_n^{(m)}$ — хорошо известной в литературе рекурсии Линдли[69]. И в подразделе 3.1 предложен рекуррентный алгоритм (см. стр. 136) приближенного расчета значений $\mathsf{E}W_{n+1}^{(m)}$, требующий при каждом n конечного

[64] При несколько более общих (чем те, что сделаны выше) предположениях о распределениях размеров заданий.

[65] Но и о системах с другими дисциплинами обслуживания тоже пойдет речь.

[66] С целью минимизации (1).

[67] Другими словами, $W_{n+1}^{(m)}$ — незаконченная работа на сервере m к моменту t_{n+1}.

[68] И, в целом, подобных характеристик, зависящих от всей (или хотя бы части) предыстории функционирования системы.

[69] См., например, [263, С. 449]. В связи с этой конструкцией в научной литературе имеется много результатов. Помимо упомянутых выше работ отметим еще [264], где (дан некоторый обзор и) для дискретной СМО $GI\,|\,GI\,|\,1\,|\,\infty$ предложен матрично–аналитический метод расчета некоторых стационарных распределений, не предполагающий конечных носителей у распределений.

числа операций сложения и умножения, даже в случае распределений, сосредоточенных на всей положительной полуоси. Сам по себе этот результат не является новым и, по-видимому, похоронен глубоко в литературе[70]. Но основные успехи диспетчеризации по предыстории, связаны не с ним, а с найденной его новой модификацией (см. стр. 140), имеющей заметно меньшую вычислительную сложность. Первая часть параграфа 3.2 посвящена изложению результатов вычислительных экспериментов[71], которые свидетельствуют о том, что для рассматриваемых частично наблюдаемых систем с параллельным обслуживанием алгоритмы диспетеризации по предыстории являются равномерно наилучшими. Почти во всем диапазоне изменений значений исходных параметров системы они позволяют уменьшить значения функционалов (типа (1)) по сравнению со всеми ранее известными из научной литературы стратегиями (т. е. RND и PROG). Когда условия таковы, что оптимизация невозможна, новые алгоритмы приводят к тем же значениям, что и наилучшие из ранее известных. Обсуждению и всевозможным дополнениям, расширяющим круг систем для которых применим аналитический подход к диспетчеризации по предыстории, посвящена оставшаяся часть параграфа 3.2. В частности, здесь показано как может быть видоизменено управление, если в серверах реализована принципиально отличная от FIFO дисциплина обслуживания. Например, при дисциплине PS относительно поступившего в момент t_{n+1} задания, следу-

[70]Впрочем недавно он появился в [265] в связи с некоторыми приложениями теории массового обслуживания, но без ссылок на источник. Отметим также, что работа Л. Такача [266] является, судя по всему, единственной в литературе, которая содержит в явном виде результат, позволяющий (хотя бы теоретически) рассчитывать необходимые для диспетчеризации величины (см. Theorem 4). Однако для целей диссертации он малопригоден т. к. требует при каждом n обращения производящих функций.

[71]Значения целевых функций, указанные в таблицах, были получены, в основном, путем имитационного моделирования и являются арифметическими средними из результатов наблюдений. Хотя можно было бы действовать и по-другому (см., например, [267] и [268, Section 7]), поскольку во всех случаях выполнялся закон Литтла. Для принятия решения о прекращении моделирования использовался инженерный подход: моделирование продолжалось до тех пор, пока не переставала меняться третья значащая цифра арифметического среднего. Для целей диссертации такой путь представляется ничем не уступающим другим, известным из литературы (например, основанным на центральной предельной теореме, или неравенстве Чебышева или [269]). По накопленным к моменту окончания имитации наблюдениям вычислялись и встречающиеся в некоторых таблицах оценки стандартного отклонения. При наличии обоих значений можно составить некоторое представление о доверительных границах для среднего. Наконец отметим, что при моделировании использовались стандартные генераторы (пседво)случайных последовательностей. Методы понижения дисперсии не применялись.

ет принять решение

$$y_{n+1} = \operatorname{argmin}_{1\leqslant m\leqslant M}\left(\theta\cdot \mathsf{E}N_{n+1}^{(m)}\right),\ n\geqslant 0,$$

где сл. в. $N_{n+1}^{(m)}$ — число заданий в сервере m в момент t_{n+1} (но до прибавления задания к какому-либо серверу), а $\theta\in(0{,}1]$ — наперед заданное число. Наличие, вообще говоря, неизвестного постоянного коэффициента (θ) в правиле y_{n+1} неслучайно. Как уже было сказано ранее, все новые алгоритмы, реализующие управление по предыстории, являются параметрическими. Цель же введения постоянного коэффициента (который далее будем называть порогом[72]) — компенсация тех изменений в исходной задаче, которые вызываются различного рода аппроксимациями, необходимыми для расчета величин $\mathsf{E}N_{n+1}^{(m)}$, $\mathsf{E}W_{n+1}^{(m)}$ и др. Поскольку, судя по вычислительным экспериментам, в каждой задаче существует единственное оптимальное значение порога, то наличие хорошего (хотя бы в каком-то смысле) начального приближения заметно упрощает поиск. Например, за такое начальное значение можно взять оптимальное значение порога в какой–нибудь аналогичной задаче, но с полным наблюдением. Завершает параграф 3.2 одна из таких задач, решение которой может служить (и в параграфе 3.4 служит) начальным приближением для значений порогов в алгоритмах диспетчеризации по полной предыстории. Речь идет о вычислении оптимальных[73] значений параметров пороговых стратегий в одном классе полностью наблюдаемых систем с параллельным обслуживанием. Пороговое управление[74] — одна из самых известных и популярных стратегий в прикладных задачах теории вероятностей[75]. Ее популярность связана как с простотой реализации, так и с тем обстоятельством, что в ряде случаев она является оптимальной (см., например, [277; 278]). В большинстве случаев задача нахождения значений параметров управления (порогов) не поддается аналитическому решению: необходимо прибегать к приближенным методам[76] или

[72] Правомочность использования здесь этого термина придется отложить до стр. 161.

[73] С точки зрения минимума стационарного среднего времени пребывания задания в системе. Допускается и варьирование (не в очень широких диапазонах) целевой функции. Например, очевидные изменения в (3.18) позволяют говорить об оптимальности с точки зрения минимума стационарного среднего времени ожидания начала обслуживания.

[74] Или, по-другому, переключающая стратегия.

[75] Например, задачи распределения ресурсов (см. [270–275]), задачи о разладке (см. [276] и [193, С. 290]).

[76] Например, решать уравнения динамического программирования, произведя предварительно дискретизацию множества состояний.

статистическому моделированию. В последнем случае, при пологих графиках целевой функции вблизи оптимальных значений порогов, для достижения высокой точности может потребоваться очень большое время имитации. В связи с этим возникает следующий вопрос. Предположим, что в полностью наблюдаемой системе с параллельным обслуживанием следует использовать пороговую стратегию. Можно ли в этом случае предложить алгоритм нахождения (хотя бы приближенно) значений оптимальных порогов, который основан только на вероятностных соображениях и свойствах пороговой стратегии, и не использует какие-либо результаты имитационного моделирования? Ограничившись классом полностью наблюдаемых систем с параллельным обслуживанием, для которого в настоящее время известно мало результатов, касающихся вопросов оптимальности, в конце параграфа 3.2 описан итерационный алгоритм (см. стр. 161), обладающий требуемыми свойствами. Опишем вкратце постановку задачи, идею и особенности предложенного решения. Пусть имеется полностью наблюдаемая система, в которую поступает один рекуррентный поток заданий одинакового размера. В системе имеется $M \geqslant 2$ серверов[77], работающих параллельно и независимо друг от друга, перед каждым из которых есть очередь неограниченной емкости для хранения заданий, которые предназначаются для обработки именно на этом сервере т. е. переход между очередями невозможен. Очередное задание при поступлении в систему направляется на один из серверов в соответствии со следующей пороговой стратегией с параметрами $\xi^{(1)}, \ldots, \xi^{(M)}$: n-е задание направляется в очередь к серверу $\operatorname{argmin}_{1 \leqslant m \leqslant M} \left(x_m + \xi^{(m)}\right)$, где x_m — незаконченная работа на сервере m в момент поступления n-го задания. После окончания обслуживания задания покидают систему. Дисциплина выбора на обслуживание из очереди — в порядке поступления. Задача — найти значения порогов $\xi^{(1)}, \ldots, \xi^{(M)}$, при которых достигается минимум стационарного среднего времени пребывания задания в системе. Отметим, что производительности серверов предполагаются фиксированными и различными[78]. Известно, что при $M = 2$ рассматриваемая задача эквивалентна известной задаче о медленном приборе [281; 282], для которой по-

[77] Занумерованных числами от 1 до M без повторений.

[78] Случай серверов одинаковой производительности исключается, поскольку здесь ответ известен [146; 279; 280]: $\xi^{(1)} = \cdots = \xi^{(M)} = 0$ т. е. оптимальной с точки зрения стационарного среднего времени пребывания в системе является стратегия LWL, предписывающая направлять очередное задание в сервер с наименьшей незаконченной работой. Таким образом, система с параллельным обслуживанием оказывается эквивалентной СМО $GI \,|\, D \,|\, N \,|\, \infty$.

роговая стратегия является оптимальной[79] (см. [272, Лемма 1] или [277; 284]). При этом оптимальные значения порогов для этих двух задач связаны простым соотношением [272, Лемма 2]. Даже в случае двух серверов не удается найти значение оптимального порога в явном виде. Отсутствуют аналитические результаты о зависимости целевой функции от значения порога, что не позволяет применять, как стандартные приемы нахождения минимума, так и известные эффективные градиентные методы в сочетании со статистическим моделированием [4; 5]. Судя по публикациям в открытой научной печати, последний шаг вперед в этой задаче удалось сделать в [272]: предложен аналитически–имитационный метод, использующий некоторые результаты из теории марковских процессов принятия решений, а также методы Монте–Карло для оценки значений входящих в уравнения величин[80]. Предложенный в диссертации прием нахождения параметров оптимальной пороговой стратегии не использует какие-либо результаты имитационного моделирования, и основан на следующем рассуждении. Пусть $\xi_{\text{opt}} \geqslant 0$ — (неизвестное) оптимальное значение порога. Тогда, если текущее значение порога, пусть $\xi_0 \geqslant 0$, является оптимальным, то решение, которое принимается в момент времени (пусть это будет момент 0), когда состояние системы находится на границе (которая естественно определяется пороговым значением), не должно влиять на значение целевой функции. Таким образом, если ввести величины $g_n^{(m)}$, $m = 1{,}2$, — стационарное среднее время пребывания задания в системе в n-й момент принятия решения, при условии, что в момент 0 было принято решение направить задание на сервер с номером m, то знак суммы

$$\sum_{n=0}^{\infty} \left(g_n^{(1)} - g_n^{(2)} \right)$$

будет говорить о том, насколько текущее значение порога ξ_0 находится близко к оптимальному значению ξ_{opt}. Если знак суммы больше нуля, то значение порога ξ_0 необходимо уменьшить, в противном случае увеличить. Таким образом, с помощью данной итерационной процедуры можно находить ξ_{opt} с

[79] Для систем, состоящих из более, чем двух серверов структура оптимальной стратегии неизвестна: она совершенно необязательно должна быть пороговой (см., например, [283]).

[80] Необходимо отметить, что этот оригинальный метод (в зарубежной литературе — First Policy Iteration) применим для нахождения хороших (и иногда практически неулучшаемых!) стратегий во многих задачах распределения ресурсов, когда есть возможность провести декомпозицию системы (как, например, при пуассоновском потоке заданий). См. подробнее, например, в [191; 285–287] и сноску на стр. 175.

заданной степенью точности. Для расчета значений $g_n^{(m)}$ приходится ограничивать множество состояний системы, которое в данном случае совпадает с $\mathbb{R}^2_+$, а также проводить его дискретизацию[81]. Экспериментально было установлено, что сумма разностей $g_n^{(1)} - g_n^{(2)}$ является очень чувствительной к тому, как вводится сетка на непрерывном множестве состояний. При этом удовлетворительные результаты удается достигнуть на неравномерной сетке специального (косоугольного) вида; см. соответствующие построения на стр. 165. Псевдокод итерационного алгоритма, реализующего изложенные идеи, дается на стр. 170, после чего на численных примерах демонстрируется его эффективность[82]. Здесь же описывается схема применения алгоритма для нахождения приближенных значений порогов в системе с произвольным числом серверов.

Несмотря на успех аналитического пути воплощения идеи диспетчеризации по предыстории, который был продолжен в параграфе 3.1, круг систем, для которых таким образом можно прийти к рабочему[83] алгоритму, нельзя считать широким. Во-первых, фактически неохваченными оказываются системы с серверами, использующими сложные и полезные дисциплины обслуживания (например, SRPT). Во-вторых, можно указать условия, в которых выбор очередного действия по трудоемкости выйдет за рамки всякого разумного представления о времени выполнения. Поэтому параграф 3.3 посвящен изложению более универсального, аналитико–имитационного подхода к задаче диспетчеризации, применимого в любой частично наблюдаемой системе с параллельным обслуживанием. Такое расширения области применения достигается путем замены ранее рассчитываемых значений величин, необходимых диспетчеру для выбора очередного действия, на их статистические оценки, получаемые посредством имитационной модели. В основу новых алгоритмов диспетчеризации по

[81] Другими словами, не удалось избежать аппроксимации исходной марковской цепи с непрерывным множеством состояний некоторой конечной цепью. Однако в отличие от уравнений Беллмана, которые теперь (после введения разбиения пространства состояний) также можно применить для получения ответа, предложенное решение использует специальный способ вычисления вероятностей перехода, не требующий использования матрицы переходных вероятностей.

[82] Предложенный прием не только свободен от недостатков имитационных методов, но и позволяет аналитически “подступиться” к задаче управления в случае непрерывного множества состояний. Но необходимо отметить, что в более общих постановках надежды на успехи, по-видимому, связаны с использованием метода Монте–Карло в сочетании с адаптивными алгоритмами для управления частично наблюдаемыми марковскими цепями (см. [178] и сноску на стр. 175).

[83] Возникает либо то, что принято называть “проклятием размерности”, либо необходимые величины вовсе не поддаются ни прямому, ни косвенному расчету.

предыстории положен прием, используемый в теории адаптивного управления и известный под названием идентификационный подход (см. стр. 177): задавшись какой-нибудь хорошей стратегией, идентифицируются (на компьютерной модели) необходимые для ее реализации, но недоступные для наблюдения, динамические характеристики серверов. Действенность аналитико–имитационного подхода показывается в параграфе 3.4 на тех же постановках, что были рассмотрены в параграфе 3.2. И результаты вычислительных экспериментов свидетельствуют о том, что, даже несмотря на замену вычислений по формулам вероятностной процедурой, новые алгоритмы диспетчеризации по предыстории позволяют уменьшить значения функционалов в сравнении со всеми ранее известными из научной литературы стратегиями (т. е. RND и PROG) почти во всем диапазоне изменений значений исходных параметров системы[84]. Кроме того, они являются достаточно чувствительными, чтобы подтвердить установленный в параграфе 3.1 контринтуитивный факт, характерный для диспетчеризаций по предыстории: при определенных значениях исходных параметров системы, стратегии, опирающиеся на (некоторые) наблюдения (например, JSQ) уступают новым стратегиям, вовсе не использующим информацию о текущем состоянии системы (далее см. обсуждение на стр. 185).

Заключительная глава диссертации (глава 4) посвящена изложению результатов поиска способов реализации идеи диспетчеризации по предыстории, во-первых, в еще более широком, чем в главе 3, классе систем с параллельным обслуживанием[85] и, во-вторых, свободных от вычислительных недостатков[86]. В основе нового подхода — идея использования для порождения действий виртуальных вспомогательных процессов, зависящих от неизвестных параметров и синхронизованных по моментам поступления заданий с основной системой. Ввиду того, что априорная информация дает возможность осуществлять имитацию траектории системы, значение неизвестных параметров может быть подобрано. Вычислительные эксперименты показывают, что новые алгоритмы (см. описание основной версии алгоритма на стр. 202) успешно конкурируют[87] со всеми ранее известными в научной литературе диспетчеризациями (т. е. RND

[84]Традиционно исключения составляют случаи загрузки, близкой к критической, где наблюдается совпадение результатов с наилучшей из ранее известных стратегий.

[85]Например, с ненадежными серверами.

[86]Другими словами, таких же простых, как и правила (5) и (8).

[87]Но уже не являются, в отличие от алгоритмов главы 3, наилучшими во всем диапазоне изменений значений исходных параметров систем.

и PROG), а в сбалансированных системах[88] часто и превосходят их. Эти свойства, вкупе с тем обстоятельством, что новые алгоритмы требуют для своей настройки оценки существенно меньшего числа параметров, дают основания назвать их лучшими для ненаблюдаемых систем с параллельным обслуживанием. Рассмотренные серии экспериментов охватывают различные варианты входного потока заданий, различные распределения длины заданий, разное число серверов, различные дисциплины обслуживания[89]. Качественная картина такова. При фиксированной дисциплине обслуживания в серверах наличие выигрыша от применения новой стратегии зависит, главным образом, от качества доступных оценок параметров наилучшей из ранее известных стратегий — стратегии PROG. Если нет возможности получить близкие к оптимальным значения или приходится исходить при их выборе из здравого смысла (например, балансируя нагрузку), то, как показывают вычислительные эксперименты, предложенный алгоритм следует признать равномерно наилучшим. В противном случае результат сравнения зависит от соотношений между[90] коэффициентом вариации C_B размера заданий, загрузкой системы ρ и ее размером M. Так, при фиксированном ρ, эффективность нового алгоритма снижается с увеличением числа серверов; при этом, начиная с некоторого M, относительный выигрыш стабилизируется. При фиксированном M соотношение между стратегиями зависит от случайности распределения размера заданий. При $C_B \ll 1$ равномерно наилучшей по ρ является новая стратегия. С увеличением C_B стабильного выигрыша удается добиться только в области малой загрузки (при $C_B = 1$ граница проходит, по-видимому, в районе средней загрузки). То, что новый алгоритм, осуществляющий диспетчеризацию по предыстории, не является лучшим во всем диапазоне загрузки является ожидаемым следствием его преимуществ — универсальности и простоты. Другой важной отличительной особенностью новых алгоритмов является возмож-

[88] По этому поводу см. сноску на стр. 203.

[89] Поскольку каждый сервер представляет собой однолинейную СМО с неограниченной очередью, то приведены примеры с двумя “крайними” случаями обобщенной дисциплины преимущественного обслуживания наикратчайшей заявки с прерыванием обслуживания: наиболее употребительная — FIFO и оптимальная с точки зрения минимизации среднего времени пребывания заявки в системе — SRPT. Отметим, что хотя первые результаты об оптимальности последней дисциплины появились более 50 лет назад (см. [288]), ее изучение остается предметом активных научных исследований [289–292].

[90] При рекуррентном входном потоке, его случайность т. е. значение коэффициента вариации C_F, судя по экспериментам, оказывает незначительное влияние.

ность естественным образом отражения в них структуры и функциональных особенностей системы. Наиболее показательным и важным для практики примером является ситуация с частичной доступностью серверов (см. обсуждение на стр. 212): при наличии точной информации о моментах выключения/включения серверов, новый алгоритм может быть наилучшим уже во всем диапазоне загрузки; в то же время для других известных стратегий такая информация является бесполезной. Оставшаяся часть главы 4 посвящена обсуждению одного вопроса, связанного с теоретической основой продуктивности столь простой в реализации, но интуитивно совершенно не очевидной конструкции. В заключении к диссертации перечисляются задачи, представляющие интерес при дальнейшей разработке темы, и кратко подводятся итоги выполенного исследования, из которых складываются **выносимые на защиту положения**:

- метод получения оценок значений стационарных вероятностно–временных характеристик изолированно функционирующих стохастических систем обслуживания на основе доступной информации о прогнозных временах обслуживания.
- доказательство эффективности предложенного метода для некоторых классов частично наблюдаемых стохастических систем обслуживания, моделируемых немарковскими системами массового обслуживания с пуассоновскими входящими потоками.
- методы диспетчеризации по полной предыстории в стохастических системах с параллельным обслуживанием в условиях, когда не наблюдаемы традиционно важные для решения задач оптимизации характеристики, включая показатель, подлежащий минимизации.
- алгоритмы квазиоптимальной диспетчеризации в частично наблюдаемых стохастических системах, составленных из параллельно работающих систем массового обслуживания с классическими дисциплинами обработки очередей.
- метод создания стратегий управления входящими потоками для некоторых классов стохастических систем с параллельным обслуживанием при отсутствии информации об их динамическом состоянии, основанный на использовании виртуальных вспомогательных процессов и экспериментальное обоснование его эффективности.

Основные результаты работы **докладывались** на Европейской конференции по математическому и имитационному моделированию (Олесунн, 2013 г.;

Регенсбург, 2016 г.; Вильгельмсхафен, 2018 г.; Вильдау, 2020 г.; Эль–Кувейт, 2021 г.), на Европейском симпозиуме по вопросам системной инженерии (Берлин, 2017 г.; Милан, 2019 г.), на Первой европейской конференции по теории массового обслуживания (Гент, 2014 г.), на Международном конгрессе по ультрасовременным телекоммуникациям и системам управления (Санкт-Петербург, 2010 г., 2012 г., 2014 г.; Лиссабон, 2016 г.), на Международной конференции по матрично-аналитическим методам в стохастических моделях (Будапешт, 2016 г.), на Международном семинаре по проблемам устойчивости стохастических моделей (Светлогорск, 2012 г.; Тампере, 2015 г.), на Международной конференции "Распределенные компьютерные и телекоммуникационные сети: управление, вычисление, связь" (Москва, 2016–2020 гг.), на Международной конференции по стохастическим методам (Геленджик, 2019 г.), на XVII и XVIII Международной конференции имени А.Ф. Терпугова "Информационные технологии и математическое моделирование" (Томск, 2018 г.; Саратов, 2019 г.), на II–IV школе молодых ученых ИПИ РАН (Москва, 2011–2013 гг.), на научном семинаре по теории массового обслуживания кафедры теории вероятностей механико–математического факультета МГУ им. М.В. Ломоносова под руководством проф. Л.Г. Афанасьевой.

Результаты диссертации опубликованы в работах[91] [174; 178; 293–317], используются в учебном процессе Российского университета дружбы народов на факультете физико–математических и естественных наук при преподавании

[91] В совместно опубликованных работах вклад автора состоит в следующем. В [293–297] автором предложены методы получения оценок фактических значений стационарных вероятностно–временных характеристик частично наблюдаемых систем; доказана их состоятельность и получены соответствующие условия. В [298; 299] автором развит аналитический аппарат решения задач стационарного анализа введенного нового класса систем массового обслуживания инверсионного типа. В [300–302] автор предложил методы порождения диспетчеризаций при полном отсутствии динамической информации о состоянии систем и получил экспериментальные обоснования их состоятельности. В [303–307] автором разработаны алгоритмы диспетчеризации для частично наблюдаемых систем с параллельным обслуживанием и классическими дисциплинами обработки очередей, и получено экспериментальное обоснование их эффективности. В [178; 308] автором предложены квазиградиентные алгоритмы определения оптимальных значений параметров дисцпетчеризаций по наблюдениям за фазовой траекторией. В [309] автору принадлежит подход к диспетчеризации по полной предыстории. В [174] автор описал основные отмеченные в мировой научной литературе и используемые на практике алгоритмы диспетчеризации, и их ключевые свойства. В [310; 311] автором получены основные аналитические результаты и на их основе разработаны алгоритмы расчета оптимальных значений порогов. В [312] автором предложена "имитационно–адаптивная" технология решения задач планирования ресурсов и схема ее реализации.

курсов «Имитационное моделирование», «Дискретные вероятностные модели», «Дискретные математические модели» и в Межведомственном Суперкомпьютерном Центре РАН при эксплуатации и развитии ряда суперкомпьютерных систем коллективного пользования.

Глава 1. Основные стационарные характеристики систем инверсионного типа с пуассоновским входящим потоком и некоторыми неконсервативными дисциплинами обслуживания

1.1 Дисциплина обобщенного вероятностного приоритета

Рассмотрим систему массового обслуживания с одним обслуживающим прибором и $(n-1)$-м местом ожидания ($n \leqslant \infty$), на вход которой поступает пуассоновский поток заявок с переменным параметром λ_k, зависящим от числа заявок k, находящихся в системе. Каждой приходящей заявки соответствует некоторая случайная величина, которую будем трактовать как исходное время обслуживания и назовем исходной длиной заявки, подразумевая, что длина измеряется в единицах времени. Будем считать, что обслуживающий прибор, работающий с единичной скоростью, не портится и способен немедленно после окончания обслуживания одной заявки приступить к обслуживанию следующей. Прибор одновременно может обслуживать только одну заявку. Кроме того, допустим, что прерывание обслуживания, смена заявки на приборе и удаление заявок из системы происходит мгновенно.

В случае, когда в системе нет свободных мест ожидания, каждая приходящая заявка теряется. В противном случае, определим дисциплину обслуживания следующим образом. Вне зависимости от предыстории функционирования системы в момент очередного поступления исходная длина u новой заявки сравнивается с (остаточной) длиной v заявки на приборе. С вероятностью $D(x,y|u,v)$, зависящей только от u и v, обслуживавшаяся ранее заявка продолжает обслуживаться, причем ее длина становится меньше y, а вновь поступившая становится на первое место в очереди и ее длина становится меньше x. Кроме того, с вероятностью $D^*(x,y|u,v)$, зависящей только от u и v, вновь поступившая заявка занимает прибор, вытесняя обслуживавшуюся ранее на первое место в очереди, причем длина заявки, бывшей ранее на приборе, становится меньше y, а вновь поступившей — меньше x.

Если на приборе находится заявка остаточной длины v и в систему поступает заявка длины u, то с вероятностью $D_0(x|u,v)$ заявка, находящаяся на приборе, покидает систему, а поступившая заявка становится на прибор, причем

ее длина становится меньше x. Кроме того, с вероятностью $D_0^*(y|u,v)$ поступившая заявка сразу же покидает систему, а заявка, находящаяся на приборе, продолжает обслуживаться, причем ее длина становится меньше y. Введем также обозначение

$$D(x|u,v) = D_0(x|u,v) + D_0^*(x|u,v).$$

Здесь $D(x|u,v)$ — вероятность того, что одна из двух заявок покинет систему, а вторая встанет на прибор и примет длину меньше x. Наконец, предполагается, что с вероятностью $d_0(u,v)$ обе заявки покидают систему, а на прибор становится первая заявка из очереди. Если длина заявки на приборе становится равной нулю, то она мгновенно покидает систему и на прибор переходит первая заявка из очереди. Остальная очередь сдвигается на единицу.

Будем считать, что все ф. р. $D(x,y|u,v)$, $D^*(x,y|u,v)$, $D_0(x|u,v)$, $D_0^*(y|u,v)$, $D(y|u,v)$ и $D_0(u,v)$ имеют непрерывные ограниченные плотности

$$\begin{aligned} d(x,y|u,v) &= \frac{\partial^2 D(x,y|u,v)}{\partial x\,\partial y},\ d^*(x,y|u,v) = \frac{\partial^2 D^*(x,y|u,v)}{\partial x\,\partial y}, \\ d_0(x|u,v) &= \frac{\partial D_0(x|u,v)}{\partial x},\ d_0^*(y|u,v) = \frac{\partial D_0^*(y|u,v)}{\partial y}, \\ d(x|u,v) &= \frac{\partial D(x|u,v)}{\partial x}. \end{aligned}$$

Естественно, для любых u и v выполнено условие

$$\int_0^\infty \int_0^\infty (d(x,y|u,v) + d^*(x,y|u,v))\,dx\,dy + \int_0^\infty d(x|u,v)\,dx + d_0(u,v) = \\ = D(\infty,\infty|u,v) + D^*(\infty,\infty|u,v) + D(\infty|u,v) + d_0(u,v) = 1. \quad (1.1)$$

Поскольку описанная дисциплина обслуживания обобщает правило обработки очереди, изученное в [318; 319], будем называть ее инверсионным порядком обслуживания с обобщенным вероятностным приоритетом[1] (далее — LIFO GPP, Last–In–First–Out with Generalized Probabilistic Priority).

[1] Частным случаем дисциплины LIFO GPP являются известные правила обработки очереди: прямой порядок обслуживания (с точки зрения стационарного распределения очереди), инверсионный порядок обслуживания с абсолютным и относительным приоритетами [320], инвариантная дисциплина обслуживания [98].

Длины заявок являются независимыми одинаково распределенными сл. в. с произвольной ф. р. $B(x)$ и средним значением $\int_0^\infty x\,dB(x) = \mathsf{E}S$. Всюду далее предполагается, что существует[2] непрерывная ограниченная плотность $b(x) = B'(x)$. Поскольку в пределах всей главы это нигде не вызовет недоразумений, будем называть длиной и остаточную длину заявки.

Условимся кодировать описанную систему как $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{LIFO\,GPP}$, где обозначение M_k указывает на тот факт, что параметр входящего пуассоновского потока зависит от числа k заявок в системе.

Введем n-мерный случайный процесс $\boldsymbol{\eta}(t)$, описывающий функционирование системы, как вектор длин заявок, находящихся в системе в момент t и расположенных в порядке, обратном очереди, т. е. если в момент t в системе находится $\boldsymbol{\nu}(t)$ заявок, то $\xi_1(t)$ — длина заявки, находящейся на последнем месте в очереди, $\xi_2(t)$ — длина заявки на предпоследнем месте в очереди, …, $\xi_{\boldsymbol{\nu}(t)}(t)$ — длина обслуживаемой заявки, $\xi_{\boldsymbol{\nu}(t)+1}(t) = \cdots = \xi_n(t) = 0$.

Положим

$$\begin{aligned} P_0(t) &= \mathsf{P}\{\boldsymbol{\nu}(t) = 0\}, \\ P_k(t; x_1, \ldots, x_k) &= \mathsf{P}\{\boldsymbol{\nu}(t) = k, \xi_k(t) < x_1, \ldots, \xi_1(t) < x_k\}, \quad 1 \leqslant k \leqslant n-1, \end{aligned}$$

и введем совместные и маргинальные стационарные распределения процесса $\boldsymbol{\eta}(t)$:

$$\begin{aligned} P_k(x_1, \ldots, x_k) &= \lim_{t\to\infty} P_k(t; x_1, \ldots, x_k), \\ P_k(x) &= P_k(x, \infty, \ldots, \infty), \quad P_k = P_k(\infty), \\ p_k(x_1, \ldots, x_k) &= \frac{\partial^k}{\partial x_1 \cdots \partial x_k} P_k(x_1, \ldots, x_k), \\ p_k(x) &= P_k'(x), \\ Q_n(t; x_1, \ldots, x_n) &= \mathsf{P}\{\boldsymbol{\nu}(t) = n, \xi_n(t) < x_1, \ldots, \xi_1(t) < x_n\}, \\ Q_n(x_1, \ldots, x_n) &= \lim_{t\to\infty} Q_n(t; x_1, \ldots, x_n), \\ Q_n(x) &= Q_n(x, \infty, \ldots, \infty), \quad Q_n = P_n(\infty), \\ q_n(x_1, \ldots, x_n) &= \frac{\partial^n}{\partial x_1 \cdots \partial x_n} Q_n(x_1, \ldots, x_n), \\ q_n(x) &= Q_n'(x). \end{aligned}$$

При $n = \infty$ величины $Q_n(t; x_1, \ldots, x_n)$, $Q_n(x_1, \ldots, x_n)$ и $q_n(x_1, \ldots, x_n)$ не определяются. Относительно плотностей $p_k(x_1, \ldots, x_k)$, $q_n(x_1, \ldots, x_n)$, $p_k(x)$ и $q_n(x)$

[2] Для доказательства некоторых утверждений от функции b потребуется большее, но каждый раз это будет оговорено особо.

будем предполагать[3], что они существуют, являются ограниченными и непрерывными.

Если не накладывать никаких дополнительных ограничений на функции d, d^*, d_0 и d_0^* дисциплина LIFO GPP не является консервативной[4]. Поэтому произвольных d, d^*, d_0 и d_0^* не удается выписать общее необходимое и достаточное условие существования стационарного режима[5]: оно зависит от конкретных параметров системы и в каждом отдельном случае нуждается в специальном исследовании. Однако из сравнения суммарной имеющейся работы в рассматриваемой системе и суммарной работы в стандартной СМО $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{FIFO}$ можно получить достаточное условие. Оно заключается в выполнении следующих соотношений[6]:

1. $\mathsf{E}S < \infty$;
2. $d(x,y|u,v) = 0$ при всех u,v и $y > v$ или $x > u$;
3. $d^*(x,y|u,v) = 0$ при всех u,v и $y > v$ или $x > u$;
4. $d(x|u,v) = 0$ при всех u,v и $x > u$;
5. $d^*(y|u,v) = 0$ при всех u,v и $y > v$,

и $\lim_{n\to\infty} \lambda_n \mathsf{E}S < 1$ при $n = \infty$.

[3]Доказательство этого для произвольных (но удовлетворяющих ряду свойств) консервативных дисциплин обслуживания основывается на известных методах (теории регенерирующих процессов и теории восстановления) [321;322]. При дисциплине LIFO GPP моменты прихода заявок в свободную систему, по-прежнему, являются точками регенерации. Однако времена обслуживания заявок уже зависят от входящего потока. Поэтому, по крайней мере напрямую, доказательство на основе известной теории не проходит. Это же обстоятельство затрудняет применение известных специальных методов [323;324] нахождения условий стационарности.

[4]Напомним (см., например, [75, С. 49]), что свойство консервативности дисциплины означает, что длительность обслуживания заявки не зависит от дисциплины обслуживания и нет искусственных простоев прибора.

[5]Забегая вперед, приведем пример. Пусть $n < \infty$. Положим $d(x,y|u,v) = e^{-v}b(x)b(ye^{-v})$, $d^*(x,y|u,v) = d(x|u,v) = d_0(u,v) = 0$, $u,v \geqslant 0$. Воспользовавшись рассуждениями, приведенными в начале доказательства *Теоремы 2*, сталкиваемся со следующей ситуацией: среднее время до того момента, когда в системе останется $(n-2)$ заявки, при условии, что в начальный момент в системе было $(n-1)$ заявок (без дополнительных ограничений на плотность $b(x)$) может быть равно бесконечности. При этом, учитывая пуассоновость входящего потока, с ненулевой вероятностью система переходит в состояние $(n-2)$ и, вообще говоря, с ненулевой вероятностью может успеть выполнить до прихода очередной заявки любой конечный объем находящейся в ней работы, т. е. полностью опустошиться.

[6]Соотношения 2–5 соответствуют тому факту, что после поступления новой заявки измененные длины заявок не превышают те длины, которые были до поступления. Отметим также, что параметр $\lim_{n\to\infty} \lambda_n \mathsf{E}S$ не является загрузкой системы в традиционном смысле и может существенно от нее отличаться.

Теорема 1. *Для СМО* $M_k\,|\,GI\,|\,1\,|\,n\,|\,\mathsf{LIFO\,GPP}$ *($n \leqslant \infty$) стационарные вероятности состояний определяются рекуррентно из следующей системы уравнений:*

$$-p_1'(x) = \lambda_0 b(x) P_0 - \lambda_1 p_1(x) + \lambda_1 \Bigg(\int_0^\infty \int_0^\infty d(x|u,v) b(u) p_1(v)\, du\, dv + \int_0^\infty \int_0^\infty \int_0^\infty (d(x,y|u,v) + d^*(y,x|u,v)) b(u) p_1(v)\, dy\, du\, dv \Bigg), \quad (1.2)$$

$$-p_k'(x_1,\ldots,x_k) = -\lambda_k p_k(x_1,\ldots,x_k) + \lambda_{k-1} \Bigg(\int_0^\infty \int_0^\infty (d(x_2,x_1|u,v) + d^*(x_1,x_2|u,v)) b(u) p_{k-1}(v,x_3\ldots,x_k)\, du\, dv \Bigg) + \lambda_k \Bigg(\int_0^\infty \int_0^\infty d(x_1|u,v) b(u) p_k(v,x_2,\ldots,x_k)\, du\, dv + \int_0^\infty \int_0^\infty \int_0^\infty (d(x_1,y|u,v) + d^*(y,x_1|u,v)) b(u) p_k(v,x_2,\ldots,x_k)\, dy\, du\, dv \Bigg), \quad (1.3)$$

при $1 \leqslant k \leqslant n-1$, *и*

$$-q_n'(x_1,\ldots,x_n) = \lambda_{n-1} \Bigg(\int_0^\infty \int_0^\infty (d(x_2,x_1|u,v) b(u) q_{n-1}(v,x_3,\ldots,x_n) + d^*(x_1,x_2|u,v) b(u) q_{n-1}(v,x_3,\ldots,x_n))\, du\, dv \Bigg), \quad (1.4)$$

с граничными условиями

$$p_1(\infty) = 0,\ p_k(\infty,x_2,\ldots,x_k) = 0,\ 1 \leqslant k \leqslant n-1,\ q_n(\infty,x_2,\ldots,x_n) = 0. \quad (1.5)$$

Постоянная P_0 *определяется из условия нормировки* $\sum_{k=0}^{n-1} P_k + Q_n = 1$.

Доказательство. То обстоятельство, что рекуррентное вычисление стационарного распределения исследуемой системы возможно, следует из (развитой ранее в ряде работ других авторов [59;98–109]) теории систем со специальными дисциплинами обслуживания, которые объединяет следующее свойство, наследуемое также и дисциплиной $\mathsf{LIFO\,GPP}$.

Пусть $m > 0$ — произвольное целое число. Рассмотрим систему $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{LIFO\,GPP}$ при любом $m < n+1$. Выделим для процесса $\boldsymbol{\eta}(t)$ те интервалы времени, когда число заявок в системе будет больше m. Тогда, несмотря на наличие функций d, d^*, d_0 и d_0^*, с того момента, как в системе впервые появится $(m+1)$-ая заявка и до того момента, как в системе снова будет m заявок, m компонент $\boldsymbol{\xi}_1(t)$, $\boldsymbol{\xi}_2(t)$, …, $\boldsymbol{\xi}_m(t)$ процесса $\boldsymbol{\eta}(t)$ не меняются. Кроме того, пока $\boldsymbol{\nu}(t) \leqslant m$, процессы $\boldsymbol{\eta}(t)$ для $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{LIFO\,GPP}$ при любом $m < n+1$ будут идентичны. Таким образом, если для процесса $\boldsymbol{\eta}(t)$ исключить все те интервалы времени, когда $\boldsymbol{\nu}(t) > m$, и оставшиеся куски склеить, то вероятностные характеристики получившегося после склейки процесса будут одинаковыми для всех $m < n+1$. Отсюда делается вывод: для всех систем $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{LIFO\,GPP}$ при $n > m-1$ стационарные распределения $P_k(x_1, \ldots, x_k)$, $0 \leqslant k \leqslant n$, совпадают с точностью до постоянного множителя, не зависящего от k. В качестве этого множителя удобно выбрать P_0.

Выпишем уравнения, которым удовлетворяют стационарные плотности $p_k(x_1, \ldots, x_k)$. Рассмотрим моменты времени t и $t+\Delta$. Для того чтобы в момент времени $t+\Delta$ в системе находилось k, $2 \leqslant k \leqslant n-1$, заявок, причём на приборе заявка длины x_1, а в очереди заявки длин $x_2, \ldots, x_k$, нужно, чтобы произошло одно из следующих событий:

- в момент t в системе находилось $(k-1)$ заявок, причём заявка на приборе имела длину v, первая заявка в очереди имела длину $x_3, \ldots$, последняя заявка в очереди имела длину x_k (с плотностью вероятностей $p_{k-1}(t; v, x_3, \ldots, x_k)$), и за время Δ поступила заявка (с вероятностью $\lambda_{k-1}\Delta$) длины u (с плотностью вероятностей $b(u)$). Поступившая заявка продолжает обслуживаться, но её длина становится равной x_1, а вновь поступившая заявка занимает первое место в очереди и её длина становится равной x_2 (с плотностью вероятностей $d(x_2, x_1 | u, v)$);
- в момент t в системе находилось $(k-1)$ заявок, причём заявка на приборе имела длину v, первая заявка в очереди имела длину $x_3, \ldots$, последняя заявка в очереди имела длину x_k (с плотностью вероятностей $p_{k-1}(t; v, x_3, \ldots, x_k)$), и за время Δ поступила заявка (с вероятностью $\lambda_{k-1}\Delta$) длины u (с плотностью вероятностей $b(u)$). Поступившая заявка занимает прибор и её длина становится равной x_1, а заявка, обслуживавшаяся до поступления новой заявки, занимает первое место в

очереди и её длина становится равной x_2 (с плотностью вероятностей $d^*(x_1,x_2|u,v)$);

– в момент t в системе находилось k заявок, причём заявка на приборе имела длину $x_1 + \Delta$, первая заявка в очереди имела длину $x_2, \ldots$, последняя заявка в очереди имела длину x_k (с плотностью вероятностей $p_k(t; x_1 + \Delta,x_2, \ldots,x_n)$), и за время Δ не поступили заявки (с вероятностью $(1 - \lambda_k\Delta)$);

– в момент t в системе находилось k заявок, причём заявка на приборе имела длину v, первая заявка в очереди имела длину $x_2, \ldots$, последняя заявка в очереди имела длину x_k (с плотностью вероятностей $p_k(t; v, \ldots,x_k)$), и за время Δ поступила заявка (с вероятностью $\lambda_k\Delta$), имеющая длину u (с плотностью вероятностей $b(u)$). Заявка, находящаяся на приборе, покидает систему, а поступившая заявка становится на прибор, причём её длина становится равной x_1, или, наоборот, поступившая заявка сразу же покидает систему, а заявка, находящаяся на приборе продолжает обслуживаться, причём её длина становится равной x_1 (с плотностью вероятностей $d(x_1|u,v)$);

– в момент t в системе находилось k заявок, причём заявка на приборе имела длину v, первая заявка в очереди имела длину $x_2, \ldots$, последняя заявка в очереди имела длину x_n (с плотностью вероятностей $p_k(t; v,x_2, \ldots,x_k)$), и за время Δ поступила заявка (с вероятностью $\lambda_k\Delta$) длины u (с плотностью вероятностей $b(u)$). Заявка, находившаяся на приборе, покинула систему, а на прибор встала поступившая заявка, длина которой стала x_1 (с плотностью вероятностей $d(x_1,y|u,v)$);

– в момент t в системе находилось k заявок, причём заявка на приборе имела длину v, первая заявка в очереди имела длину $x_2, \ldots$, последняя заявка в очереди имела длину x_k (с плотностью вероятностей $p_k(t; v,x_2, \ldots,x_k)$), и за время Δ поступила заявка (с вероятностью $\lambda_k\Delta$) длины u (с плотностью вероятностей $b(u)$). Вновь поступившая заявка покидает систему, на приборе продолжает обслуживаться заявка, находящаяся на приборе до поступления новой заявки, но длина её становится равной x_1 (с плотностью вероятностей $d^*(y,x_1|u,v)$).

Вероятности других событий равны $o(\Delta)$. Применяя формулу полной вероятности, имеем

$$p_k(t+\Delta; x_1,\ldots,x_k) = \lambda_{k-1}\Delta\Bigg(\int_0^\infty\int_0^\infty (d(x_2,x_1|u,v)b(u)p_{k-1}(t; v,x_3\ldots,x_k)+$$
$$+ d^*(x_1,x_2|u,v)b(u)p_{k-1}(t; v,x_3,\ldots,x_k))\,du\,dv\Bigg)+$$
$$+ (1-\lambda_k\Delta)p_k(t; x_1+\Delta,x_2,\ldots,x_k)+$$
$$+\lambda_k\Delta\Bigg(\int_0^\infty\int_0^\infty d(x_1|u,v)b(u)p_k(t; v,x_2,\ldots,x_k)\,du\,dv+$$
$$+\int_0^\infty\int_0^\infty\int_0^\infty (d(x_1,y|u,v)b(u)p_k(t; v,x_2,\ldots,x_k)+$$
$$+ d^*(y,x_1|u,v)b(u)p_k(t; v,x_2,\ldots,x_k))\,dy\,du\,dv\Bigg) + o(\Delta),\quad k\geqslant 2,$$

откуда, перенося слагаемое $p_k(t; x_1+\Delta,x_2,\ldots,x_k)$ в левую часть равенства, деля на Δ, устремляя Δ к нулю и учитывая стационарный режим функционирования системы, получаем (1.3). Уравнения (1.2) для $p_1(x)$ и (если имеется в виду случай $n<\infty$) уравнение (1.4) для $q_n(x_1,\ldots,x_k)$ получаются аналогичным образом.

Интегрируя равенство (1.2) в пределах от 0 до некоторого $a>0$, получаем

$$p_1(a) = p_1(0) - \lambda_0 B(a)p_0 + \lambda_1 P_1(a) -$$
$$-\lambda_1\int_0^\infty\int_0^\infty b(u)p_1(v)\int_0^a\left(d(x|u,v)+\int_0^\infty (d(x,y|u,v)+d^*(y,x|u,v))\,dy\right)du\,dvdx.$$

С учётом (1.1) правая часть имеет конечный предел при $a\to\infty$. Поэтому $p_1(a)$ также стремится при $a\to\infty$ к пределу, который для плотности вероятностей не может быть ни чем иным, кроме нуля, что доказывает справедливость первого равенства в (1.5). Остальные равенства получаются аналогичным образом.

□

Если для функций d, d^*, d_0 и d_0^* известны сепарабельные аппроксимации[7]

$$d(x,y|u,v) = \sum_{i=1}^{n_1} \tilde{\alpha}_{1i}(x)\tilde{\beta}_{1i}(y)\tilde{\gamma}_{1i}(u)\tilde{\delta}_{1i}(v),$$
$$d^*(x,y|u,v) = \sum_{i=1}^{n_2} \tilde{\alpha}_{2i}(x)\tilde{\beta}_{2i}(y)\tilde{\gamma}_{2i}(u)\tilde{\delta}_{2i}(v),$$
$$d_0(x|u,v) = \sum_{i=1}^{n_3} \tilde{\alpha}_{3i}(x)\tilde{\gamma}_{3i}(u)\tilde{\delta}_{3i}(v),$$
$$d_0^*(x|u,v) = \sum_{i=1}^{n_4} \tilde{\alpha}_{4i}(x)\tilde{\gamma}_{4i}(u)\tilde{\delta}_{4i}(v),$$

где n_1, n_2, n_3, n_4 — некоторые натуральные числа, а функции $\tilde{\alpha}_{ij}(x)$, $\tilde{\beta}_{ij}(x)$, $\tilde{\gamma}_{ij}(x)$, $\tilde{\delta}_{ij}(x)$ являются неотрицательными и таковыми, что выполнено условие нормировки (1.1), то уравнения (1.2)—(1.4) можно упростить (см. [338, параграф 1.2.3]).

Система (1.2)—(1.4) решается рекуррентным образом. Сначала определяется $p_1(x)$, затем через $p_{k-1}(x_1,\ldots,x_{k-1})$ вычисляется значение $p_k(x_1,\ldots,x_k)$. Наконец, в случае $n < \infty$, $q_n(x_1,\ldots,x_k)$ определяется из последнего уравнения системы. Необходимо отметить, что практическое применение такого подхода невозможно, поскольку при $n \to \infty$ число аргументов x_i стационарных плотностей вероятностей $p_k(x_1,\ldots,x_k)$ стремится к бесконечности. Поэтому методы решения необходимо искать каждый раз, отталкиваясь от конкретных параметров системы. В [299] подробно исследован случай нахождения маргинальных плотностей вероятностей $p_k(x)$ при $n = \infty$ и $\lambda_k = \lambda$. В частности, показано, после интегрирования (1.2)—(1.3) по x_2,...,x_k в пределах от нуля до бесконечности, заменой $p_k(x) = e^{-\lambda x}q_k(x)$ получаются интегральные уравнения Фредгольма 2-го рода для $q_k(x)$ со свободным членом и ядром — неотрицательными функциями. Их решения могут быть найдены известными методами [339–342]. Например, хорошие результаты дает итерационный метод, когда в качестве начальной итерации берется нулевое приближение $q_k(x) \equiv 0$; при этом итерации

[7] Из научных работ, в которых освещаются вопросы подобной аппроксимации, можно отметить [325–334]. В практических расчетах зачастую приходится иметь дело с функциями $d(x,y|u,v)$ и др., определенными в некоторой ограниченной замкнутой области. Для функции двух переменных удобны разложения по многочленам Чебышева. Соответствующая процедура описана, например, в [335]. В случае трех переменных можно воспользоваться результатами работы [336]. Процедура для функций трех и четырех переменных описана в [337].

будут возрастающими, что позволяет контролировать сходимость к точному решению. В тех же предположениях для численного нахождения, например, моментов стационарного распределения общего числа заявок в системе можно использовать и метод производящих функций.

Предполагая[8], что решения каждого из уравнений (1.2)—(1.4) единственно в классе ограниченных неотрицательных суммируемых функций, полученные в *Теореме 1* соотношения позволяют последовательно по k найти стационарное распределение системы, а через него и основные вероятностные характеристики [299]; например:

- вероятность того, что заявка не будет потеряна при поступлении, (не) будет обслужена до конца и за время пребывания в системе сменит длину $i \geqslant 0$ раз, при условии, что ее исходная длина равнялась x;
- распределение отношения времени пребывания заявки в системе к ее длине или времени обслуживания[9];
- вероятность того, что заявка будет потеряна при поступлении, при условии, что ее исходная длина равна x.

Заметим, что, поскольку входящий поток — пуассоновский, значения последней характеристики могут быть рассчитаны по формуле

$$\pi(x) = \frac{\sum\limits_{k=1}^{n-1} \lambda_k \int\limits_0^\infty \left(d_0(x,y) + \int\limits_0^\infty d_0^*(w|x,y)dw \right) dP_k(y) + \lambda_n Q_n}{\sum\limits_{k=0}^{n-1} \lambda_k P_k + \lambda_n Q_n}.$$

Широкие возможности выбора функций d, d^*, d_0 и d_0^* приводят большому разнообразию временных характеристик у поступающих в систему заявок, полный перечень которых составить не представляется возможным. Следующая теорема указывает путь получения формул для расчета (в терминах преобразований), по крайней мере, наиболее употребительных из них.

[8] Доказательство для одного частного случая дисциплины LIFO GPP дано, например, в [321, С. 59].

[9] Если $V(x)$ — время пребывания в системе заявки длины x, то $V(x)/x$ показывает во сколько раз время пребывания заявки в системе отличается от ее исходной длины. Поскольку вполне естественно считать, что более длинные заявки должны находиться в системе дольше, чем короткие, моменты случайной величины $V(X)/X$ используются в задачах оценки справедливости дисциплин обслуживания. В зарубежной литературе отношение $V(X)/X$ называется slowdown [343–345]; в отечественной литературе общепринятый термин пока не выкристаллизовался (см., однако, п. 4 в [75, С. 88] и [205, С. 199]).

Теорема 2. *Для СМО* $M_k \,|\, GI \,|\, 1 \,|\, n \,|\, \mathsf{LIFO\,GPP}$ *(*$n < \infty$*) ПЛС* $\varphi(s;x)$ *стационарного распределения времени пребывания в системе заявки исходной длины* x*, которая была обслужена до конца, равно*

$$\varphi(s;x) = \left(\sum_{k=0}^{n-1} \lambda_k P_k + \lambda_n Q_n\right)^{-1} \Bigg(\lambda_0 P_0 \psi_1(s;x) + \\ + \sum_{k=1}^{n-1} \lambda_k \int_0^\infty \int_0^\infty (d_0(v|x,y) + \int_0^\infty d^*(v,w|x,y)\,dw)\psi_{k+1}(s;v)\,dv dP_k(y) + \\ + \sum_{k=1}^{n-1} \lambda_k \int_0^\infty \int_0^\infty \int_0^\infty \psi_k(s;v) u_{k+1}(s;w) d(v,w|x,y)\,dv\,dw\,dP_k(y)\Bigg), \quad (1.6)$$

где $\psi_k(s;x)$ *— ПЛС распределения времени обслуживания (с учетом прерываний) заявки длины* x*, при условии, что в момент ее поступления на прибор в системе находится* k *заявок, а* $u_k(s;x)$ *— ПЛС распределения времени до момента, когда в системе впервые окажется* $(k-1)$ *заявок, при условии, что на приборе начала обслуживаться заявка длины* x *и в системе находится* k *заявок.*

Доказательство. Обозначим через $u_k(s;x)$, $1 \leqslant k \leqslant n$, преобразование Лапласа–Стилтьеса распределения времени до момента, когда в системе впервые окажется $(k-1)$ заявок, при условии, что на приборе начала обслуживаться заявка длины x и в системе находится k заявок. Поскольку заявка, поступающая в заполненную систему, теряется и не оказывает на нее воздействия, то $u_n(s;x) = e^{-sx}$. При $1 \leqslant k \leqslant n-1$ соотношение между $u_k(s;x)$ получается по свойству ПЛС и формуле полной вероятности:

$$u_k(s;x) = e^{-(\lambda_k+s)x} + \int_0^x \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} d_0(y,x-t)dt\,dB(y) + \\ + \int_0^x \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} dt u_k(s;v) d(v|y,x-t)dv\,dt\,dB(y) + \\ + \int_0^x \int_0^\infty \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} u_{k+1}(s;w) u_k(s;v) d(v,w|y,x-t)dw\,dv\,dt\,dB(y) + \\ + \int_0^x \int_0^\infty \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} u_{k+1}(s;v) u_k(s;w) d^*(v,w|y,x-t)dv\,dw\,dt\,dB(y). \quad (1.7)$$

Учитывая, что $u_k(s;x) \leqslant 1$ при любом $1 \leqslant k \leqslant n$, по принципу сжимающих отображений каждое из уравнений (1.7) имеет единственное решение [319, С. 93]. Решение системы может быть найдено рекуррентным образом: поставив $u_{k+1}(s;x)$ в (1.7), определяем $u_k(s;x)$, $k = n-1,n-2,\ldots,1$.

Вычислим теперь $\psi_k(s;x)$ — ПЛС распределения полного (т. е. включающего все времена, на которые обслуживание было прервано) времени обслуживания заявки длины x, при условии, что в момент ее поступления на прибор в системе находится k заявок. Т. к. поступающая в заполненную систему заявка теряется, $\psi_n(s;x) = e^{-sx}$. Уравнения для $\psi_k(s;x)$ при $1 \leqslant k \leqslant n-1$ получаются путем следующих рассудений. За время x с вероятностью $e^{-\lambda_k x}$ не поступит ни одной заявки, а с вероятностью $\int_0^x \int_0^\infty \lambda_k e^{-\lambda_k t} dB(y)dt$ на интервале $[t,t+dt]$ поступит заявка длины $[y,y+dy]$. Если осуществляется последнее, то возвожны три случая: либо с плотностью вероятности $d_0^*(w|y,x-t)$ новая заявка изменит длину заявки на приборе на w, а сама покинет систему; либо с плотностью вероятности $d(v,w|y,x-t)$ новая заявка встанет на первое место в очереди с новой длиной v, а заявка на приборе получит новую длину w; либо с плотностью вероятности $d^*(v,w|y,x-t)$ новая заявка встанет на прибор, получив новую длину v, а заявка с прибора будет вытеснена на первое место в очереди с новой длиной w. В терминах ПЛС получаем:

$$\psi_k(s;x) = e^{-(\lambda_k+s)x} + \int_0^x \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} \psi_k(s;w) d_0^*(w|y,x-t) d\,dw\,dt\,dB(y) +$$
$$+ \int_0^x \int_0^\infty \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} \psi_{k+1}(s;w) d(v,w|y,x-t)\,dv\,dw\,dt\,dB(y) +$$
$$+ \int_0^x \int_0^\infty \int_0^\infty \int_0^\infty \lambda_k e^{-(\lambda_k+s)t} \psi_k(s;w) u_{k+1}(s;v) d^*(v,w|y,x-t)\,dv\,dw\,dt\,dB(y). \quad (1.8)$$

По поводу решения этой системы уравнений справедливо то же само замечание, что было дано к системе (1.7).

Для заершения доказательства осталось заметить следующее. С вероятностью P_0 время пребывания в системе поступающей заявки совпадает с временем ее пребывания на приборе (с учетом прерываний). Но, если поступающая заявка длины x застает в системе $1 \leqslant k \leqslant n-1$ заявок, причем на приборе — заявку длины y (с плотностью вероятности $p_k(y)$), то возможны два варианта:

– либо с плотностью вероятности $d_0(v|x,y) + d^*(v,w|x,y)$ поступающая заявка встанет на прибор, причем ее длина станет равной v и тогда время ее пребывания в системе будет совпадать с временем ее пребывания на приборе (с учетом прерываний);
– либо с плотностью вероятности $d(v,w|x,y)$ поступающая заявка станет на первое место в очереди, получит новую длину v, а заявка на приборе – новую длину w; при этом время пребывания в системе поступившей заявки будет равно сумме двух времен: времени до того момента, когда в системе снова станет k заявок, и времени пребывания на приборе (с учетом прерываний) заявки длины v.

Применяя формулу полной вероятности, получаем (1.6).

□

Очевидно, ПЛС $\varphi(s)$ безусловного стационарного распределения времени пребывания в системе заявки, которая была обслужена до конца, получается путем усреднения $\varphi(s;x)$ по распределению длины заявки т. е. $\varphi(s) = \int_0^\infty \varphi(s;x)dB(x)$. Обратить в явном виде ПЛС, речь о которых идет в *Теореме 2*, нет никакой возможности. Перейти от изображений к оригиналам можно только численно, для чего можно воспользоваться известными, хорошо разработанными методами[10] (см., например, [348]).

В заключение этого параграфа сделаем несколько замечаний.

1. Исследуемую систему можно было бы незначительно обобщить на случай, когда распределение длины заявки, поступающей в пустую систему, отлично от $B(x)$.

2. Результаты *Теоремы 2* справедливы и в случае $n = \infty$. Однако здесь придется столкнуться с системами с бесконечным числом уравнений, поиск решения которых при произвольных функциях d, d^*, d_0 и d_0^* является бесперспективным.

3. Рассуждения в доказательстве *Теоремы 2* позволяют находить в терминах преобразований и другие временные характеристики. Например, условное стационарное распределение времени ожидания начала обслуживания заявки

[10] Отметим здесь недавно разработанный СМЕ–метод (см. [346]) обращения ПЛ, в основе которого — матрично–экспоненциальные распределения (см. [347, Раздел 3]) с маленьким коэффициентом вариации.

исходной длины x, застающей при поступлении $1 \leqslant k \leqslant n-1$ заявок в системе, имеет ПЛС

$$\int_0^\infty \int_0^\infty \left(d_0(v|x,y) + \int_0^\infty (d^*(v,w|x,y) + u_{k+1}(s;w) d(v,w|x,y))\, dw \right) dv dP_k(y),$$

поскольку поступающая заявка длины x, застающая в системе k заявок, с вероятностью $\int_0^\infty \int_0^\infty (d_0(v|x,y) + \int_0^\infty d^*(v,w|x,y)\, dw)\, dv dP_k(y)$ сразу же поступает на обслуживание, а с вероятностью $\int_0^\infty \int_0^\infty \int_0^\infty d(v,w|x,y)\, dw)\, dv dP_k(y)$ оказывается на первом месте в очереди. Распределение периода занятости в терминах ПЛС есть $\int_0^\infty u_1(s;x) dB(x)$. Заметим, что для этого результата не требуется эргодичность системы; можно также отказаться от конечности средней длины заявки и тогда длина периода занятости может принимать бесконечное значение с ненулевой вероятностью.

4. Все вышеизложенные результаты будут справедливы в случае произвольного распределения $B(x)$, не обязательно с плотностью, если под производной $B'(x)$ понимать обобщенную функцию. Поскольку $B'(x)$ входит линейно под интегралом, то вообще никаких затруднений при переписывании формул не возникает. Также можно получить явные формулы в случае, когда длина заявки есть сл. в., принимающая конечное число значений.

1.2 Не сохраняющий работу инверсионный порядок обслуживания

Рассмотрим систему $M\,|\,GI\,|\,1\,|\,\infty$, в которую поступает пуассоновский поток заявок интенсивности λ. Длина заявки распределена по закону $B(x)$ с плотностью $b(x) = B'(x)$ и средним $\int_0^\infty x\,b(x)dx = \mathsf{E}S$. Преобразование Лапласа–Стилтьеса (ПЛС) ф. р. $B(x)$ обозначим через $\beta(s)$ т. е. $\beta(s) = \int_0^\infty e^{-xx}dB(x)$. В системе реализован инверсионный порядок обслуживания и следующее правило обработки заявки на приборе. В момент поступления новой заявки становится известной ее длина и, если система непуста, приостанавливается обслуживание. Вне зависимости от всей предыстории функционирования системы заявке на приборе назначается новая (остаточная) длина[11] в соответствии с распределением $B(x)$, процесс ее обслуживания возобновляется, а новая заявка помещается на первое место в очереди. Далее, говоря об этой дисциплине обслуживания, вместо того, чтобы каждый раз писать "дицсиплина инверсионный порядок обслуживания без прерывания и обслуживанием заново с новой реализацией длительности обслуживания", будем писать просто — дисциплина $\mathsf{LIFO\,Re}$ (Re — от англ. resampling).

Система $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ является частным случаем исследованной в предыдущем параграфе СМО $M_k\,|\,GI\,|\,1\,|\,n\,|\,\mathsf{LIFO\,GPP}$, причем $n = \infty$, $\lambda_k = \lambda$ и

$$\begin{gathered} d(x,y|u,v) = b(x)b(y),\ d^*(x,y|u,v) = 0, \\ d(x|u,v) = 0,\ d_0(u,v) = 0,\ u \geqslant 0,\ v \geqslant 0. \end{gathered} \tag{1.9}$$

Поэтому далее будем пользоваться введенными ранее обозначениями и, когда это не вызывает недоразумений, будем опускать нижний индекс (например, у $\psi_k(s;x)$ и $u_k(s;x)$), указывавший ранее на зависимость той или иной характеристики от числа заявок в системе.

[11] Смысл рассмотрения такой экзотической дисциплины обслуживания выяснится только в главе 2. Отметим, однако, что сама идея о назначении заявке новой длины (в момент поступления и независимо от всей предыстории функционирования системы) не является чем-то новым. Еще в [205, С. 362] упоминается, что такая идея оказывается плодотворной, например, при аналитическом исследовании сетей связи: всякий раз, когда сообщение принимается в узле внутри сети, независимо выбирается его новая длина (в действительности же сообщения сохраняют длину при их прохождении по сети).

Теорема 3. *Необходимое и достаточное условие существования стационарного режима системы* $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ *имеет вид*[12]

$$\frac{1}{2} < \beta(\lambda) < 1. \tag{1.10}$$

Доказательство. Свяжем с рассматриваемой системой процесс Гальтона–Ватсона [349; 350], в котором первоначально имеется одна частица, которая в конце жизни производит случайное число потомков в соответствии с (пока еще неизвестным) распределением $\{g_k,\ k \geqslant 0\}$. Заметим, что число заявок, обслуженных рассматриваемой системой за период занятости, равно общему числу частиц, появившихся во введенном процессе Гальтона–Ватсона до его вырождения. Последнее же имеет место с вероятностью единица (за конечное среднее время) тогда и только тогда, когда среднее число $\sum_{k=1}^{\infty} k g_k$ потомков от одной частицы меньше единицы.

Покажем, что последнее условие равносильно (1.10). Обозначим через $r_k(x)$ плотность вероятности того, что длительность пребывания на приборе заявки, которая только что на него поступила, будет равно x и за время ее пребывания на приборе в систему поступит k новых заявок. Поскольку поступающая на прибор заявка имеет длину x с плотностью вероятности $b(x)$, по формуле полной вероятности имеем

$$r_0(x) = e^{-\lambda x} b(x), \tag{1.11}$$

$$r_k(x) = \int_0^x \lambda e^{-\lambda u}(1 - B(u)) r_{k-1}(x-u)\,du,\ k \geqslant 1. \tag{1.12}$$

Для завершения доказательства осталось заметить, что $g_k = \int_0^\infty r_k(x)dx = \beta(\lambda)(1-\beta(\lambda))^k$ и $\sum_{k=1}^{\infty} k g_k = (1-\beta(\lambda))/\beta(\lambda)$.

□

Отметим, что условие (1.10) не зависит от моментов длины заявки какого-либо порядка, т. е. для любого распределения длины заявки при достаточно малой интенсивности λ существует стационарный режим.

[12] Таким образом, роль загрузки в системе $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ играет величина $\beta(\lambda)$.

Теорема 4. *В системе* $M\,|\,GI\,|\,1\,|\,\infty\,|$ LIFO Re *стационарное распределение* P_k, $k \geqslant 0$, *общего числа заявок в системе является геометрическим:*

$$P_k = \left(2 - \frac{1}{\beta(\lambda)}\right)\left(\frac{1-\beta(\lambda)}{\beta(\lambda)}\right)^k; \tag{1.13}$$

стационарные плотности вероятностей состояний $p_k(x_1,\ldots,x_k)$, $k \geqslant 1$, *определяются формулой:*

$$p_k(x_1,\ldots,x_k) = (P_{k-1} + P_k)\int_{x_1}^{\infty} \lambda e^{-\lambda(u-x_1)} dB(u) b(x_2)\cdots b(x_k); \tag{1.14}$$

ПЛС $u(s)$ *периода занятости имеет вид*

$$u(s) = \frac{\lambda + s - \sqrt{(\lambda+s)^2 - 4\lambda(1-\beta(s+\lambda))\beta(s+\lambda)(\lambda+s)}}{2\lambda(1-\beta(s+\lambda))}; \tag{1.15}$$

ПЛС $\varphi(s;x)$ *стационарного распределения времени пребывания в системе заявки длины* x *задается выражением*

$$\varphi(s;x) = P_0\psi(s;x) + (1-P_0)\psi(s;x)u(s), \tag{1.16}$$

где ПЛС $\psi(s;x)$ *распределения времени пребывания заявки длины* x *на приборе, опредяется формулой*

$$\psi(s;x) = e^{-(\lambda+s)x} + \frac{\lambda\left(1-e^{-(\lambda+s)x}\right)}{\lambda+s}\psi(s), \quad \text{и} \quad \psi(s) = \frac{\beta(\lambda+s)(\lambda+s)}{s+\lambda\beta(\lambda+s)}. \tag{1.17}$$

Доказательство. Воспользуемся *Теоремой 1*. Интегрируя (1.3) по x_2,...,x_k в пределах от нуля до бесконечности, с учетом (1.9), получаем следующую систему[13] дифференциальных уравнений для $p_k(x)$, $k \geqslant 1$:

$$-p_k'(x) = \lambda b(x) P_{k-1} - \lambda p_k(x) + \lambda b(x) P_k. \tag{1.18}$$

Определим ПФ

$$\pi(z,x) = \sum_{k=1}^{\infty} z^k p_k(x), \quad \pi(z) = \sum_{k=1}^{\infty} z^k P_k.$$

[13] Доказательство единственности ее решения полностью повторяет доказательство единственности для системы в [321, С. 59–62].

Вообще говоря, из *Теоремы 1.1.1* следует, что P_0 зависит от n. Поэтому каждая из введенных ПФ без дополнительной оговорки не имеет смысла обычной ПФ. Но, если не связывать P_0 с какой-то определенной системой (т. к. с каким-то определенным значением n), а оставить как свободный параметр, то по аналогии с тем, как это сделано в [321, С. 48] (см. также [59, С. 89]), можно показать, что ряды в определениях ПФ $\pi(z,x)$ и $\pi(z)$ сходятся, по крайней мере, при достаточно малых z. Умножая (1.18) на z^k и суммируя, получаем

$$-\pi'(z,x) = \lambda z b(x) P_0 + \lambda b(x)(1+z)\pi(z) - \lambda\pi(z,x).$$

Интегрируя это уравнение с учетом граничного условия $\pi(z,\infty) = 0$, имеем

$$\pi(z,x) = \int_x^\infty e^{-\lambda(u-x)}(\lambda z b(u) P_0 + \lambda(1+z)b(u)\pi(z))du.$$

Осталось определить только $\pi(z)$, для чего предыдущую формулу проинтегрируем по x от нуля до бесконечности и приведем подобные слагаемые:

$$\pi(z) = P_0 \frac{z\frac{1-\beta(\lambda)}{\beta(\lambda)}}{1 - z\frac{1-\beta(\lambda)}{\beta(\lambda)}}. \tag{1.19}$$

Вероятность P_0, как обычно, находится из условия нормировки $\sum_{k=0}^\infty P_k = 1$ и имеет вид $P_0 = 2 - (\beta(\lambda))^{-1}$. Коэффициенты при z^k в разложении в ряд по степеням z функции $\pi(z)$ дают (1.13). Из формулы для $\pi(z,x)$, с учетом найденного вида $\pi(z)$, получается (1.14) при $k = 1$. Методом[14] математической индукции можно убедиться, что (1.14) верно и при произвольном $k \geqslant 2$. Покажем, что это действительно возможно, на случае $k = 2$. Плотность $p_1(t;x)$ вероятности того, что в системе в момент времени t находится одна заявка остаточной длины x удовлетворяет, согласно формуле полной вероятности, соотношению

$$p_1(t+\Delta;x) = P_0(t)\lambda\Delta b(x) + p_1(t;x+\Delta)(1-\lambda\Delta) + (1-\lambda\Delta)\int_0^\Delta p_2(t;y,x)dy + o(\Delta).$$

Перенося $p_1(t;x+\Delta)$ в левую часть, деля на Δ, утремляя Δ к нулю и учитывая стационарным режим функционирования, получаем уравнение

$$-p_1'(x) = P_0\lambda b(x) - \lambda p_1(x) + p_2(0,x),$$

[14] Или заметив, что длины заявок в очереди являются независимыми в совокупности случайными величинами, каждая с ф. р. $B(x)$.

решение которого, с учетом граничого условия $p_1(\infty) = 0$, имеет вид

$$p_1(x) = \int_x^\infty e^{-\lambda(u-x)} \left(P_0 \lambda b(u) + p_2(0,u)\right) du. \tag{1.20}$$

Выпишем уравнение для плотности $p_2(t; x,y)$ вероятности того, что в системе в момент времени t находится две заявки, на приборе — остаточной длины x, в очереди — остаточной длины y. Рассматривая моменты t и $t+\Delta$, получаем соотношение

$$p_2(t+\Delta; x,y) = \int_0^\infty p_1(t;u)du\lambda\Delta b(x)b(y) + p_2(t; x+\Delta, y)(1-\lambda\Delta) + \\ + (1-\lambda\Delta)\int_0^\Delta p_3(t; u,x,y)du + o(\Delta),$$

из которого следует, что

$$p_2(x,y) = \int_x^\infty e^{-\lambda(u-x)} \left(P_1 \lambda b(u) b(y) + p_3(0,u,y)\right) du. \tag{1.21}$$

Подставляя в (1.20) вместо $p_1(x)$ ее выражение по формуле (1.14) и дифференцируя левую и правую части один раз по x, находим

$$p_2(0,x) = \lambda P_1 b(x).$$

Вернемся к (1.21), положим $x = 0$ и подставим найденное выражение для $p_2(0,x)$. После приведения подобных слагаемых имеем

$$\int_0^\infty e^{-\lambda u} p_3(0,u,y) du = \lambda P_1 (1 - \beta(\lambda)) b(y).$$

Предположим, что $p_3(0,u,y)$ имеет вид $p_3(0,u,y) = f(u)b(y)$, где f — некоторая неизвестная, но непрерывная и ограниченная функция при $u \geqslant 0$. Тогда используя новый вид $p_3(0,u,y)$ в (1.21) и интегрируя по всем y в пределах от 0 до ∞, приходим к соотношению

$$p_2(x) = \int_x^\infty e^{-\lambda(u-x)} \left(P_1 \lambda b(u) + f(u)\right) du,$$

или, учитывая вид $p_2(x)$ по формуле (1.14), —

$$\lambda \left(P_1 + P_2\right) \int_x^\infty e^{-\lambda(u-x)} b(u) du = \int_x^\infty e^{-\lambda(u-x)} \left(P_1 \lambda b(u) + f(u)\right) du.$$

Приводя подобные слагаемые и дифференцируя левую и правую части один раз по x, находим $f(x) = \lambda P_2 b(x)$. Подставляя $p_3(0,u,y) = f(u)b(y) = \lambda P_2 b(u)b(y)$ в (1.21), убеждаемся, что (1.14) справедливо при $k = 2$.

Для нахождения указанных в формулировке теоремы временных характеристик достаточно воспользоваться *Теоремой 2*, поскольку при дисциплине LIFO Re заявки не могут покинуть систему недообслуженными. Полагая в (1.7) $k = 1$ и замечая, что $u_1(s;x) \equiv u_2(s;x)$, получаем уравнение для ПЛС $u(s)$ распределения периода занятости:

$$u(s) = \int_0^\infty u_1(s;x)dB(x) = \frac{\lambda}{\lambda+s}u^2(s) + \beta(s+\lambda)\left(1 - \frac{\lambda}{\lambda+s}u^2(s)\right).$$

Традиционные рассуждения (см., например, [320, С. 64–66], [119, С. 14–15] или [263, С. 300–301]) показывают, что из двух корней этого квадратного уравнения подходит лишь тот, который задается формулой[15] (1.15). Когда стационарный решим существует (т. е. выполнено (1.10)), $0 < u(s) \leqslant 1$ при каждом $s \geqslant 0$, причем $u(0) = 1$ и $u(s) \to 1$ при $s \to 0$. Дифференцируя (1.15) один раз в точке $s = 0$, получаем выражение для средней длины $\mathsf{E}U$ периода занятости:

$$\mathsf{E}U = -u'(0) = \frac{1 - \beta(\lambda)}{\lambda(2\beta(\lambda) - 1)}. \tag{1.23}$$

[15] Для $u(s)$ можно предложить и другую формулу, переписав (1.15) так:

$$u(s) = \frac{P + L + M - \sqrt{(P+L+M)^2 - Q^2}}{2L}, \tag{1.22}$$

где $P = P(s) = s(1-\beta(s+\lambda))$, $L = L(s) = \lambda(1-\beta(s+\lambda))$, $M = M(s) = \beta(s+\lambda)(\lambda+s)$, $Q = Q(s) = 2\sqrt{LM}$. Для выражений вида (1.22) известно интегральное представление (см., например, [351]):

$$\frac{P + L + M - \sqrt{(P+L+M)^2 - Q^2}}{2L} = \frac{2M}{\pi}\int_{-1}^{1} \frac{\sqrt{1-t^2}}{P+L+M-Qt}dt.$$

Подставляя в него явный вид P, L, M и Q, получаем

$$\begin{aligned} u(s) &= \frac{2\beta(s+\lambda)}{\pi}\int_{-1}^{1} \frac{\sqrt{1-t^2}}{1-2t\sqrt{\frac{\lambda(1-\beta(s+\lambda))\beta(s+\lambda)}{\lambda+s}}}dt = \\ &= \frac{2\beta(s+\lambda)}{\pi}\int_{-1}^{1} \sqrt{1-t^2}\sum_{m=0}^{\infty}\left(4t^2\lambda(1-\beta(s+\lambda))\beta(s+\lambda)\right)^{\frac{m}{2}}(\lambda+s)^{-\frac{m}{2}}dt = \\ &= \beta(s+\lambda)\sum_{m=0}^{\infty} u_m\left(\lambda(1-\beta(s+\lambda))\beta(s+\lambda)\right)^{\frac{m}{2}}(\lambda+s)^{-\frac{m}{2}}, \end{aligned}$$

где $u_m = \frac{2^{m+1}}{\pi}\int_{-1}^{1}\sqrt{1-t^2}t^m dt$.

Таким образом, при $1/2 < \beta(\lambda) < 1$ период занятости не только конечен с вероятностью единица, но и имеет конечное среднее значение. Отметим, что при $0 < \beta(\lambda) \leqslant 1/2$ период занятости конечен с вероятностью $\beta(\lambda)/(1-\beta(\lambda))$.

Для завершения доказательства осталось заметить, что формулы (1.16) и (1.17) следуют соответственно из (1.6) и (1.8).

□

Дифференцируя (1.19) по z необходимое число раз, можно получить[16] моменты всех порядков стационарного распределения общего числа заявок в системе. Обозначим через N сл. в., распределенную как общее число заявок в системе в стацинарном режиме. Тогда стационарное среднее $\mathsf{E}N$ и второй момент $\mathsf{E}N^2$ определяются следующими формулами:

$$\begin{aligned} \mathsf{E}N &= \frac{1-\beta(\lambda)}{2\beta(\lambda)-1}, \\ \mathsf{E}N^2 &= \frac{1}{2\beta(\lambda)-1}\mathsf{E}N. \end{aligned} \tag{1.24}$$

Аналогичным образом, но из (1.16) и (1.17), находятся и моменты временных характеристик. Например, дифференцируя (1.16) один раз по s в точке $s=0$, получаем формулу[17] для стационарного среднего времени $\mathsf{E}V(x)$ пребывания в системе заявки длины x:

$$\mathsf{E}V(x) = -\varphi'(0;x) = e^{-\lambda x}\frac{1-2\beta(\lambda)}{\lambda\beta^2(\lambda)} + \frac{3\beta^2(\lambda)-3\beta(\lambda)+1}{\lambda\beta(\lambda)(2\beta(\lambda)-1)}.$$

Обозначим через V сл. в., распределенную как время пребывания заявки в системе, находящейся в стацинарном режиме. Усредняя $\mathsf{E}V(x)$ по распределению длины заявки, находим стационарное среднее $\mathsf{E}V$ время пребывания в системе произвольной заявки:

$$\mathsf{E}V = \frac{1-\beta(\lambda)}{\lambda(2\beta(\lambda)-1)}. \tag{1.25}$$

Сравнивая (1.23), (1.24) и (1.25), приходим к следующим выводам:

[16] Для моментов высших порядков более полезными могут оказаться рекуррентные формулы расчета; см., например, [352].

[17] Очевидно, что для консервативных СМО $\frac{\mathsf{E}V(x)}{x} \geqslant 1$. Отличительной особенностью исследуемой системы является то, что $\frac{\mathsf{E}V(x)}{x} \to \infty$ при $x \to 0$ и $\frac{\mathsf{E}V(x)}{x} \to 0$ при $x \to \infty$. Такое положение дел (вспоминая смысл, который можно придать отношению $\frac{V(x)}{x}$; см. стр. 46) свидетельствует о том, что в неконсервативных однолинейных СМО отношение $\frac{V(x)}{x}$ может не иметь никакого отношения к справедливости принятого правила обслуживания.

– для исследуемой системы $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$, которая является неконсервативной, справедлив закон Литтла[18];
– средняя длина периода занятости системы $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ равна среднему времени пребывания в системе произвольной заявки[19]. В [294] показано, что это свойство системы может быть объяснено путем сравнения (с помощью каплинг метода [355–357]) исследуемой системы с классической однолинейной СМО с дисциплиной LIFO, абсолютным приоритетом и дообслуживанием. Необходимо отметить, что при увеличении числа входящих потоков и/или числа обслуживающих приборов, это свойство не сохраняется.

Еще одна характеристика, которую можно получить из *Теоремы 4*, — это распределение незаконченной работы в системе в стационарном режиме. Обозначим через $\chi(t)$ величину незаконченной работы в системе в момент t, т. е. это[20] та "работа", которую должен совершить, начиная от момента t, прибор, если после момента t в системе не будут больше поступать заявки. Обозначим через $R(x) = \lim_{t\to\infty} \mathsf{P}\{\chi(t) < x\}$ стационарное распределение незаконченной

[18]Справедливо и другое соотношение, названное в [268, С. 263] считающим законом Литтла (odrinary Little's law): стационарное среднее число заявок в системе в момент поступления равно среднему числу поступлений за время пребывания заявки в системе, находящейся в стационарном режиме. Убедиться в этом можно, воспользовавшись аппаратом ПФ и методом коллективных меток [353, С. 281–288] (метод "катастроф" [320, С. 13]). Действительно, обозначая через $Q_0(z)$ и $Q_1(z)$ ПФ числа заявок, поступивших соответственно за время пребывания заявки на приборе и за время ее пребывания в очереди, по формуле полной вероятности находим искомое среднее: $(p_0Q_0(z) + (1-p_0)Q_1(z)Q_0(z))'\big|_{z=1}$, где $Q_0(z) = \frac{\beta(\lambda)}{1-(1-\beta(\lambda))z}$, а $Q_1(z)$ — соответствующее решение уравнения $Q_1(z) = \beta(\lambda) + (1-\beta(\lambda))zQ_1^2(z)$. Вопрос справедливости считающего закона Литтла в многолинейной системе остается невыясненным.

[19]Как известно (см., например, [354, С. 487]), также обстоит дело и в системе $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$.

[20]Не виртуальное время ожидания, т. к. время ожидания начала обслуживания поступившей в момент t заявки может быть как больше, так и меньше $\chi(t)$.

работы[21]. По свойству ПЛС из (1.14) немедленно получаем[22]:

$$\varrho(s) = \int_0^\infty e^{-sx} dR(x) = P_0 + \frac{\lambda P_0}{\beta(\lambda)} \frac{\beta(s) - \beta(\lambda)}{\lambda - s} \frac{1}{1 - \frac{1-\beta(\lambda)}{\beta(\lambda)}\beta(s)}.$$

Дифференцируя последнюю формулу по s необходимое число раз, можно получить моменты всех порядков незаконченной работы в системе в стационарном режиме. Например, среднее значение равно

$$\mathsf{E}R = \frac{\mathsf{E}S}{P_0} - P_0 \frac{1 - \beta(\lambda)}{\lambda(2\beta(\lambda) - 1)}. \tag{1.26}$$

где, напомним, $\mathsf{E}S = \int_0^\infty x b(x)\, dx$ — среднее значение длины заявки.

Теорема 5. *Если для системы* $M \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\ Re}$ *выполнено* (1.10), *то при* $\beta(\lambda) \uparrow \frac{1}{2}$

$$\mathsf{P}\left\{\frac{N}{\mathsf{E}N} < x\right\} \to 1 - e^{-x},\ x \geqslant 0. \tag{1.27}$$

Доказательство. Положим $\pi^*(s) = \pi(e^{-s/\mathsf{E}N})$ — преобразование Лапласа–Стилтьеса нормированной своим математическим ожиданием стационарной очереди для системы $M \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\ Re}$. Подставляя в формулу (1.19) $e^{-s/\mathsf{E}N}$ вместо z, получаем

$$\pi^*(s) = \frac{\beta(\lambda) - 1}{1 - \beta(\lambda)\left(1 + e^{\frac{2\beta(\lambda)-1}{\beta(\lambda)}s}\right)} \left(2 - \frac{1}{\beta(\lambda)}\right).$$

Разлагая теперь $e^{-s/\mathsf{E}N}$ по степеням $s/\mathsf{E}N$ до первой степени, получаем, что при $\beta(\lambda) \uparrow \frac{1}{2}$

$$\pi^*(s) \to \frac{1}{1+s},$$

[21] Оно существует всегда, когда существует стационарное распределение длины очереди. Действительно, условие $1/2 < \beta(\lambda) < 1$ гарантирует, что средняя длина цикла регенерации системы, состоящего из периода занятости и следующего за ним свободного периода, конечна. Значит, и число раз, когда происходило изменение остаточной длины заявки на приборе, также конечно. Процесс $\chi(t)$ является регенерирующим относительно моментов окончания периодов регенерации системы. Таким образом, выполнены условия теоремы Смита для регенерирующих процессов (см. [263, Теорема 3] или [358, С. 184]), из следствия из которой следует утверждение. См. также доказательство Теоремы 2.2 в [75].

[22] Под значением $\varrho(s)$ в точке $s = \lambda$ понимается $\lim_{s\to\lambda} \varrho(s) = P_0\left(1 - \lambda\frac{\beta'(\lambda)}{\beta(\lambda)^2}\right)$.

т. е. при загрузке, стремящейся к единице, предельное распреление нормированного числа заявок в системе является экспоненциальным и стационарное распределение общего числа заявок в системе имеет вид

$$\mathsf{P}\{N=k\} = \left(1 - e^{-\frac{2\beta(\lambda)-1}{\beta(\lambda)}}\right) e^{-\frac{2\beta(\lambda)-1}{\beta(\lambda)}k}, \ k \geqslant 0.$$

□

Вопрос нахождения распределения, аппроксимирующего распределение времени пребывания заявки в стационарной системе, но при загрузке близкой к критической, является открытым. Его изучение традиционными методами [359–362] невозможно[23]. Выберем λ^* так, чтобы $\beta(\lambda^*) = 1/2$; такое λ^* существует, единственно и $0 < \lambda^* < \infty$. Предполагая, что в формуле для $\varphi(s) = \int_0^\infty \varphi(s;x)dB(x)$ можно положить $\lambda = \lambda^*$, получаем, что на левой границе интервала стационарности (1.10) время пребывания заявки в системе есть собственная случайная величина с распределением, имеющим ПЛС

$$\varphi^*(s) = \frac{\beta(\lambda^*+s)(\lambda^*+s)}{s+\lambda^*\beta(\lambda^*+s)}u^*(s), \tag{1.28}$$
$$u^*(s) = \frac{\lambda^*+s-\sqrt{(\lambda^*+s)^2 - 4\lambda^*(1-\beta(s+\lambda^*))\beta(s+\lambda^*)(\lambda^*+s)}}{2\lambda^*(1-\beta(s+\lambda^*))},$$

и бесконечное среднее т. к. $-(\varphi^*(s))'|_{s=0} = \infty$. Значит, нельзя подобрать такую величину $c \to \infty$ одновременно с $\beta(\lambda) \to 1/2$, что сл. в. V/c сходится (в смысле слабой сходимости) к собственной ненулевой сл. в. Этот эффект[24] не является удивительным и известен для некоторых классических однолинейных СМО (например, СМО с абсолютным приоритетом при обратном порядке обслуживания [93; 363]).

Невыясненным остается и вопрос получения в этом направлении содержательных результатов на основе хорошо развитой теории случайного

[23] Отдельно отметим подход из работ [363; 364] для СМО с инверсионным порядком обслуживания, попытка применить который к исследуемой системе не предпринималась.

[24] Исчезает, при замене в исследуемой системе дисциплины LIFO на FIFO (см. формулу (19) в [294]).

суммирования [365–375] (см. также [376–381]). Введем обозначения[25]: U — сл. в., равная длительности ПЗ, K — сл. в., равная времени пребывания на приборе (только что поступившей на него) заявки при условии, что приход очередной (новой) заявки случился раньше, чем закончилось обслуживание; D — сл. в., равная времени пребывания на приборе (только что поступившей на него) заявки при условии, что ее обслуживание закончилось раньше прихода очередной (новой) заявки; C — сл. в., равную полному времени обслуживания (только что поступившей на прибор) заявки; A — сл. в., равную числу новых поступления за время пребывания на приборе (только что поступившей на него) заявки. Тогда время пребывания V в системе произвольной заявки представимо в виде суммы трех слагаемых: C, D и безгранично делимой случайной суммы, а именно[26]:

$$V = C + D + \sum_{i=1}^{A-1}(K_i + U_i),$$

где U_i и K_i — независимые копии U и K. Однако $\mathsf{E}A \to 2$ и $\mathsf{E}U_i \to \infty$ при $\beta(\lambda) \to 1/2$.

Теорема 6. *В системе* $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ *в установившемся режиме поток заявок, покидающих систему, является пуассновским тогда и только тогда, когда распределение* $B(x)$ *длины заявки имеет экспоненциальное распределение. ПЛС совместного стационарного распределения длительностей двух последовательных интервалов времени между моментами окончания*

[25] Формулы для ПЛС распределений и числовых характеристик введенных сл. в. не требуют пояснений:

$$\mathsf{P}\{A=k\} = \beta(\lambda)(1-\beta(\lambda))^{k-1},\ k \geqslant 1,\quad \mathsf{E}e^{-sD} = \frac{\beta(\lambda+s)}{\beta(\lambda)},\ \mathsf{E}D = -\frac{\beta'(\lambda)}{\beta(\lambda)},\ \mathsf{Var}D = \frac{\beta''(\lambda)}{\beta(\lambda)} - \left(\frac{\beta'(\lambda)}{\beta(\lambda)}\right)^2,$$
$$\mathsf{E}e^{-sK} = \frac{\lambda}{\lambda+s}\frac{1-\beta(\lambda+s)}{1-\beta(\lambda)},\quad \mathsf{E}K = \frac{1}{\lambda} + \frac{\beta'(\lambda)}{1-\beta(\lambda)},\quad \mathsf{Var}K = \frac{1}{\lambda^2} - \frac{\beta''(\lambda)}{1-\beta(\lambda)} - \left(\frac{\beta'(\lambda)}{1-\beta(\lambda)}\right)^2,$$
$$\mathsf{E}C = \mathsf{E}D + \mathsf{E}K\frac{1-\beta(\lambda)}{\beta(\lambda)},\quad \mathsf{E}U = \mathsf{E}D\frac{\beta(\lambda)}{2\beta(\lambda)-1} + \mathsf{E}K\frac{1-\beta(\lambda)}{2\beta(\lambda)-1}.$$

[26] Ввиду результатов изложенных в главе 2, полезным может оказаться изучение свойств распределения сл. в. V (и свойств распределений сл. в., связанных с ней). За основу могут быть взяты известные результаты для классических СМО. Например (см. [382]), если времена обслуживания в СМО $GI\,|\,G\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$ имеют убывающую функцию интенсивности, то в стационарном режиме ф. р. времени ожидания начала обслуживания является вогнутой.

обслуживания имеет вид[27]

$$\eta(s_1,s_2)=\psi(s_2)\left(\psi(s_1)-\frac{P_0\left(s_1\psi(s_1)+\frac{\lambda s_2\beta(s_1+\lambda)}{\lambda+s_2}\right)}{\lambda+s_1}-\frac{P_1 s_2\beta(s_1+\lambda)}{\lambda+s_2}\right). \qquad (1.29)$$

В установившемся режиме первый и второй моменты длины интервала времени между моментами выхода заявок из системы, а также ковариция между длинами двух последовательных интервалов соответственно равны

$$\frac{1}{\lambda},\quad \frac{2}{\lambda^2\beta(\lambda)^2}\left(\beta(\lambda)+\lambda\beta'(\lambda)\right),\quad \frac{P_0}{\lambda^2}\left(1+\beta(\lambda)-\lambda\beta'(\lambda)-\lambda(1-P_0)\beta'(\lambda)\right). \qquad (1.30)$$

Доказательство. Обозначим через $\tau_1,\tau_2,\ldots,\tau_n,\ldots$ моменты окончания обслуживания первой, второй, ..., n-й, ... заявки, и через $\nu_n=\nu(\tau_n+0)$ — общее число заявок в системе сразу после момента τ_n. Пусть $P_k^+=\lim_{n\to\infty}\mathsf{P}\{\nu_n=k\}$, — стационарная вероятность того, что число заявок в системе в момент окончания обслуживания очередной заявки равно k, $k\geqslant 0$. Для существования P_k^+ необходимо и достаточно выполнения[28] (1.10). Матрица переходных вероятностей $\mathbb{P}=(p_{ij})_{i,j\geqslant 0}$ вложенной по моментам окончания обслуживания цепи Маркова $\{\nu_n, n\geqslant 1\}$ имеет вид такой же, как и для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$, а переходные вероятности равны:

$$p_{ij}=\begin{cases}0, & 0\leqslant j<i-1,\\ \beta(\lambda)(1-\beta(\lambda))^{j-i+1}=g_{j-i+1}, & j\geqslant i-1,\end{cases}\quad i\geqslant 1,$$
$$p_{0j}=p_{1j},\ j\geqslant 0,$$

Из системы уравнений равновесия $\vec{P}^+=\vec{P}^+\mathbb{P}$, $\vec{P}^+\vec{1}=1$, $\vec{P}^+=(P_0,P_2,\ldots)$, $\vec{1}=(1,1,\ldots)^{\mathrm{T}}$, следует, что $P_k^+=P_k$, $k\geqslant 0$. Таким образом, для исследуемой системы справедлив закон стационарной очереди Хинчина: стационарное по времени распределение (1.13) числа заявок в системе совпадает со стационарным распределением P_k^+, $k\geqslant 0$, числа заявок в системе для вложенной цепи Маркова, порожденной моментами ухода заявок из системы.

[27] Необходимо отметить схожесть вида (1.29) с ПЛС аналогичной характеристики в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$ [383, (2.7)]. Кроме того, для последней СМО имеет место BRAVO–эффект [384]; не известно имеет ли он место в изучаемых здесь СМО.

[28] Достаточное условие следует из критерия Мустафы [263, С. 260]; необходимое — из того, что вероятность P_0^+ должна быть положительной, но, как будет видно далее, $P_0^+=P_0=2-1/\beta(\lambda)$.

Пусть l_n длина интервала времени между моментами ухода n-й и $(n+1)$-й заявок из системы и пусть

$$h_{n+1}(t,j)dt = \mathsf{P}\{\nu_{n+1} = j, t < l_n < t + dt\},\ j \geqslant 0,$$
$$h_{n+1}(t)dt = \sum_{j=0}^{\infty} h_{n+1}(t,j)dt = \mathsf{P}\{t < l_n < t + dt\}.$$

Положим $r(x) = \sum_{k=0}^{\infty} r_k(x)$, где $r_k(x)$ задаются (1.11) и (1.12), и сразу же заметим[29], что $\int_0^{\infty} e^{-su} r(u)du = \psi(s)$, где $\psi(s)$ задается (1.17). Воспользовавшись теперь формулой полной вероятности, находим соотношение для плотности $h_{n+1}(t)$:

$$h_{n+1}(t) = (1 - \mathsf{P}\{\nu_{n+1} = 0\})r(t) + \mathsf{P}\{\nu_{n+1} = 0\} \int_0^t \lambda e^{-\lambda(t-u)} r(u)du.$$

Т. к. в установившемся режиме $\mathsf{P}\{\nu_{n+1} = k\} = P_k^+$, $k \geqslant 0$, то существует предел $\lim_{n\to\infty} h_{n+1}(t)$, который обозначим $h(t)$. Переходя в предыдущем соотношении к пределу при $n \to \infty$, имеем

$$h(t) = (1 - P_0^+)r(t) + P_0^+ \int_0^t \lambda e^{-\lambda(t-u)} r(u)du. \tag{1.31}$$

Соотношение для $h_{n+1}(t,j)$ получается из формулы полной вероятности тем же самым образом:

$$h_{n+1}(t,j) = \mathsf{P}\{\nu_n = 0\} \int_0^t \lambda e^{-\lambda u} r_j(t-u)du + \sum_{k=1}^{j+1} \mathsf{P}\{\nu_n = k\} r_{j+1-k}(t),\ j \geqslant 0.$$

И опять, поскольку в установившемся режиме $\mathsf{P}\{\nu_{n+1} = k\} = P_k^+$, $k \geqslant 0$, то существуют и пределы $\lim_{n\to\infty} h_{n+1}(t,j) = h(t,j)$, $j \geqslant 0$, причем

$$h(t,j) = P_0^+ \int_0^t \lambda e^{-\lambda u} r_j(t-u)du + \sum_{k=1}^{j+1} P_k^+ r_{j+1-k}(t).$$

Учитывая, что приход каждой новой заявки в непустую систему откладывает (на случайное время) момент окончания обслуживания заявки на приборе, естественно ожидать, что число заявок в системе в момент ухода произвольной заявки и длина интервала времени от момента предыдущего ухода являются зависимыми величинами, т. е. равенства

$$h(t,j) = h(t)P_j^+,\ j \geqslant 0, \tag{1.32}$$

[29] Предполагая, что операция почленного интегрирования законна.

не выполняются. Докажем[30], что (1.32) справедливо тогда и только тогда, когда распределение длины заявки имеет экспоненциальное распределение. Для этого достаточно показать, что (1.32) не выполняется при произвольном распределении $B(x)$ уже при $j = 0$. Используя явный вид $h(t,0)$, $h(t)$, P_0^+ и $r_k(x)$, распишем подробнее равенство $h(t,0) = h(t)P^+(0)$:

$$P_0^+\lambda e^{-\lambda t}B(t)+P_1^+e^{-\lambda t}b(t)=P_0^+(1-P_0^+)r(t)+(P_0^+)^2\int_0^t\lambda e^{-\lambda(t-u)}r(u)du. \quad (1.33)$$

Продифференцируем левую и правую части по t. Имеем

$$\begin{aligned}&-P_0^+\lambda^2e^{-\lambda t}B(t)+P_0^+\lambda e^{-\lambda t}b(t)-P_1^+\lambda e^{-\lambda t}b(t)+P_1^+e^{-\lambda t}b'(t)=\\&\quad=P_0^+(1-P_0^+)r'(t)-(P_0^+)^2\lambda\int_0^t\lambda e^{-\lambda(t-u)}r(u)du+(P_0^+)^2\lambda r(t). \quad (1.34)\end{aligned}$$

Домножив (1.33) на λ и сложив с (1.34), получим

$$P_0^+\lambda e^{-\lambda t}b(t)+P_1^+e^{-\lambda t}b'(t)=P_0^+(1-P_0^+)r'(t)+\lambda P_0^+r(t),$$

или, учитывая, что $P_0^+(1-P_0^+)=P_1^+$,

$$P_0^+\lambda e^{-\lambda t}b(t)+P_1^+e^{-\lambda t}b'(t)=\lambda P_0^+r(t)+P_1^+r'(t).$$

Вспоминая, что $r(x)=\sum_{k=0}^{\infty}r_k(x)=e^{-\lambda x}b(x)+\sum_{k=1}^{\infty}r_k(x)$, после приведения подобных слагаемых[31], получаем уравнение

$$P_1^+\lambda e^{-\lambda t}b(t)-\lambda P_0^+\sum_{k=1}^{\infty}r_k(t)-P_1^+\sum_{k=1}^{\infty}r_k'(t)=0$$

или, учитывая явный вид P_1^+ и P_0^+, — уравнение

$$\lambda e^{-\lambda t}b(t)-\frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}\sum_{k=1}^{\infty}r_k(t)-\sum_{k=1}^{\infty}r_k'(t)=0.$$

Решение найдем в терминах ПЛ. Положив $\psi^*(s)=\sum_{k=1}^{\infty}\int_0^{\infty}e^{-su}r_k(u)du$, предыдущее уравнение можно переписать в виде:

$$\lambda\beta(s+\lambda)-\frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}\psi^*(s)-s\psi^*(s)=0,$$

[30] В предположении, что b — аналитическая функция.

[31] И предполагая, что операция почленного дифференцирования законна.

откуда следует, что

$$\psi^*(s) = \frac{\lambda\beta(s+\lambda)}{s + \frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}}.$$

Значит, если (1.32) выполняется при $j = 0$, то ПЛС времени пребывания заявки на приборе с одной стороны равно $\psi(s)$ (которое задается формулой (1.17)), а с другой стороны равно $\beta(\lambda + s) + \psi^*(s)$ т. е. $\psi(s) = \beta(\lambda + s) + \psi^*(s)$. После подстановки явного вида слагаемых имеем:

$$\frac{\beta(\lambda+s)(\lambda+s)}{s + \lambda\beta(\lambda+s)} = \beta(\lambda+s) + \frac{\lambda\beta(s+\lambda)}{s + \frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}}.$$

Приводя теперь подобные слагаемые, получаем, что (1.32) выполняется при $j = 0$ тогда и только тогда, когда

$$\beta(\lambda+s) = \frac{\frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}}{s + \lambda + \frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}},$$

или, после обратного преобразования, тогда и только тогда, когда

$$b(x) = \frac{\lambda\beta(\lambda)}{1-\beta(\lambda)} e^{-\frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}x}, \ x \geqslant 0.$$

Разлагая левую и правую части в ряд по степеням x, обозначая через $b^{(i)}(x)$ i-ю производную функции b в точке x, получаем, что (1.32) выполняется при $j = 0$ тогда и только тогда, когда

$$b(0) = \frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}, \quad b^{(i)}(0) = (-1)^i \left(\frac{\lambda\beta(\lambda)}{1-\beta(\lambda)}\right)^{i+1}, \ i \geqslant 1.$$

Из первого равенства немедленно следует, что $\beta(\lambda) = \frac{b(0)}{\lambda + b(0)}$: это возможно тогда и только тогда, когда $B(x)$ имеет экспоненциальное распределение с интенсивностью $b(0)$, и в этом случае справедливо выражение для $b^{(i)}(0)$ при $i \geqslant 1$.

Итак, если $B(x)$ имеет неэкспоненциальное распределение, то (1.32) не может выполняться т. е. число заявок в системе в момент окончания обслуживания зависит от того, сколько прошло времени с момента окончания предыдущего обслуживания. Если же $B(x)$ имеет экспоненциальное распределение (и тогда его интенсивность равна $b(0)$), то (1.32) выполняется при любом $j \geqslant 0$. Для $j = 0$ это было показано выше. Остановимся на случае $j \geqslant 1$.

Из того, что $B(x)$ имеет экспоненциальное распределение следует, что

$$\begin{aligned}
\psi(s) &= \frac{b(0)}{s+b(0)},\\
r(t) &= b(0)e^{-b(0)t},\\
P_0^+ &= 1-\frac{\lambda}{b(0)},\quad P_k^+ = P_0^+\left(\frac{\lambda}{b(0)}\right)^k,\ k\geqslant 1,\\
h(t) &= \lambda e^{-\lambda t},\\
r_k(x) &= b(0)\frac{(\lambda x)^k}{k!}e^{-(\lambda+b(0))x}, k\geqslant 0.
\end{aligned}$$

Тогда формулу для $h(t,j)$, с учетом явного вида входящих в нее слагаемых, можно переписать следующим образом:

$$\begin{aligned}
h(t,j) &= P_0^+\int_0^t \lambda e^{-\lambda(t-u)}r_j(u)du + \sum_{k=1}^{j+1}P_k^+ g_{j+1-k}(t) =\\
&= P_0^+\left(\frac{\lambda}{b(0)}\right)^j \lambda e^{-\lambda t}\int_0^t \frac{b(0)^{j+1}}{j!}u^j e^{-b(0)u}du + \sum_{k=1}^{j+1}P_k^+ r_{j+1-k}(t) =\\
&= P_j^+\lambda e^{-\lambda t}\left(1-\sum_{k=0}^{j}\frac{(b(0)t)^k}{k!}e^{-b(0)t}\right) + \sum_{k=1}^{j+1}P_k^+ r_{j+1-k}(t).
\end{aligned}$$

Учитывая, что последняя сумма равна

$$\sum_{k=1}^{j+1}P_k^+ r_{j+1-k}(t) = \lambda P_j^+\sum_{k=0}^{j}\frac{(b(0)t)^k}{k!}e^{-(\lambda+b(0))t},$$

получаем, что $h(t,j) = \lambda e^{-\lambda t}P_j^+ = h(t)P_j^+$, $j\geqslant 0$.

Изучим теперь распределение интервалов времени между последовательными моментами окончания обслуживания. Введем обозначения:

$$\begin{aligned}
h_{n+2}(t|\tau)dt &= \mathsf{P}\{t < l_{n+1} < t+dt | l_n = \tau\},\\
P_{n+1}(j|\tau) &= \mathsf{P}\{\nu_{n+1} = j | l_n = \tau\},
\end{aligned}$$

Так $h_{n+2}(t|\tau)$ есть условная плотность вероятности того, что длина интервала l_{n+1} равна t, при условии, что длина интервала $l_n = \tau$, а $P_{n+1}(j|\tau)$ — условная вероятность того, что в момент окончания обслуживания $(n+1)$-й заявки в системе окажется j заявок, при условии, что длина интервала $l_n = \tau$. Рассуждая как и ранее, получаем

$$h_{n+2}(t|\tau) = (1-P_{n+1}(0|\tau))r(t) + P_{n+1}(0|\tau)\int_0^t \lambda e^{-\lambda(t-u)}r(u)du,$$

или, в установившемся режиме, —

$$h(t|\tau) = (1 - P(0|\tau))r(t) + P(0|\tau)\int_0^t \lambda e^{-\lambda(t-u)} r(u)du.$$

Если бы длины интервалов между последовательными моментами окончания обслуживания были независимы, то $h(t|\tau) - h(t) = 0$ т. е., учитывая (1.31),

$$(P_0^+ - P(0|\tau))r(t) - (P_0^+ - P(0|\tau))\int_0^t \lambda e^{-\lambda(t-u)} r(u)du = 0.$$

Это равенство возможно только в двух случаях: а) $P_0^+ = P(0|\tau)$, б) $r(t) = \int_0^t \lambda e^{-\lambda(t-u)} r(u)du$. Но а) выполняется тогда и только тогда, когда распределение $B(x)$ длины заявки имеет экспоненциальное распределение. Если же а) не выполняется, то для того, чтобы $h(t|\tau) = h(t)$ необходимо, чтобы выполнялось б). Но это невозможно т. к. из б) следует, что

$$r'(t) = \lambda r(t) - \lambda \int_0^t \lambda e^{-\lambda(t-u)} r(u)du$$

и, значит, $r'(t) = 0$ т. е. $r(t)$ — постоянная, чего быть не может. Таким образом, за исключением случая экспоненциальной длины заявки, промежутки между последовательными окончаниями обслуживания являются зависимыми величинами. Для завершения доказательства найдем их совместное распределение в установившемся режиме. Для плотности $h(t_1,t_2)$ по формуле полной вероятности имеем

$$h(t_1,t_2) = P_0^+ \int_0^{t_1} \lambda e^{-\lambda u} r_0(t_1 - u)du \int_0^{t_2} \lambda e^{-\lambda v} r(t_2 - v)dv +$$
$$+ P_0^+ \int_0^{t_1} \lambda e^{-\lambda u} \sum_{k=1}^{\infty} r_k(t_1 - u)du\, r(t_2) + P_1^+ r_0(t_1) \int_0^{t_2} \lambda e^{-\lambda u} r(t_2 - v)dv +$$
$$+ P_1^+ \sum_{j=1}^{\infty} r_j(t_1)r(t_2) + \sum_{k=2}^{\infty} P_k^+ r(t_1)r(t_2).$$

Переходя к ПЛ $\eta(s_1,s_2) = \int_0^\infty \int_0^\infty e^{-s_1 t_1} e^{-s_2 t_2} h(t_1,t_2)dt_1 dt_2$, получаем (1.29). Числовые характеристики (1.30) выходящего потока получаются дифференцированием (1.29).

□

На основе полученных результатов можно было бы углубиться в анализ[32] стационарных характеристик СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ (например, совместного стационарного распределения основных характеристик обслуживания на одном периоде занятости). Однако, как станет понятно только в главе 2, больший интерес (по крайней мере для задач практики) представляют другие вопросы — вопросы изучения разновидностей этой системы. Некоторым из них посвящены следующие параграфы.

[32]И в изучение связей между характеристиками подобных неконсервативных и классических СМО. Например, вопросы приближения многолинейных систем однолинейными хорошо освещены в научной литературе (см., например, [385; 386]). Полученные в этом параграфе результаты позволяют по-другому посмотреть, по крайней мере, на простейшие из них. Рассмотрим классическую СМО $M\,|\,M\,|\,\infty$ с интенсивностью входящего потока λ и интенсивностью обслуживания μ. Как известно, стационарное среднее число заявок в такой бесконечнолинейной системе равно λ/μ. Однако, такое же среднее наблюдается и в стационарной СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с ф. р. длины заявки $B(x)$ ПЛС которой равно $\beta(\lambda) = \frac{\lambda+\mu}{2\lambda+\mu}$ (см. выражение для $\mathsf{E}N$ на стр. 57). При этом стационарные распределения чисел заявок в системах не совпадают. Вернемся к $B(x)$: подходящая ф. р. содержит дискретную компоненту, а плотность может быть формально записана в виде $b(x) = B'(x) = \frac{1}{2}\delta(x) + \frac{\mu}{4}e^{-\frac{\mu}{2}x}$, $x \geqslant 0$, где δ — дельта–функция Дирака. Поэтому, говоря строго, результаты *Теоремы 4* неприменимы, поскольку были получены в предположении абсолютной непрерывности распределения длины заявки. Однако воспользовавшись вместо δ какой-нибудь несимметричной функцией f (например, $f(x) = \frac{\varepsilon}{x^2}e^{-\frac{\varepsilon}{x}}$, $x \geqslant 0$, дающей δ в пределе при $\varepsilon \to 0$), получаем подходящую функцию b, при которой среднее число заявок в новой СМО $\approx \lambda/\mu$.

1.3 Обслуживание нескольких потоков без преимущества

Рассмотрим систему $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$, в которую поступает r $(r < \infty)$, независимых пуассоновских потоков интенсивностей λ_i, $1 \leqslant i \leqslant r$. Длины заявок i-го потока (типа) имеют ф. р. $B_i(x)$ с непрерывной ограниченной плотностью $b_i(x) = B_i'(x)$. Будем считать, что в случае нескольких входящих потоков дисциплина $\mathsf{LIFO\,Re}$ работает следующим образом. В момент поступления новой заявки любого типа становится известной ее длина и, если система непуста, приостанавливается обслуживание. Вне зависимости от всей предыстории функционирования системы заявке на приборе назначается новая (остаточная) длина в соответствии с распределением, соответствующим ее типу. Затем процесс ее обслуживания возобновляется, а новая заявка помещается на первое место в очереди. В момент окончания обслуживания на прибор выбирается первая заявка из очереди.

Отличительной особенностью рассматриваемой системы является отсутствие приоритетов для входящих потоков т. е. она не относится к хорошо известному классу приоритетных СМО [117–119; 387]. Как следствие, если две или более из ф. р. $B_1(x), \ldots, B_r(x)$ совпадают, то система не может отличить друг от друга заявки этих потоков. Естественный выход[33] из такого положения — считать все заявки с одинаковыми ф. р. длин принадлежащими одному потоку, но большей интенсивности. Далее будем считать, что такой анализ системы уже проведен (т. е. все $B_i(x)$ различны) и (для удобства изложения) типы потоков занумерованы в порядке возрастания значений $1 - \beta_i(s)$ т. е. 1-й тип присваивается потоку с наименьшим значением $1 - \beta_i(s)$ и т. д.

Введем случайный процесс $\eta(t) = (\nu(t), \xi_1(t), \iota_1(t), \ldots, \xi_{\nu(t)}(t), \iota_{\nu(t)}(t))$, описывающий функционирование системы, как вектор длин заявок, находящихся в системе в момент t. Когда в момент t в системе находится k заявок, то $\nu(t) = k$. При этом, координаты $\xi_1(t)$ и $\iota_1(t)$ хранят остаточное время обслуживания заявки на приборе и ее тип, а $\xi_i(t)$ и $\iota_i(t)$ — длину и тип $(i-1)$-й заявки в очереди. Процесс $\eta(t)$ является марковским с множеством состояний

$$\bigcup_{k=1}^{\infty} \{(k; (x_1, y_1), \ldots, (x_k, y_k)) : \; x_1, \ldots, x_k \geqslant 0, 1 \leqslant y_1, \ldots, y_k \leqslant r\} \bigcup \{0\}$$

[33] Поскольку все входящие потоки предполагаются пуассоновскими.

и описывает состояние очереди и прибора в момент t. Положим

$$
\begin{aligned}
P_0(t) &= \mathsf{P}\{\nu(t) = 0\},\\
P_k^{(y_1,\ldots,y_k)}(t;x_1,\ldots,x_k) &= \mathsf{P}\left\{\nu(t) = k;\, \xi_1(t) < x_1, \iota_1(t) = y_1, \ldots, \xi_k(t) < x_k, \iota_k(t) = y_k\right\},\\
P_0 &= \lim_{t\to\infty} P_0(t),\\
P_k^{(y_1,\ldots,y_k)}(x_1,\ldots,x_k) &= \lim_{t\to\infty} P_k^{(y_1,\ldots,y_k)}(t;x_1,\ldots,x_k),\\
P_k^{(i)}(x) &= \sum_{y_2,\ldots,y_k=1}^{r} P_k^{(i,y_2,\ldots,y_k)}(x_1,\ldots,x_k)(x,\infty,\ldots,\infty),\\
p_k^{(i)}(x) &= \frac{d}{dx}P_k^{(i)}(x),\ \ P_k = \sum_{i=1}^{r} P_k^{(i)}(\infty).
\end{aligned}
$$

При выполнении некоторого условия, которое будет получено в дальнейшнем (см. *Теорему 9* ниже), у рассматриваемой системы существует стационарный режим. Относительно стационарных плотностей $p_k^{(i)}(x)$ будем предполагать, что они существуют, являются ограниченными и непрерывными.

Введем необходимые для дальнейшего изложения обозначения[34]:

$$
\begin{aligned}
\mathbf{p}_k^{\mathrm{T}} &= (P_k^{(1)},\ldots,P_k^{(r)}),\ \ k \geqslant 1,\\
\mathbf{b}^{\mathrm{T}} &= (1-\beta_1(\lambda),\ldots,1-\beta_r(\lambda)),\\
\mathbf{1}^{\mathrm{T}} &= (1,\ldots,1),\\
\mathbf{B} &= diag(1-\beta_1(\lambda),\ldots,1-\beta_r(\lambda)),\\
\mathbf{A} &= diag(a_1,\ldots,a_r),\ \ a_i = \frac{\lambda_i}{\lambda},\ \ \lambda = \sum_{i=1}^{r}\lambda_i,\\
\mathbf{I} &= diag(1,\ldots,1),\\
\frac{1}{\kappa_1} &= \sum_{i=1}^{r} a_i\beta_i(\lambda),\\
\frac{1}{\kappa_2} &= 1 - \kappa_1\sum_{i=1}^{r}\frac{a_i(1-\beta_i(\lambda))^2}{\beta_i(\lambda)},\\
\kappa_3 &= \sum_{i=1}^{r}\frac{a_i(1-\beta_i(\lambda))}{\beta_i(\lambda)}.
\end{aligned}
$$

Теорема 7. *В системе* $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\ Re}$ $(r < \infty)$ *стационарное распределение* $\boldsymbol{p}_k^{\mathrm{T}}$, $k \geqslant 1$, *общего числа заявок в системе имеет модифицированное*

[34]Если разерность вектора или матрицы не указана явно, то она без труда определяется из контекста.

геометрическое распределение:

$$\mathbf{p}_k^{\mathrm{T}} = \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{B} + \kappa_1 \mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{k-1}, \ k \geqslant 1, \tag{1.35}$$

$$\mathbf{p}_1^{\mathrm{T}} = P_0 \kappa_1 \mathbf{b}^{\mathrm{T}}\mathbf{A}, \tag{1.36}$$

где стационарная вероятность P_0 отсутствия заявок в системе определяется из условия нормировки и равна $P_0 = 1 - \kappa_3$.

Доказательство. Выпишем уравнения, которым удовлетворяют стационарные плотности $p_k^{(i)}(x)$. Начнем со случая $k = 1$. Выберем любое $1 \leqslant i \leqslant r$. Рассмотрим моменты времени t и $t + \Delta$ и воспользуемся основным свойством рассматриваемой системы (см. *Теорему 1*). Для того чтобы в момент времени $t + \Delta$ в системе находилась одна заявка i-го потока длины x, нужно, чтобы произошло одно из следующих событий:

- в момент t в системе не было заявок и за время Δ поступила заявка i-го потока длины x;
- в момент t в системе находилась одна заявка i-го потока остаточной длины $x + \Delta$ и за время Δ в систему не поступили новые заявки;
- в момент t в системе находилась одна заявка и за время Δ поступила заявка i-го типа длины x.

Вероятности других событий равны $o(\Delta)$. Применяя формулу полной вероятности, имеем

$$p_1^{(i)}(t + \Delta; x) = \lambda_i \Delta P_0 b_i(x) + (1 - \lambda\Delta) p_1^{(i)}(t; x + \Delta) + \lambda_i \Delta b_i(x) P_1(t),$$

Перенося слагаемое $p_1^{(i)}(t; x + \Delta)$ в левую часть равенства, деля на Δ, устремляя Δ к нулю и учитывая стационарный режим функционирования системы, получаем уравнение

$$-\frac{d}{dx} p_1^{(i)}(x) = -\lambda p_1^{(i)}(x) + \lambda_i P_0 b_i(x) + \lambda_i P_1 b_i(x),$$

решение которого, с учетом граничного условия $p_1^{(i)}(\infty) = 0$, имеет вид

$$p_1^{(i)}(x) = (P_0 + P_1) \int_x^\infty e^{-\lambda(t-x)} \lambda_i b_i(t) dt, \ 1 \leqslant i \leqslant r. \tag{1.37}$$

Интегрирование последнего соотношения дает $P_1^{(i)} = (P_0 + P_1) a_i (1 - \beta_i(\lambda))$, откуда, просуммировав по всем i, с учетом равенства $\sum_{i=1}^r P_1^{(i)} = P_1$, получаем (1.36).

Для плотностей $p_k^{(i)}(x)$ при $k \geqslant 2$, по аналогии с тем, как это было сделано выше, получаем систему линейных однородных дифференциальных уравнений

$$-\frac{d}{dx}p_k^{(i)}(x) = -\lambda p_k^{(i)}(x) + \lambda P_{k-1}^{(i)} b_i(x) + \lambda_i P_k b_i(x),$$

решение которой, с учетом граничных условий $p_k^{(i)}(\infty) = 0$, имеет вид

$$p_k^{(i)}(x) = (\lambda P_{k-1}^{(i)} + \lambda_i P_k) \int_x^\infty e^{-\lambda(t-x)} b_i(t)dt, \ 1 \leqslant i \leqslant r, \ k \geqslant 2. \tag{1.38}$$

В формуле для $p_k^{(i)}(x)$ неизвестными остаются $P_{k-1}^{(i)}$ и P_k. Для их нахождения воспользуемся аппаратом матрично–аналитических методов [133; 134; 388–395]. Интегрирование (1.38) приводит к соотношению $P_k^{(i)} = (P_{k-1}^{(i)} + a_i P_k)(1 - \beta_i(\lambda))$, которое можно переписать в матричном виде, с учетом введенных перед формулировкой теоремы обозначений, как

$$\mathbf{p}_k^{\mathrm{T}} = \mathbf{p}_{k-1}^{\mathrm{T}}\mathbf{B} + \mathbf{p}_k^{\mathrm{T}}\mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A}, \ k \geqslant 2,$$

или

$$\mathbf{p}_k^{\mathrm{T}}(\mathbf{I} - \mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A}) = \mathbf{p}_{k-1}^{\mathrm{T}}\mathbf{B}. \tag{1.39}$$

Поскольку $1 - \mathbf{b}^{\mathrm{T}}\mathbf{A}\mathbf{1} = \sum_{i=1}^r a_i\beta_i(\lambda) = \frac{1}{\kappa_1} \neq 0$ при любом положительном конечном λ, то существует обратная матрица $(\mathbf{I} - \mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1}$, вид которой дает известная формула Шермана–Моррисона (см., например, соотн. (2) в [396] или [397, С. 121]):

$$(\mathbf{I} - \mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1} = \mathbf{I} + \kappa_1\mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A}.$$

Таким образом из (1.39) немедленно следует (1.35).

Для нахождения $\mathbf{p}_1^{\mathrm{T}}$ осталось воспользоваться условием нормировки $P_0 + \sum_{k=1}^\infty \mathbf{p}_k^{\mathrm{T}}\mathbf{1} = 1$. Однако прежде необходимо доказать, что максимальное собственное значение матрицы $\mathbf{B} + \kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}$ меньше единицы. Воспользуемся[35] известным результатом (см., например, [399, Proposition 5.5.3] или [400, Example 7.10.3]) о том, что максимальное собственное значение неотрицательной квадратной матрицы (пусть $\mathbf{H}$) не превосходит $a > 0$ тогда и только тогда, когда $(a\mathbf{I} - \mathbf{H})^{-1}$ существует и $(a\mathbf{I} - \mathbf{H})^{-1} \geqslant 0$. Поскольку матрица $(\mathbf{I} - \mathbf{B})$ обратима и

$$1 - \kappa_1\mathbf{b}^{\mathrm{T}}\mathbf{A}(\mathbf{I} - \mathbf{B})^{-1}\mathbf{b} = 1 - \kappa_1 \sum_{i=1}^r \frac{a_i(1 - \beta_i(\lambda))^2}{\beta_i(\lambda)} = \frac{1}{\kappa_2} \neq 0,$$

[35] Для этого можно воспользоваться и более общим результатом — законом инерции Сильвестра (законом инерции квадратичных форм), см. [398] и комментарии после доказательства теоремы.

то выполнены все условия для того, чтобы применить формулу Шермана–Морисона к матрице $(\mathbf{I}-\mathbf{B}-\kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1}$. Имеем:

$$(\mathbf{I}-\mathbf{B}-\kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1}=(\mathbf{I}-\mathbf{B})^{-1}+\kappa_1\kappa_2(\mathbf{I}-\mathbf{B})^{-1}\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}(\mathbf{I}-\mathbf{B})^{-1}.$$

Т.к. матрица $(\mathbf{I}-\mathbf{B})^{-1}$ неотрицательна, $\kappa_1\kappa_2=\frac{1}{1-\kappa_3}$ и $\kappa_3>0$ по определению, то для неотрицательности матрицы $(\mathbf{I}-\mathbf{B}-\kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1}$ достаточно, чтобы $1-\kappa_3>0$. Таким образом, при выполнении условия[36] $\kappa_3<1$, из (1.35), просуммировав по всем возможным k, получаем

$$\begin{aligned}\sum_{k=1}^{\infty}\mathbf{p}_k^{\mathrm{T}}\mathbf{1}&=\mathbf{p}_1^{\mathrm{T}}(\mathbf{I}-\mathbf{B}-\kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{1}=\\&=\mathbf{p}_1^{\mathrm{T}}\left((\mathbf{I}-\mathbf{B})^{-1}+\kappa_1\kappa_2(\mathbf{I}-\mathbf{B})^{-1}\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}(\mathbf{I}-\mathbf{B})^{-1}\right)\mathbf{1}=\\&=\mathbf{p}_1^{\mathrm{T}}(\mathbf{I}-\mathbf{B})^{-1}(\mathbf{I}+\kappa_1\kappa_2\kappa_3\mathbf{B})\mathbf{1}.\end{aligned}$$

Подставляя сюда явный вид $\mathbf{p}_1^{\mathrm{T}}$ из (1.36), находим

$$\sum_{k=1}^{\infty}\mathbf{p}_k^{\mathrm{T}}\mathbf{1}=P_0\left(\kappa_1\kappa_3+\kappa_1\kappa_2\kappa_3\left(1-\frac{1}{\kappa_2}\right)\right).$$

Отсюда, учитывая равенство $\kappa_1\kappa_2=\frac{1}{1-\kappa_3}$ и условие нормировки, приходим к искомому виду стационарной вероятности P_0 отсутствия заявок в системе.

□

Отметим, что при доказательстве *Теоремы 7* получено несколько больше, чем заявлено в ее формулировке: формулы (1.37) и (1.38) позволяют рассчитывать совместное стационарное распределение общего числа заявок в системе, тип заявки на приборе и ее остаточное время обслуживания. Отсюда, в силу свойства PASTA пуассоновского потока и свойства дисциплины LIFO Re, остается один шаг до совместного стационарного распределения, включающего остаточные длины всех находящихся в системе заявок.

Прежде, чем переходить к изучению стационарных временных характеристик остановимся на вопросе существования и расчета моментов стационарного распределения общего числа заявок в системе. Подразумевая под матрицей $\mathbf{H}^n$ матрицу $\mathbf{H}$, каждый элемент которой возведен в n-ю степень, можем записать

[36] Далее будет показано, что это условие является необходимым и достаточным для существования стационарного режима.

$\mathbf{B}+\kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}=\mathbf{A}^{-\frac{1}{2}}\mathbf{D}\mathbf{A}^{\frac{1}{2}}$. Здесь $\mathbf{D}=\mathbf{B}+\kappa_1\mathbf{A}^{\frac{1}{2}}\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}^{\frac{1}{2}}$ — вещественная симметричная матрица. Задача спектрального разложения симметричной матрицы с вещественными элементами, модифицированной внешним произведением, хорошо изучена. В частности, собственные значения (с. з.) матрицы $\mathbf{D}$ — далее d_i, $1\leqslant i\leqslant r$, — все различны и являются корнями секулярного уравнения (см., например, [396, Lemma 2.1])

$$1+\kappa_1\sum_{i=1}^{r}\frac{a_i(1-\beta_i(\lambda))^2}{(1-\beta_i(\lambda))-d}=0. \tag{1.40}$$

Значения d_i могут быть найдены из (1.40) только численно[37]; для их расчета пригоден метод деления отрезка пополам, а также другие, специальные методы (см., например, [402; 403]).

Поскольку $1-\beta_1(\lambda)>0$ и, как известно из общей теории, корни секулярного уравнения (1.40) упорядочены следующим образом

$$1-\beta_1(\lambda)<d_1<1-\beta_2(\lambda)<d_2<\cdots<1-\beta_r(\lambda)<d_r,$$

то все d_i положительны (и различны). Выясним, при каком условии максимальное с. з. d_r матрицы $\mathbf{D}$ меньше единицы. Очевидно, так будет тогда и только тогда, когда отрицательны все с. з. матрицы $-\mathbf{I}+\mathbf{D}$. Теперь заметим, что

$$-\mathbf{I}+\mathbf{D}=(\mathbf{I}-\mathbf{B})^{\frac{1}{2}}(-\mathbf{I}+\underbrace{\kappa_1(\mathbf{I}-\mathbf{B})^{-\frac{1}{2}}\mathbf{A}^{\frac{1}{2}}\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}^{\frac{1}{2}}(\mathbf{I}-\mathbf{B})^{-\frac{1}{2}}}_{=\mathbf{D}^c})(\mathbf{I}-\mathbf{B})^{\frac{1}{2}}=$$
$$=(\mathbf{I}-\mathbf{B})^{\frac{1}{2}}(-\mathbf{I}+\mathbf{D}^c)(\mathbf{I}-\mathbf{B})^{\frac{1}{2}},$$

т. е. матрицы $-\mathbf{I}+\mathbf{D}$ и $-\mathbf{I}+\mathbf{D}^c$ связаны преобразованием подобия. По закону инерции Сильвестра[38], если все с. з. матрицы $-\mathbf{I}+\mathbf{D}$ отрицательны, то отрицательны и все с. з. матрицы $-\mathbf{I}+\mathbf{D}^c$. Но $\mathbf{I}$ — диагональная матрица, а матрица $\mathbf{D}^c$ имеет ранг 1. Поэтому у матрицы $-\mathbf{I}+\mathbf{D}^c$ всего два с. з.: -1 кратности $r-1$ и $-1+tr(\mathbf{D}^c)$ кратности 1. Прямыми вычислениями нетрудно убедиться, что след $tr(\mathbf{D}^c)$ матрицы $\mathbf{D}^c$ равен κ_3. Таким образом, при $\kappa_3<1$ макимальное с. з. d_r матрицы $\mathbf{D}$ действительно меньше единицы.

[37] Здесь полезным может быть соотношение между с. з. матриц $\mathbf{B}$ и $\mathbf{D}$ [401, С. 98]: $d_i=1-\beta_i(\lambda)+\alpha_i\kappa_1\sum_{i=1}^{r}a_i(1-\beta_i(\lambda))^2$, где $\alpha_i\in(0,1)$ и $\sum_{i=1}^{r}\alpha_i=1$.

[38] Если вещественная симметричная матрица приводится вещественным конгруэнтным преобразованием к диагональному виду, то число положительных, отрицательных и нулевых элементов на диагонали не зависит от способа приведения [404, 12.92].

Из общей теории известно, что собственный вектор $\mathbf{d}_i$, отвечающий собственному значению d_i матрицы $\mathbf{D}$, имеет вид

$$\mathbf{d}_i^{\mathrm{T}} = \frac{\kappa_1^{-1}}{\sum\limits_{i=1}^{r} \frac{a_i(1-\beta_i(\lambda))^2}{((1-\beta_i(\lambda))-d_i)^2}} \left(\frac{\sqrt{a_1}(1-\beta_1(\lambda))}{(1-\beta_1(\lambda))-d_i}, \ldots, \frac{\sqrt{a_r}(1-\beta_r(\lambda))}{(1-\beta_r(\lambda))-d_i} \right). \tag{1.41}$$

Решив численно (1.40), для получения собственных векторов можно воспользоваться (1.41), подставляя вместо точных значений d_i вычисленные. Известен такой эффект, что матрица из подобным образом полученых собственных векторов может не быть орготональной. Поэтому для расчетов предпочтительны специальные алгоритмы (см., например, [396, Algorithm II]).

Положим $\widetilde{\mathbf{D}} = (\mathbf{d}_1, \ldots, \mathbf{d}_r)$ и $\widehat{\mathbf{D}} = diag(d_1, \ldots, d_r)$. Воспользовавшись спектральным разложением $\mathbf{D} = \widetilde{\mathbf{D}}\widehat{\mathbf{D}}\widetilde{\mathbf{D}}^{\mathrm{T}}$, получаем

$$\mathbf{p}_k^{\mathrm{T}} = \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{B} + \kappa_1 \mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{k-1} = \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{A}^{-\frac{1}{2}}\mathbf{D}\mathbf{A}^{\frac{1}{2}}\right)^{k-1} = \mathbf{p}_1^{\mathrm{T}}\mathbf{A}^{-\frac{1}{2}}\widetilde{\mathbf{D}}\widehat{\mathbf{D}}^{k-1}\widetilde{\mathbf{D}}^{\mathrm{T}}\mathbf{A}^{\frac{1}{2}}, k \geqslant 1.$$

Поскольку $\widehat{\mathbf{D}}^n = diag(d_1^n, \ldots, d_r^n)$ и (при $\kappa_3 < 1$) ряд $\sum_{k=1}^{\infty} k^n d_i^k$ сходится при любом $n > 0$ и $1 \leqslant i \leqslant r$, то существуют моменты всех порядков стационарного распределения общего числа заявок в системе. Например, стационарное среднее $\mathsf{E}N$ число заявок в системе в стационарном режиме имеет вид[39]

$$\begin{aligned}
\mathsf{E}N &= \sum_{k=1}^{\infty} k\mathbf{p}_k^{\mathrm{T}}\mathbf{1} = \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{I} - \mathbf{B} - \kappa_1 \mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-2} \mathbf{1} = \\
&= \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{I} - \mathbf{B} - \kappa_1 \mathbf{B}\mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-1} \left((\mathbf{I} - \mathbf{B})^{-1} + \frac{\kappa_3}{1-\kappa_3}(\mathbf{I} - \mathbf{B})^{-1}\mathbf{B} \right) \mathbf{1} = \\
&= \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{I} - \mathbf{B} - \kappa_1 \mathbf{B}\mathbf{1}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-1} \left(\mathbf{I} + \frac{1}{1-\kappa_3}(\mathbf{I} - \mathbf{B})^{-1}\mathbf{B} \right) \mathbf{1} = \qquad (1.42) \\
&= \kappa_3 + \frac{1}{1-\kappa_3} \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{(1-\beta_i(\lambda))^2}{\beta_i(\lambda)^2}. \qquad (1.43)
\end{aligned}$$

Для произвольного $n > 0$ формулу для $\mathsf{E}N^n$ в скалярном виде выписать трудно из-за необходимости расчета бесконечных сумм $\sum_{k=2}^{\infty} k^n d_i^k$. Примечательно, что $\sum_{k=1}^{\infty} k^n d_i^k (1 - d_i)$ есть n-й момент числа заявок в системе $M \,|\, M \,|\, 1 \,|\, \infty \,|\, \mathsf{FIFO}$ с загрузкой d_i и, значит, значения сумм могут быть вычислены через производные производящей функции $\frac{(zd_i)^2}{1-d_i z}$ в точке $z = 1$ или, как уже было указано в предыдущем параграфе, рекуррентно (см. [352]).

[39] При переходе от второй к третьей строке использовано легко проверяемое соотношение $(\mathbf{I} - \mathbf{B})^{-1} = \mathbf{I} + (\mathbf{I} - \mathbf{B})^{-1}\mathbf{B}$.

Обозачим через $\varphi_i(s)$ ПЛС стационарного распределения времени пребывания в системе заявки i-го типа, через $\psi_i(s)$ — ПЛС распределения времени пребывания только что поступившей на прибор заявки i-го типа, и через $u_i(s)$ — ПЛС распределения периода занятости системы, открываемого заявкой i-го типа. Положим $\vec{\varphi}(s)^{\mathrm{T}} = (\varphi_1(s), \ldots, \varphi_r(s))$, $\vec{\psi}(s)^{\mathrm{T}} = (\psi_1(s), \ldots, \psi_r(s))$ и $\mathbf{u}(s)^{\mathrm{T}} = (u_1(s), \ldots, u_r(s))$.

Теорема 8. *В СМО $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\,$ LIFO Re ПЛС $\psi_i(s)$ распределение времени пребывания только что поступившей на прибор заявки i-го типа равно*

$$\psi_i(s) = \frac{\beta_i(\lambda + s)(\lambda + s)}{s + \lambda\beta_i(\lambda + s)}; \tag{1.44}$$

ПЛС $u_i(s)$ периода занятости, открываемого заявкой i-го типа вычисляется по формуле

$$u_i(s) = \frac{(\lambda + s)\beta_i(s + \lambda)}{\lambda + s - \lambda u(s)(1 - \beta_i(s + \lambda))}, \tag{1.45}$$

где $u(s) = \sum_{i=1}^{r}(\lambda_i/\lambda)u_i(s)$ — ПЛС периода занятости, открываемого заявкой произвольного типа;
ПЛС $\varphi_i(s)$ стационарного распределения времени пребывания заявки i-го типа в системе совпадает с i-й компонентой вектора

$$\vec{\varphi}(s)^{\mathrm{T}} = P_0\vec{\psi}(s)^{\mathrm{T}} + \mathbf{p}_1^{\mathrm{T}}\left(\mathbf{I} - \mathbf{B} - \kappa_1\mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-1}\mathbf{u}(s)\vec{\psi}(s)^{\mathrm{T}}. \tag{1.46}$$

Доказательство. Обозначим через $u_i(s; x)$ ПЛС стационарного распределения периода занятости системы, открываемого заявкой i-го типа длины x. На основании формулы полной вероятности и с учетом свойств дисциплиы LIFO Re, находим

$$u_i(s; x) = e^{-sx}e^{-\lambda x} + \int_0^x e^{-su}\lambda e^{-\lambda u}u_i(s)\sum_{j=1}^{r} a_j u_j(s)du$$

или, после интегрирования по всем возможным x,

$$u_i(s) = \beta_i(s + \lambda) + \lambda u_i(s)\frac{1 - \beta_i(s + \lambda)}{s + \lambda}\sum_{j=1}^{r} a_j u_j(s),\ 1 \leqslant i \leqslant r.$$

Вводя обозначения

$$\mathbf{B}(s) = diag(1 - \beta_1(s + \lambda), \ldots, 1 - \beta_r(s + \lambda)), \tag{1.47}$$

$$\mathbf{b}(s)^{\mathrm{T}} = (1 - \beta_1(s + \lambda), \ldots, 1 - \beta_r(s + \lambda)), \tag{1.48}$$

предыдущую систему уравнений можно записать в следующем матричном виде:

$$\mathbf{u}(s)^{\mathrm{T}} = \mathbf{1}^{\mathrm{T}}(\mathbf{I} - \mathbf{B}(s)) + \frac{\lambda}{\lambda + s} u(s)\mathbf{u}(s)^{\mathrm{T}}\mathbf{B}(s),$$

где $u(s) = \mathbf{u}(s)^{\mathrm{T}}\mathbf{A}\mathbf{1} = \sum_{i=1}^{r} a_i u_i(s)$. Решение этого уравнения имеет вид

$$\mathbf{u}(s)^{\mathrm{T}} = \mathbf{1}^{\mathrm{T}}(\mathbf{I} - \mathbf{B}(s)) \left(\mathbf{I} - \frac{\lambda}{\lambda + s} u(s)\mathbf{B}(s)\right)^{-1},$$

причем обратная матрица в правой части существует т. к. ее определитель $\prod_{i=1}^{r} \left(1 - \frac{\lambda}{\lambda+s} u(s)(1 - \beta_i(s + \lambda))\right)$ отличен от нуля при любом $s \geqslant 0$. Сделав элементарные преобразования убеждаемся, что

$$\mathbf{u}(s)^{\mathrm{T}} = \left(\frac{(\lambda + s)\beta_1(s + \lambda)}{\lambda + s - \lambda u(s)(1 - \beta_1(s + \lambda))}, \ldots, \frac{(\lambda + s)\beta_r(s + \lambda)}{\lambda + s - \lambda u(s)(1 - \beta_r(s + \lambda))}\right),$$

т. е. (1.45) имеет место. Неизвестной остается функция u, удовлетворяющая функциональному уравнению $u(s) = \sum_{i=1}^{r} (\lambda_i/\lambda) u_i(s)$. Как будет показано в *Теореме 9* при $\kappa_3 < 1$ его решение (удовлетворяющее необходимым свойствам) единственно при каждом $s \geqslant 0$, но (в общем случае) может быть найдено только численно.

Для доказательства (1.44) достаточно заметить, что раз попав на прибор, заявка его уже не покидает. Поэтому ПЛС времени пребывания заявки i-го типа на приборе совпадает с ПЛС времени пребывания на приборе заявки в аналогичной СМО, но с одним потоком т. е. задается формулой (1.17), в которой $\beta(\lambda)$ необходимо заменить на $\beta_i(\lambda)$.

Наконец, для ПЛС $\varphi_i(s)$ стационарного распределения времени пребывания заявки i-го типа в системе по формуле полной вероятности имеем

$$\varphi_i(s) = \psi_i(s)P_0 + \sum_{n=1}^{\infty} \sum_{j=1}^{r} P_n^{(j)} u_j(s)\psi_i(s), \ 1 \leqslant i \leqslant r.$$

Переписывая эту систему уравнений в матричной форме, с учетом введенных обозначений, и производя элементарные преобразования, приходим к (1.46).

□

Теорема 9. *Необходимое и достаточное условие существования стационарного режима системы* $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$ *имеет вид*[40]

$$0 < \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{(1-\beta_i(\lambda))}{\beta_i(\lambda)} = \kappa_3 < 1. \tag{1.49}$$

Доказательство. Из общей теории цепей Маркова известно[41], что для неприводимой и непериодической цепи Маркова необходимым и достаточным условием существования (собственного) предельного распределения является конечность среднего времени возвращения в некоторое состояние. Применим этот результат к состоянию, в котором общее число заявок в системе равно нулю.

Как было показано в *Теореме 8* ПЛС $u(s)$ периода занятости рассматриваемой системы удовлетворяет уравнению

$$u(s) = \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{(\lambda+s)\beta_i(s+\lambda)}{\lambda+s-\lambda u(s)(1-\beta_i(s+\lambda))}.$$

Традиционные рассуждения (см., например, [320, С. 77–78] или [119, С. 120–122]) показывают, что это функциональное уравнение определяет единственную функцию u, аналитическую в комплексной области $\mathrm{Re}\, s > 0$, в которой $|u(s)| < 1$. Рассмотрим функцию f двух аргументов, задаваемую следующим образом:

$$f(x,s) = \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{(\lambda+s)\beta_i(s+\lambda)}{\lambda+s-\lambda x(1-\beta_i(s+\lambda))} \quad s \geqslant 0,\ 0 \leqslant x \leqslant 1.$$

Изучим некоторые ее свойства. При фиксированном s имеем:

$$(f(x,s))'_x = \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{\lambda(\lambda+s)\beta_i(s+\lambda)(1-\beta_i(s+\lambda))}{(\lambda+s-\lambda x(1-\beta_i(s+\lambda)))^2} > 0,$$

$$(f(x,s))''_x = 2\sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{\lambda^2(\lambda+s)\beta_i(s+\lambda)(1-\beta_i(s+\lambda))^2}{(\lambda+s-\lambda x(1-\beta_i(s+\lambda)))^3} > 0.$$

[40] Таким образом, в отличие от однопотоковой системы (см. *Теорему 3*), роль загрузки в системе $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$ играет величина $\sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{(1-\beta_i(\lambda))}{\beta_i(\lambda)}$. Примечательно сравнить (1.49) с условием существования стационарного распределения в классической системе с абсолютным приоритетом, работающей по Схеме 3а (см. [119, С. 133]).

[41] Заметим, что доказательство этой теоремы можно провести и по-другому, обратившись к известным результатам для ветвящихся процессов (процессов Гальтона–Ватсона) с несколькими типами частиц [405, Гл. 11].

Значит $f(x,s)$ является строго возрастающей и строго выпуклой вниз функцией x, принимающей в точках $x = 0$ и $x = 1$ соответственно значения $f(0,s) = \sum_{i=1}^{r}(\lambda_i/\lambda)\beta_i(s+\lambda) > 0$ и $f(1,s) = \sum_{i=1}^{r}(\lambda_i/\lambda)\frac{(\lambda+s)\beta_i(s+\lambda)}{s+\lambda\beta_i(s+\lambda)} \leqslant 1$.

При фиксированном x, имеем:

$$(f(x,s))'_s = \sum_{i=1}^{r}\frac{\lambda_i}{\lambda}\frac{(\lambda+s)\beta_i'(s+\lambda)(\lambda+s-\lambda x)+\lambda x\beta_i(s+\lambda)(\beta_i(s+\lambda)-1)}{(\lambda+s-\lambda x(1-\beta_i(s+\lambda)))^2} < 0,$$

т. е. $f(x,s)$ является строго убывающей функцией s.

Рассмотрим уравнение

$$x = f(x,s). \tag{1.50}$$

При каждом $s \geqslant 0$, поскольку $f(0,s) > 0$ и $f(1,s) \leqslant 1$, оно имеет единственное решение тогда и только тогда, когда

$$(f(x,s))'_x|_{x=1} = \sum_{i=1}^{r}\frac{\lambda_i}{\lambda}\frac{(1-\beta_i(\lambda))}{\beta_i(\lambda)} = \kappa_3.$$

Предположим, что $\kappa_3 < 1$. В этом случае решение $x(s)$ уравнения (1.50) при каждом $s \geqslant 0$ удовлетворяет неравенствам $0 < x(s) \leqslant 1$, причем $x(0) = 1$ и $x(s) \to 1$ при $s \to 0$. Последнее означает, что период занятости системы конечен с вероятностью единица. Дифференцируя (1.50) по s в точке $s = 0$ получаем уравнение для определения средней длины периода занятости $\mathsf{E}U = -x'(0)$:

$$\mathsf{E}U = -\sum_{i=1}^{r}\frac{\lambda_i}{\lambda}\frac{\lambda\beta_i(\lambda)(\beta_i(\lambda)-1)-\lambda^2\beta_i(\lambda)(1-\beta_i(\lambda))\mathsf{E}U}{\lambda^2(\beta_i(\lambda)))^2},$$

откуда

$$\mathsf{E}U = \frac{1}{\lambda}\frac{\kappa_3}{1-\kappa_3}. \tag{1.51}$$

Таким образом, при $\kappa_3 < 1$ период занятости системы не только конечен с вероятностью единица, но и имеет конечное среднее значение.

Пусть теперь $\kappa_3 = 1$. В этом случае, уравнение (1.50) при каждом $s \geqslant 0$ также имеет единственное решение, скажем $x(s)$. Однако, хотя $x(s) \to 1$ при $s \to 0$, как видно из (1.51), средняя длина периода занятости равна бесконечности.

Наконец, при $\kappa_3 > 1$ уравнение (1.50) имеет два решения при $s = 0$: $x_1(0) = 1$ и $x_2(0) = x(0) < 1$. При $s > 0$ решение $x(s)$ по-прежнему единственно,

но $x(s) \to x(0)$ при $s \to 0$. Последнее означает, что с вероятностью $1 - x(0)$ период занятости системы никогда не кончится.

□

Простейшие изменения в формуле (1.46) позволяют получить выражение для ПЛС стационарного распределения времени ожидания заявкой i-го типа начала обслуживания. Как обычно, дифференцируя (1.44)—(1.46) по s необходимое число раз, можно получить моменты любых порядков основных временных характеристик. Например, ПЛС $\varphi(s)$ стацинарного распределения времени пребывания в системе заявки произвольного типа равно $\varphi(s) = \vec{\varphi}(s)^{\mathrm{T}}\mathbf{A1}$. Подразумевая под $(\mathbf{h}(s)^{\mathrm{T}})'$ вектор–строку $\mathbf{h}(s)^{\mathrm{T}}$, в котором на месте каждого элемента стоит значение его производной в точке s, формально из (1.46) имеем

$$(\varphi(s))' = P_0 \left(\vec{\psi}(s)^{\mathrm{T}}\right)' \mathbf{A1} + \\ + \mathbf{p}_1^{\mathrm{T}} \left(\mathbf{I} - \mathbf{B} - \kappa_1 \mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-1} \left((\mathbf{u}(s))'\vec{\psi}(s)^{\mathrm{T}} + \mathbf{u}(s)\left(\vec{\psi}(s)^{\mathrm{T}}\right)'\right)\mathbf{A1}.$$

Отсюда, замечая, что

$$\vec{\psi}(0)^{\mathrm{T}} = \mathbf{1}^{\mathrm{T}}, \quad \mathbf{u}(0) = \mathbf{1},$$
$$(\vec{\psi}(0)^{\mathrm{T}})' = -\frac{1}{\lambda}\mathbf{b}^{\mathrm{T}}(\mathbf{I} - \mathbf{B})^{-1},$$
$$(\mathbf{u}(0))' = -\frac{1}{\lambda}\frac{1}{1-\kappa_3}(\mathbf{I} - \mathbf{B})^{-1}\mathbf{b},$$

следует выражение для стационарного среднего времени $\mathsf{E}V = -(\varphi(0))'$ пребывания произвольной заявки в системе:

$$\mathsf{E}V = \frac{1}{\lambda}\mathbf{p}_1^{\mathrm{T}} \left(\mathbf{I} - \mathbf{B} - \kappa_1 \mathbf{b}\mathbf{b}^{\mathrm{T}}\mathbf{A}\right)^{-1} \left(\mathbf{I} + \frac{1}{1-\kappa_3}(\mathbf{I} - \mathbf{B})^{-1}\mathbf{B}\right)\mathbf{1}. \tag{1.52}$$

В качестве другого примера приведем формулу для стационарного среднего времени $\mathsf{E}V_i$ пребывания заявки i-го типа в системе, которая аналогичным же образом находится из (1.46):

$$\mathsf{E}V_i = \frac{1 - \beta_i(\lambda)}{\lambda\beta_i(\lambda)} + \frac{1}{\lambda}\frac{1}{1-\kappa_3}\sum_{i=1}^{r}\frac{\lambda_i}{\lambda}\frac{(1-\beta_i(\lambda))^2}{\beta_i^2(\lambda)}. \tag{1.53}$$

Сравнивая (1.42), (1.51), (1.52) и (1.43), (1.53), приходим к следующим выводам:

– для немарковской многопотоковой безприоритетной неконсервативной однолинейной системы с дисциплиной LIFO Re справедлив закон Литтла, причем как для произвольной заявки, так и для заявки каждого типа[42];
– в отличие от однопотоковой системы (см. (1.23) и (1.25)), средняя длина периода занятости в многопотокой системе с дисциплиной LIFO Re уже не совпадает со средним временем пребывания в системе произвольной заявки. Это обстоятельство, как будет видно в главе 2, играет важную роль при выяснении физического смысла, который можно придать характерикам производительности рассматриваемых неконсервативных СМО.

[42]Здесь уместно упомянуть, что подобным же образом обстоит дело и в классической однолинейной СМО с дисциплиной справедливого разделения процессора $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{PS}$.

1.4 Дополнения

Обслуживание одного потока двумя идентичными приборами

Рассмотрим систему $M\,|\,GI\,|\,2\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с идентичными приборами, в которую поступает пуассоновский поток заявок интенсивности λ. Длина заявки распределена по закону $B(x)$ с плотностью $b(x) = B'(x)$ и средним $\int_0^\infty xb(x)dx < \infty$. Будем считать, что в случае нескольких приборов дисциплина $\mathsf{LIFO\,Re}$ работает следующим образом. В момент прихода очередной заявки становится известной ее длина и, если система не пуста, приостанавливается обслуживание на всех приборах. Каждой заявке, обслуживание которой было прервано, независимо от всей предыстории функционирования системы назначается новая остаточная длина в соответствии с распределением $B(x)$. Затем обслуживание возобновляется. Новая заявка (мгновенно) становится на свободный прибор, если такой имеется; в противном случае она занимает первое место в очереди. Когда остаточная длина заявки на приборе становится равной нулю, она покидает систему и на обслуживание выбирается заявка с первого места в очереди.

Введем случайный процесс $\eta(t) = (\nu(t),\xi_1(t),\ldots,\xi_{\nu(t)}(t))$, описывающий функционирование системы, как вектор длин заявок, находящихся в системе в момент t. Когда в момент t в системе находится k заявок, то $\nu(t) = k$. Координаты $\xi_1(t)$ и $\xi_2(t)$ — это остаточные времена обслуживания заявок на приборах, $\xi_3(t)$ — длина первой заявки в очереди, а $\xi_{\nu(t)}(t)$ — последней. В том случае, когда в системе отсутствуют заявки, координаты $\eta(t)$, начиная со второй, не определяются. Наконец, при $\nu(t) = 1$ координата $\xi_1(t)$ хранит остаточное время обслуживания единственной заявки на приборе. Процесс $\eta(t)$ является марковским и описывает состояние очереди и приборов в момент t. Предположим[43], что существуют стационарные вероятности

$$P_0 = \lim_{t\to\infty} \mathsf{P}\{\nu(t) = 0\},$$
$$P_k(x_1,\ldots,x_k) = \lim_{t\to\infty} \mathsf{P}\{\nu(t) = k, \xi_1(t) < x_1,\ldots,\xi_k(t) < x_k\},\quad k \geqslant 1,$$

[43] Ниже будет получено только необходимое условие существования стационарного режима.

и ограниченные непрерывные плотности вероятностей

$$p_1(x) = P_1'(x),$$
$$p_k(x_1,x_2) = \frac{\partial^2}{\partial x_1\,\partial x_2} P_k(x_1,x_2,\infty,\ldots,\infty),\ k \geqslant 2.$$

В приводимой ниже *Теореме 10* показано, что плотности $p_1(x)$ и $p_k(x_1,x_2)$ могут быть рассчитаны рекуррентным образом. Прием, позволяющий получить соответствующие формулы в идейном плане не отличается от того, что использовался для вывода рекуррентных соотношений *Теоремы 4*. Но напрямую применить его к новой системе нельзя, поскольку, допуская некоторую вольность речи, для правильной "склейки" кусков процесса $\eta(t)$ необходимо учитывать длину заявки на другом приборе — т.е. делать то, что не требовалось в однолинейной системе. Опишем конструкцию, которая расширяет область применения метода доказательства *Теоремы 4* на 2–линейные СМО с пуассоновским входящим потоком и дисциплиной LIFO Re.

Пусть в некоторый момент в систему поступила новая заявка и сразу после ее поступления в системе оказалось $n \geqslant 3$ заявок с длинами[44] $y_1,\ldots,y_n$. Обозначим через $f_n(s;x_1,x_2,y_4,\ldots,y_n|y_1,\ldots,y_n)$ ПЛС времени до момента, когда в системе впервые останется $(n-1)$ заявка и плотность вероятности того, что в тот же момент длины оставшихся в системе заявок будут равны $x_1,x_2,y_4,\ldots,y_n$. Из описания системы и свойств дисциплины LIFO Re следует, что функции f_n симметричны на паре переменных (x_1,x_2), не зависят от $y_4, y_5, \ldots, y_n$ и совпадают при $n \geqslant 3$. Воспользовавшись формулой полной вероятности, получаем уравнение для расчета условной плотности $f = f_n$, $n \geqslant 3$:

$$\begin{aligned} f(s;x_1,x_2|y_1,y_2,y_3) &= \mathbf{1}_{(y_1\leqslant y_2)} e^{-(\lambda+s)y_1}\delta\left(y_2-(y_1+x_2)\right)\delta(y_3-x_1)+ \\ &+ \mathbf{1}_{(y_1>y_2)} e^{-(\lambda+s)y_2}\delta\left(y_1-(y_2+x_1)\right)\delta(y_3-x_2)+ \\ &+ \frac{\lambda\left(1-e^{-(\lambda+s)\min(y_1,y_2)}\right)}{\lambda+s}\int_0^\infty\int_0^\infty f(s;u_1,u_2)f(s;x_1,x_2|u_1,u_2,y_3)du_1du_2, \quad (1.54) \end{aligned}$$

где $f(s;x_1,x_2) = \int_0^\infty\int_0^\infty\int_0^\infty f(s;x_1,x_2|u_1,u_2,u_3)b(u_1)b(u_2)b(u_3)du_1du_2du_3$, δ — дельта–функция Дирака, а $\mathbf{1}_{(A)}$ — индикатор множества A. Решая (1.54), получаем, что для $f(s;x_1,x_2)$ справедлива формула $f(s;x_1,x_2) = b(x_1)g(s;x_2) + b(x_2)g(s;x_1)$, $s, x_1, x_2 \geqslant 0$, в которой неизвестная неотрицательная функция g

[44] Предполагается, что на первом приборе заявка длины y_1, на втором — y_2, в очереди на первом месте заявка длины y_3, на втором — y_4 и т. д.

есть решение интегрального уравнения

$$g(s;x) = y(s;x) + \gamma(s)\int_0^\infty K(s;u,x)g(u)du, \tag{1.55}$$

в котором

$$\begin{aligned} y(s;x) &= \int_0^\infty e^{-(\lambda+s)u}b(u)b(u+x)du, \\ K(s;u,x) &= \theta(u-x)e^{-(\lambda+s)(u-x)}b(u-x) + e^{-(\lambda+s)u}b(u+x), \\ \gamma(s) &= \frac{\frac{2\lambda}{\lambda+s}\int_0^\infty\int_0^u b(u)b(v)\left(1-e^{-(\lambda+s)v}\right)dvdu}{1-\frac{2\lambda}{\lambda+s}\int_0^\infty\int_0^u\left(1-e^{-(\lambda+s)v}\right)\left(b(u)g(s;v)+b(v)g(s;u)\right)dvdu}, \end{aligned}$$

а θ — функция Хевисайда[45]. Отметим, что значение $\gamma(s)$ зависит от неизвестной функции g, и уравнение (1.55), по-видимому, не обладает хорошими особенностями, кроме одной: свободный член и ядро являются неотрицательными функциями. В некоторых частных случаях[46] решение (1.55) может быть выписано в явном виде. В общем же случае его приходится искать численно. Хорошие результаты дает итерационный метод, причем в качестве начальной итерации необходимо брать нулевое приближение. Тогда итерации будут возрастающими, что позволит контролировать сходимость к точному решению. При $s = 0$ для контроля точности можно пользоваться условием нормировки, из которого следует, что $\int_0^\infty g(0;x)dx = 1/2$.

Пусть теперь в некоторый момент в систему поступила новая заявка и сразу после ее поступления в системе оказалось две заявки с остаточными длинами y_1 и y_2. Обозначим через $f_2(s;x|y_1,y_2)$ ПЛС времени до момента, когда в системе впервые останется одна заявка, и плотность вероятности того, что в тот же момент ее остаточная длина будет равна x. Выписывая для $f_2(s;x|y_1,y_2)$ уравнение, аналогичное (1.54), и решая его, получаем, что $f_2(s;x) = 2g(s;x)$.

Введем обозначения:

$$\hat\beta(\lambda) = \lambda\int_0^\infty e^{-\lambda u}(1-B(u))^2\,du, \tag{1.56}$$

$$\bar\beta(\lambda) = 2\lambda\int_0^\infty\int_0^\infty e^{-\lambda z}(1-B(z))g(0;z+x)\,dxdz, \tag{1.57}$$

$$\tilde\beta(\lambda) = 2\lambda\int_0^\infty\int_0^\infty e^{-\lambda z}g(0;z+x)\,dxdz. \tag{1.58}$$

[45] Т. е. $\theta(x) = 1$ при $x \geqslant 0$ и $\theta(x) = 0$ иначе.

[46] Например, при $s = 0$ в случае $B(x) = 1 - e^{-\mu x}$ решение (1.55) есть $g(0;x) = \frac{1}{2}\mu e^{-\mu x}$.

Теорема 10. *В системе* $M\,|\,GI\,|\,2\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ *стационарное распределение* P_k, $k \geqslant 0$, *общего числа заявок в системе образует, начиная с* P_1, *геометрическую прогрессию:*

$$P_k = \left(\frac{\hat{\beta}(\lambda)}{1-\bar{\beta}(\lambda)}\right)^{k-1} P_1,\ k \geqslant 1, \tag{1.59}$$

$$P_1 = P_0\frac{1-\beta(\lambda)}{1-\tilde{\beta}(\lambda)}, \quad P_0 = \left(1+\frac{1-\beta(\lambda)}{1-\tilde{\beta}(\lambda)}\frac{1-\bar{\beta}(\lambda)}{1-\bar{\beta}(\lambda)-\hat{\beta}(\lambda)}\right)^{-1}; \tag{1.60}$$

маргинальные стационарные плотности вероятностей состояний $p_1(x)$ *и* $p_k(x_1,x_2)$, $k \geqslant 2$, *определяются формулами:*

$$p_k(x_1,x_2) = \int_{x_1}^{\infty} \lambda e^{-\lambda(u-x_1)}\left(b(u)b(x_2-x_1+u)P_{k-1}+f(0;u,x_2-x_1+u)P_k\right)\,du, \tag{1.61}$$

$$p_1(x) = \int_{x}^{\infty} e^{-\lambda(u-x)}\left(\lambda b(u)P_0 + 2\lambda g(0;u)P_1\right)\,du; \tag{1.62}$$

ПЛС $u(s;x)$ *периода занятости, открываемого заявкой длины* x, *имеет вид*

$$u(s;s) = e^{-(s+\lambda)x} + \left(1-e^{-(\lambda+s)x}\right)\frac{\frac{2\lambda}{\lambda+s}\int\limits_0^{\infty} e^{-(s+\lambda)u}g(s;u)dv}{1-\frac{2\lambda}{\lambda+s}\int\limits_0^{\infty}\left(1-e^{-(\lambda+s)u}\right)g(s;u)du}; \tag{1.63}$$

ПЛС $\varphi(s;x)$ *стационарного распределения времени пребывания в системе заявки длины* x *задается выражением*

$$\varphi(s;x) = P_0\psi(s;x) + (1-P_0)\psi(s;x)\int_0^{\infty} f_2(s;u)du, \tag{1.64}$$

где ПЛС $\psi(s;x)$ *распределения времени пребывания заявки длины* x *на приборе опредяется формулой*

$$\psi(s;x) = e^{-(\lambda+s)x} + \frac{\lambda\left(1-e^{-(\lambda+s)x}\right)}{\lambda+s}\psi(s), \quad \text{и} \quad \psi(s) = \frac{\beta(\lambda+s)(\lambda+s)}{s+\lambda\beta(\lambda+s)}. \tag{1.65}$$

Доказательство. Пусть $m > 1$ — произвольное целое число. Выделим для процесса $\boldsymbol{\nu}(t)$ те интервалы времени, когда число заявок в системе будет больше m, т. е. $\boldsymbol{\nu}(t) > m$. Тогда, в силу свойств дисцилины $\mathsf{LIFO\,Re}$, с того момента, как

в системе впервые появится $(m+1)$-я заявка и до того момента, как в системе снова будет m заявок, последние $(m-1)$ компонент процесса $(\xi_1(t),\ldots,\xi_{\nu(t)}(t))$ не меняются. Следовательно, если для процесса выкинуть все те интервалы времени, когда $\nu(t) > m$, и оставшиеся куски склеить, то вероятностные характеристики получившегося после склейки процесса будут одинаковыми для всех m и плотности $p_1(x)$, $p_k(x_1,x_2)$, $2 \leqslant k \leqslant m$, будут совпадать с точностью до постоянного множителя, не зависящего от k.

Положим $P_1 = \int_0^\infty p_1(u)du$, $P_k = \int_0^\infty \int_0^\infty p_k(u,v)dudv$, $k \geqslant 2$. С учетом описанного выше свойства дисциплины LIFO Re, для стационарных плотностей вероятностей состояний справедлива следующая система уравнений (Колмогорова–Чепмена):

$$-p_1'(x) = -\lambda p_1(x) + \lambda b(x) P_0 + \lambda f_2(0;x) P_1, \tag{1.66}$$

$$-\frac{\partial p_k(x_1,x_2)}{\partial x_1} - \frac{\partial p_k(x_1,x_2)}{\partial x_2} = -\lambda p_k(x_1,x_2) + \\ + \lambda b(x_1) b(x_2) P_{k-1} + \lambda f_{k+1}(0;x_1,x_2) P_k, \ k \geqslant 2, \tag{1.67}$$

с граничными условиями $p_1(\infty) = 0$, $p_k(\infty,x_2) = p_k(x_1,\infty) = p_k(\infty,\infty) = 0$, $k \geqslant 2$. Дифференциальные уравнения (1.67) с частными производными первого порядка нетрудно разрешить методом характеристик. Предположим, что решение каждого существует и единственно. Зафиксируем в (1.67) любое целое $k \geqslant 2$ и сделаем замену $p_k(x_1,x_2) = \omega(\varepsilon_1,\varepsilon_2)$. Поскольку $\frac{dx_1}{1} = \frac{dx_2}{1}$ — характеристика уравнения (1.67), ε_1 и ε_2 выберем следующим образом: $\varepsilon_1 = x_1$, $\varepsilon_2 = x_1 - x_2$. Заметим, что якобиан такого преобразования $\frac{\partial \varepsilon_2}{\partial x_2} \neq 0$. Тогда учитывая, что

$$\frac{\partial p_k(x_1,x_2)}{\partial x_1} = \frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_1}\frac{\partial \varepsilon_1}{\partial x_1} + \frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_2}\frac{\partial \varepsilon_2}{\partial x_1} = \frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_1} + \frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_2},$$
$$\frac{\partial p_k(x_1,x_2)}{\partial x_2} = 0 + \frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_2}\partial\varepsilon_2 \cdot (-1),$$

получаем из (1.67) обыкновенное дифференциальное уравнение первого порядка для функции ω:

$$-\frac{\partial \omega(\varepsilon_1,\varepsilon_2)}{\partial \varepsilon_1} = -\lambda\omega(\varepsilon_1,\varepsilon_2) + \lambda b(\varepsilon_1) b(\varepsilon_1-\varepsilon_2) P_{k-1} + \lambda f_{k+1}(0;\varepsilon_1,\varepsilon_1-\varepsilon_2) P_k.$$

Отсюда, трактуя ε_2 как параметр и вспоминая, что $f_k = f$ при $k \geqslant 3$, уже нетрудно получить следующее выражение для плотности $p_k(x_1,x_2)$ при $k \geqslant 2$:

$$p_k(x_1,x_2) = e^{\lambda x_1} \int_{x_1}^{\infty} e^{-\lambda u} \left(\lambda b(u) b(x_2 - x_1 + u) P_{k-1} + \lambda f(0;u,x_2-x_1+u) P_k\right) du, \tag{1.68}$$

где $f(0;x_1,x_2) = b(x_1)g(0;x_2) + b(x_2)g(0;x_1)$, а функция g есть решение уравнения (1.55) при $s = 0$. Для нахождения неизвестных вероятностей P_{k-1} и P_k, фигурирующих в (1.68), проинтегрируем (1.68) по всем значениям x_1 и x_2. С помощью обычных преобразований с учетом (1.56)–(1.58), получим соотношение $P_k = \hat{\beta}(\lambda)P_{k-1} + \bar{\beta}(\lambda)P_k$, из которого следует (1.59). Поступая аналогичным образом с решением уравнения (1.66), которое имеет вид

$$p_1(x) = e^{\lambda x}\int_x^\infty e^{-\lambda u}(\lambda b(u)P_0 + 2\lambda g(0;u)P_1)\,du,$$

получаем первое соотношение в (1.60). Оставшаяся неизвестной вероятность P_0, как обычно, находится из условия нормировки $\sum_{k=0}^{\infty} P_k = 1$, откуда следует второе соотношеие в (1.60).

Будем считать, что период занятости системы начинается в момент поступления заявки в пустую систему и заканчивается в тот момент, когда система впервые оказалась свободной от заявок. Тогда по формуле полной вероятности находится уравнение для ПЛС $u(s;u)$ распределения периода занятости, открываемого заявкой длины $x \geqslant 0$:

$$u(s;x) = e^{-(s+\lambda)x} + \frac{\lambda}{\lambda+s}\left(1 - e^{-(\lambda+s)x}\right)\int_0^\infty u(s;u)f_2(s;u)du.$$

Его решение находится стандартным образом и, с учетом явного вида f_2, имеет вид (1.63).

Время пребывания заявки длины x в системе, находящейся в стационарном режиме, представляет собой сумму двух независимых частей: времени ожидания начала обслуживания и собственно времени нахождения заявки на приборе. Поскольку с вероятностью $P_0 + P_1$ поступающая заявка попадает сразу на прибор, а с дополнительной вероятностью — в очередь, то ПЛС $\varphi(s;x)$ стационарного распределения времени пребывания в системе заявки длины x имеет вид

$$\varphi(s;x) = (P_0 + P_1)\psi(s;s) + (1 - P_0 - P_1)\psi(s;s)\omega(s),$$

где $\psi(s;x)$ — ПЛС распределения времени пребывания заявки длины x на приборе, $\omega(s)$ — ПЛС распределения времени, необходимого для уменьшения общего числа заявок в системе на единицу, при условии, что обслуживание

заявок на обоих приборах только началось (причем длины заявок выбираны независимо из распределения $B(x)$). Очевидно, что для заявки, поступившей на прибор, уже безразлично сколько в системе приборов и сколько еще заявок находится в системе. Таким образом, ПЛС $\psi(s;x)$ условного и ПЛС $\psi(s) = \int_0^\infty \psi(s;x)dB(x)$ безусловного распределения времени пребывания заявки на приборе, совпадает с аналогичными характеристиками однолинейной системы. Это доказывает (1.65). Замечая, что из самого определения $\omega(s)$ следует $\omega(s) = \int_0^\infty f_2(s;u)du$, получаем (1.64).

□

Всюду выше существование стационарного распределения[47] лишь предполагалось. Необходимое условие, при котором существуют $P_k > 0$ следует из только что доказанной теоремы (см. (1.60)), однако критерий по-прежнему не ясен. Результаты, полученные для одноканальной системы (см. *Теорему 3*), подсказывают, что для двухканальной системы необходимое и достаточное условие существования стационарного режима, по-видимому, не должно зависеть от моментов длины заявки какого-либо порядка, т. е. для любого распределения длины заявки при достаточно малой интенсивности λ существует стационарное распределение. Пусть $\mathsf{E}V$ — стационарное среднее время пребывания в системе произвольной заявки, а $\mathsf{E}N$ — стационарное среднее число заявок в системе. Как показывают вычислительные эксперименты, средняя длина ПЗ рассматриваемой системы равна[48] $\frac{3}{2}\mathsf{E}V$. Отсюда получаем, что, в случае справедливости формулы Литтла[49] (которую используемым здесь методом доказать не удается), условие $P_0 > 0$ является критерием существования стационарного режима.

[47] В рамках изучаемых в первой части диссертации вопросов, анализ СМО ограничен только стационарными характеристиками. Ввиду важности для задач практики и вопросов получения вероятностно-временных характеристик систем в переходном режиме, в рамках диссертации внимание было уделено развитию одного из известных в этой области исследований методов — метода логарифмической нормы [406]. Для новых классов СМО изучены вопросы построение оценок скорости сходимости к стационарному режиму, облегчающие численное решение бесконечных систем (прямых) дифференциальных уравнений Колмогорова. Результаты этого исследования, выполненного в рамках проекта 075–15–2020–799 Министерства науки и высшего образования Российской Федерации “Методы построения и моделирования сложных систем на основе интеллектуальных и суперкомпьютерных технологий, направленные на преодоление больших вызовов”, изложены в работе [407].

[48] В отличие от случая одноканальной системы, в которой стационарное среднее время пребывания заявки в системе совпадает со средней длиной ПЗ.

[49] Из которой следует, что $\mathsf{E}V = \lambda^{-1}\mathsf{E}N = (1-P_0)^2/(\lambda P_1)$.

Инверсионный порядок обслуживания с вероятностным приоритетом в системе $M\,|\,GI\,|\,1\,|\,\infty$ с групповым потоком разнородных заявок

Рассмотрим систему $M\,|\,GI\,|\,1\,|\,\infty$ на вход которой поступает групповой пуассоновский поток заявок с переменной интенсивностью λ_n, зависящей от числа заявок n, находящихся в системе. Через $B_k(x_1,\ldots,x_k)$ будем обозначать вероятность того, что в поступившей группе будет k заявок, причем первая заявка будет иметь длину меньше x_1, вторая — меньше x_2 и т. д. Будем предполагать, что функции B_k имеют непрерывные ограниченные плотности[50]

$$b_k(x_1,\ldots,x_k)=\frac{\partial^k}{\partial x_1\cdots\partial x_k}B_k(x_1,\ldots,x_k),\ k\geqslant 1,$$

а длины заявок в различных группах независимы между собой.

В системе реализована дисциплина инвесионный порядок обслуживания с вероятностным приоритетом (далее — LIFO PP), которая работает следующим образом. В момент прихода очередной группы заявок замеряется остаточное время обслуживания первой заявки из группы. Пусть она равна u. Эта длина сравнивается с остаточной длиной заявки, находящейся на обслуживании. Если оставшееся время обслуживания заявки на приборе равно v, то с вероятностью $w(u,v)$ первая заявка из группы становится на обслуживание, за ней (в очередь) становятся остальные заявки группы, затем обслуживавшаяся ранее и остальные заявки, прежде находившиеся в системе. С вероятностью $\overline{w}(u,v)=1-w(u,v)$ обслуживавшаяся ранее заявка продолжает обслуживаться на приборе, вновь поступившие заявки становятся (в очередь) за ней, затем остальные находившиеся прежде в системе заявки. Когда остаточная длина заявки на приборе становится равной нулю, она покидает систему и на обслуживание выбирается заявка с первого места в очереди. Недообслуженные заявки дообслуживаются.

Условимся кодировать описанную систему как $M_k^{[X_i]}\,|\,GI\,|\,1\,|\,\infty\,|\,$LIFO PP, где обозначение $M_k^{[X_i]}$ указывает на тот факт, что входящий поток — групповой, состоит из разнородных заявок, а его параметр зависит от числа k заявок в системе.

[50] Для сокращения записи будем также писать $B_k(dx_1,\ldots,dx_k)$ вместо $b_k(x_1,\ldots,x_k)dx_1\cdots dx_k$.

Очевидно, дисциплина $\mathsf{LIFO\,PP}$ является частным случаем введенной в параграфе 1.1 дисциплины $\mathsf{LIFO\,GPP}$, причем[51]

$$d(x,y|u,v) = \delta(x-u)\delta(y-v)w(u,v),$$
$$d^*(x,y|u,v) = \delta(x-u)\delta(y-v)\overline{w}(u,v).$$

Введем случайный процесс $\eta(t) = (\nu(t),\xi_1(t),\ldots,\xi_{\nu(t)}(t))$, описывающий функционирование системы, как вектор длин заявок, находящихся в системе в момент t. Если в момент t в системе находится k заявок, то $\nu(t) = k$, причем $\xi_1(t)$ хранит остаточное время обслуживания заявки, находящейся в этот момент на приборе, $\xi_2(t)$ — остаточное время обслуживания первой заявки в очереди, ..., $\xi_k(t)$ —последней, $(k-1)$-й заявки в очереди. Процесс $\eta(t)$ является марковским и описывает состояние очереди и прибора в момент t.

Положим

$$\begin{aligned} P_0(t) &= \mathsf{P}\{\nu(t) = 0\}, \\ P_k(t;x_1,\ldots,x_k) &= \mathsf{P}\{\nu(t) = k, \xi_1(t) < x_1,\ldots,\xi_k(t) < x_k\}, \quad k \geqslant 1, \end{aligned}$$

и введем совместные и маргинальные стационарные распределения процесса $\eta(t)$:

$$\begin{aligned} P_k(x_1,\ldots,x_k) &= \lim_{t\to\infty} P_k(t;x_1,\ldots,x_k), \\ P_k(x) &= P_k(x,\infty,\ldots,\infty), \quad P_k = P_k(\infty), \\ p_k(x_1,\ldots,x_k) &= \frac{\partial^k}{\partial x_1\cdots\partial x_k} P_k(x_1,\ldots,x_k), \\ p_k(x) &= P_k'(x). \end{aligned}$$

Относительно плотностей $p_k(x_1,\ldots,x_k)$ и $p_k(x)$ будем предполагать, что они существуют, являются ограниченными и непрерывными.

Теорема 11. *Для СМО* $M_k^{[X_i]} \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,PP}$ *маргинальные стационарные вероятности состояний определяются из рекуррентной системы*

[51] Напоним, что всюду δ обозначает дельта–функцию Дирака.

уравнений

$$-p_1'(x) = a_1(x) - \lambda_1 p_1(x) + \lambda_1 \int_0^\infty p_1(y) K(x,y)\, dy + \lambda_1 p_1(x) g_1(x), \quad (1.69)$$

$$-p_k'(x) = a_k(x) - \lambda_k p_k(x) + \lambda_k \int_0^\infty p_k(y) K(x,y)\, dy + \\ + \sum_{i=1}^{k-1} \lambda_i \left(p_i(x) g_{k,i}(x) + \int_0^\infty p_i(y) G_{k,i}(x,y)\, dy \right), \ k \geqslant 2, \quad (1.70)$$

с граничными условиями $p_k(\infty) = 0$, *где* a_k, K, g *и* $G_{k,i}$ — *некоторые явным образом выписываемые функции.*

Доказательство. Убедиться в справедливости (1.69) и (1.70) и указать явный вид функций a_k, K, g и $G_{k,i}$, можно, выписав уравнения для совместных стационарных плотностей $p_k(x_1,\ldots,x_k)$ и проинтегрировав их по $x_2,\ldots,x_k$. Для их получения воспользуемся свойством дисциплины LIFO GPP, описанном в доказательстве *Теоремы 1* и наследуемым рассматриваемой дисциплиной LIFO PP. Начнем с $p_1(x)$ и рассмотрим моменты времени t и $t+\Delta$. Для того чтобы в момент времени $t+\Delta$ в системе находилась одна заявка длины x, нужно, чтобы произошло одно из следующих событий:

- в момент t в системе находилось 0 заявок и за время Δ поступила группа заявок произвольного размера, причем заявка на последем месте в группе имела длину x;
- в момент t в системе находилась одна заявка остаточной длины $x+\Delta$ и за время Δ новая группа не поступила в систему;
- в момент t в системе находилась одна заявка остаточной длины $x+\Delta$, за время Δ поступила группа заявок размера i, $i \geqslant 1$, первая заявка в группе имела длину $y_1,\ldots,$ последняя заявка имела длину y_i, и первая заявка в группе заняла прибор (с вероятностью $w(y_1,x)$);
- в момент t в системе находилась одна заявка, за время Δ поступила одна заявка длины x, которая встала в очередь;
- в момент t в системе находилась одна заявка остаточной длины y_1, за время Δ поступила группа заявок размера i, $i \geqslant 2$, первая заявка в группе имела длину $y_2,\ldots,$ предпоследняя заявка имела длину y_k, а по-

следняя — длину x, и группа целиком вставлв в очередь (с вероятностью $\overline{w}(y_2,y_1)$).

Вероятности других событий равны $o(\Delta)$. Применяя формулу полной вероятности, имеем

$$p_1(t+\Delta;x) = \lambda_0\Delta P_0(t)\left(b_1(x) + \sum_{i=2}^{\infty}\int\limits_{y_1,\ldots,y_{i-1}>0}\ldots\int b_i(y_1,\ldots,y_{i-1},x)\,dy_1\cdots dy_{i-1}\right)+$$
$$+(1-\lambda_1\Delta p_1(t;x+\Delta)) + \lambda_1\Delta\int\limits_0^{\infty} p_1(t;y)b_1(x)\overline{w}(x,y)\,dy+$$
$$\lambda_1\Delta p_1(t;x+\Delta)\sum_{i=1}^{\infty}\int\limits_{y_1,\ldots,y_i>0}\ldots\int w(y_1,x)B_i(dy_1,\ldots,dy_i)+$$
$$+\lambda_1\Delta\sum_{i=2}^{\infty}\int\limits_{y_1,\ldots,y_i>0}\ldots\int p_1(t;y_1)b_i(y_2,\ldots,y_i,x)\overline{w}(y_2,y_1)\,dy_1\cdots dy_i,$$

откуда, перенося слагаемое $p_1(t;x)$ в левую часть равенства, деля на Δ, устремляя Δ к нулю и учитывая стационарный режим функционирования системы, получаем

$$-p_1'(x) = \lambda_0 P_0\left(b_1(x) + \sum_{i=2}^{\infty}\int\limits_{y_1,\ldots,y_{i-1}>0}\ldots\int b_i(y_1,\ldots,y_{i-1},x)\,dy_1\cdots dy_{i-1}\right) - \lambda_1 p_1(x)+$$
$$+\lambda_1 p_1(x)\sum_{i=1}^{\infty}\int\limits_{y_1,\ldots,y_i>0}\ldots\int w(y_1,x)B_i(dy_1,\ldots,dy_i) + \lambda_1\int\limits_0^{\infty} p_1(y)b_1(x)\overline{w}(x,y)\,dy+$$
$$+\lambda_1\sum_{i=2}^{\infty}\int\limits_{y_1,\ldots,y_i>0}\ldots\int p_1(y_1)b_i(y_2,\ldots,y_i,x)\overline{w}(y_2,y_1)\,dy_1\cdots dy_i. \quad (1.71)$$

Уравнение для $p_k(x_1,\ldots,x_k)$ получается аналогичным образом, но несколько сложнее. Рассмотрим, как и выше, моменты времени t и $t+\Delta$. Для того чтобы в момент времени $t+\Delta$ в системе находилось k, $k \geqslant 2$, заявок, причём на приборе заявка длины x_1, а в очереди заявки длин $x_2,\ldots,x_k$, нужно, чтобы произошло одно из следующих событий:

– в момент t в системе находилось 0 заявок и за время Δ поступила группа заявок размера k, причем первая заявка в группе имела длину $x_1,\ldots,$ последняя заявка имела длину x_k;

- в момент t в системе находилось 0 заявок и за время Δ поступила группа заявок размера i, $i \geqslant k+1$, причем первые $i-k$ заявок в группе имели произвольные длины, а другие k заявок по порядку — длины $x_1, \ldots, x_k$ соответственно;
- в момент t в системе находилось i, $1 \leqslant i \leqslant k-1$, заявок, причём заявка на приборе имела длину x_{k-i+1}, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $k-i$ заявок, причем первая заявка в группе имела длину x_1, вторая — $x_2, \ldots$, последняя заявка имела длину x_{k-i} и первая заявка из поступившей группы заняла на прибор (с вероятностью $w(x_1,x_{k-i+1})$);
- в момент t в системе находилось i, $1 \leqslant i \leqslant k-1$, заявок, причём заявка на приборе имела длину x_1, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $k-i$ заявок, причем первая заявка в группе имела длину x_2, вторая — $x_3, \ldots$, последняя заявка имела длину x_{k-i+1} и заявка на приборе продолжила обслуживаться (с вероятностью $\overline{w}(x_2,x_1)$);
- в момент t в системе находилась k заявок, причём заявка на приборе имела длину x_1, первая заявка в очереди имела длину $x_2, \ldots$, последняя заявка в очереди имела длину x_k, и за время Δ не произошло поступлений;
- в момент t в системе находилось i, $1 \leqslant i \leqslant k$, заявок, причём заявка на приборе имела длину y, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $(k-i+1)$ заявок, причем первая заявка в группе имела длину x_1, вторая — $x_2, \ldots$, последняя заявка имела длину x_{k-i+1} и заявка на приборе продолжила обслуживаться (с вероятностью $\overline{w}(x_1,y)$);
- в момент t в системе находилось i, $1 \leqslant i \leqslant k$, заявок, причём заявка на приборе имела длину x_{k-i+1}, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $(k-i+1)$ заявок, причем первая заявка в группе имела длину y, вторая — $x_1, \ldots$, последняя заявка имела длину x_{k-i}, и первая заявка из поступившей группы заняла прибор (с вероятностью $w(y,x_{k-i+1})$);

– в момент t в системе находилось i, $1 \leqslant i \leqslant k$, заявок, причём заявка на приборе имела длину y_1, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $(k - i + m)$, $m \geqslant 2$, заявок, причем первые $(m-1)$ заявок имели длины $y_2, \ldots, y_m$, а следующие — длины $x_1, x_2, \ldots, x_{k-i+1}$, и заявка на приборе продолжила обслуживаться (с вероятностью $\overline{w}(y_2, y_1)$);

– в момент t в системе находилось i, $1 \leqslant i \leqslant k$, заявок, причём заявка на приборе имела длину x_{k-i+1}, первая заявка в очереди имела длину $x_{k-i+2}, \ldots$, последняя заявка в очереди имела длину x_k, за время Δ поступила группа из $(k - i + m)$, $m \geqslant 2$, заявок, причем первые m заявок имели длины $y_1, \ldots, y_m$, а следующие — длины $x_1, x_2, \ldots, x_{k-i}$, и первая заявка из поступившей группы заняла прибор (с вероятностью $w(y_1, x_{k-i+1})$).

Вероятности других событий равны $o(\Delta)$. Применяя формулу полной вероятности и действуя стандартным образом, приходим к следующему уравнению,

справедливому при $k \geqslant 2$:

$$
\begin{aligned}
-p_k'(x_1,\ldots,x_k) = \lambda_0 P_0 \Bigg(& b_k(x_1,\ldots,x_k) + \\
& + \sum_{i=k+1}^{\infty} \int\limits_{y_1,\ldots,y_{i-k}>0} \ldots \int b_i(y_1,\ldots,y_{i-k},x_1,\ldots,x_k)\, dy_1 \cdots dy_{i-k} \Bigg) + \\
& + \sum_{i=1}^{k-1} \lambda_i w(x_1,x_{k-i+1}) b_{k-i}(x_1,\ldots,x_{k-i}) p_i(x_{k-i+1},\ldots,x_k) + \\
& + \sum_{i=1}^{k-1} \lambda_i \overline{w}(x_2,x_1) b_{k-i}(x_2,\ldots,x_{k-i+1}) p_i(x_1,x_{k-i+2},\ldots,x_k) - \lambda_k p_k(x_1,\ldots,x_k) + \\
& + \sum_{i=1}^{k} \int_0^{\infty} \lambda_i \Bigg(\overline{w}(x_1,y) b_{k-i+1}(x_1,\ldots,x_{k-i+1}) p_i(y,x_{k-i+2},\ldots,x_k) + \\
& + w(y,x_{k-i+1}) b_{k-i+1}(y,x_1,\ldots,x_{k-i}) p_i(x_{k-i+1},\ldots,x_k) \Bigg)\, dy + \\
& + \sum_{i=1}^{k} \lambda_i \sum_{m=2}^{\infty} \int\limits_{y_1,\ldots,y_m>0} \ldots \int \Bigg(\overline{w}(y_2,y_1) b_{k-i+m}(y_2,\ldots,y_m,x_1,\ldots,x_{k-i+1}) p_i(y_1,x_{k-i+2},\ldots,x_k) + \\
& + w(y_1,x_{k-i+1}) b_{k-i+m}(y_1,\ldots,y_m,x_1,\ldots,x_{k-i}) p_i(x_{k-i+1},\ldots,x_k) \Bigg) dy_1 \cdots dy_m. \quad (1.72)
\end{aligned}
$$

Граничные условия для полученной системы уравнений выводятся таким же образом, как и в *Теореме 1*, и имеют вид

$$p_1(\infty) = 0,\ p_k(\infty,x_2,\ldots,x_k) = 0,\ k \geqslant 2. \quad (1.73)$$

Введем обозначения:

$$b_{k,m}(x) = \int\limits_{y_1,\ldots,y_{k-1}>0} \ldots \int b_k(y_1,\ldots,y_{m-1},x,y_m,\ldots,y_{k-1})\, dy_1 \cdots dy_{k-1}, 1 \leqslant m \leqslant k,$$

$$b_{2,1,2}(y,x) = b_2(y,x),$$

$$b_{k,1,m}(y,x) = \int\limits_{y_1,\ldots,y_{k-2}>0} \ldots \int b_k(y,y_1,\ldots,y_{m-2},x,y_{m-1},\ldots,y_{k-2})\, dy_1 \cdots dy_{k-2}, 2 \leqslant m \leqslant k.$$

Интегрируя (1.71) и (1.72) по $x_2,\ldots,x_n$ в пределах от 0 до ∞ и учитывая введенные обозначения, получаем уравнения (1.69) и (1.70), в которых функции a_k,

K, g и $G_{k,i}$ являются неотрицательными и задаются следующими формулами:

$$a_1(x)=\lambda_0 P_0\left(b_1(x)+\sum_{k=2}^{\infty} b_{kk}(x)\right), a_k(x)=\lambda_0 P_0\left(b_{k1}(x)+\sum_{i=k+1}^{\infty} b_{i,i-k+1}(x)\right), \quad (1.74)$$

$$K(x,y) = \overline{w}(x,y)b_1(x) + \sum_{i=2}^{\infty}\int_0^{\infty} \overline{w}(z,y)b_{i,1,i}(z,x)\,dz, \quad (1.75)$$

$$g_1(x) = \int_0^{\infty} w(y,x)b_1(y)\,dy + \sum_{i=2}^{\infty}\int_0^{\infty} w(y,x)b_{i1}(y)\,dy, \quad (1.76)$$

$$g_{k,k-1}(x)=\int_0^{\infty}\overline{w}(y,x)b_1(y)\,dy, g_{k,i}(x)=\int_0^{\infty}\overline{w}(y,x)b_{k-i,1}(y)\,dy,\ 1\leqslant i\leqslant k-2, \quad (1.77)$$

$$G_{k,k-1}(x,y) = w(x,y)b_1(x), \quad (1.78)$$

$$G_{k,i}(x,y)=\sum_{m=2}^{\infty}\int_0^{\infty}\left(\overline{w}(z,y)b_{k-i+m,1,m}(z,x)+w(z,y)b_{k-i+m,1,m+1}(z,x)\right)dz+$$
$$+ w(x,y)b_{k-i,1}(x) + \overline{w}(x,y)b_{k-i+1,1}(x)+$$
$$+\int_0^{\infty} w(z,y)b_{k-i+1,1,2}(z,x)\,dz, 1\leqslant i\leqslant k-2. \quad (1.79)$$

□

Предполагая, что решения каждого из уравнений (1.69) и (1.70) единственно в классе ограниченных неотрицательных суммируемых функций, полученные в *Теореме 11* соотношения позволяют последовательно по k найти стационарное распределение системы, а через него и основные вероятностные характеристики. Если для функции $w(x,y)$ известна сепарабельная аппроксимация (см. комментарий к *Теореме 1*), то в некоторых случаях[52] (1.69) и (1.70) сводятся к системе линейных алгебраических уравнений. Отметим также, что соотношения (1.71)—(1.73) позволяют последовательно по k вычислять и совместное стационарное распределение $P_k(x_1,\ldots,x_k)$ с точностью до вероятности P_0, которая находится из условия нормировки.

Для нахождения основных стационарных временных характеристик системы введем следующие обозначения:

[52] Как, например, при выполнении приводимых ниже условий (1.84).

– $\tilde{B}(k,i,x) = B_k(\infty,\ldots,\infty,x,\infty,\ldots,\infty)$, $k \geqslant 1$, $1 \leqslant i \leqslant k$, — вероятность того, что пришла группа из k заявок и i-я заявка в группе имеет длину меньше x;
– $\bar{B}(x_1,\ldots,x_{i-1};k,i,x) = d_x B_k(x_1,\ldots,x_{i-1},x,\infty,\ldots,\infty)/d\tilde{B}(k,i,x)$ — условная вероятность[53] того, что первая заявка имеет длину меньше x_1, вторая — меньше x_2, …, $(i-1)$-я — меньше x_{i-1}, при условии, что пришла группа из k заявок, причем заявка на i-м месте имеем длину x.
– $\hat{B}(x) = \sum_{k=1}^{\infty}\sum_{i=1}^{k}\tilde{B}(k,i,x)$ — среднее число заявок длины меньше x в поступающей группе;
– $\hat{B}(k,i;x) = d_x\tilde{B}(k,i,x)/d\hat{B}(x)$, $k \geqslant 1$, $1 \leqslant i \leqslant k$, — условная вероятность того, что поступила группа из k заявок, среди них есть ровно одна заявка длины x и она находится на i-м месте, при условии, что поступила группа, в которой имеются заявки длины x.

Теорема 12. *В СМО* $M_k^{[X_i]} \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,PP}$ *ПЛС* $\omega(s;x)$ *стационарного распределения времени ожидания начала обслуживания заявки длины* x *равно*

$$\omega(s;x) = \sum_{k=1}^{\infty}\sum_{i=1}^{k}\omega_{ki}(s;x)\hat{B}(k,i;x), \tag{1.80}$$

где $\omega_{ki}(s;x)$ *— ПЛС условного стационарного распределения времени ожидания начала обслуживания заявки длины* x *при условии, что она поступила в группе из* $k \geqslant 2$ *заявок и оказалась на* i*-м месте.*

Доказательство. Обозначим через $u_k(s;x)$, $k \geqslant 1$, ПЛС распределения времени до того момента, когда в системе останется $(k-1)$ заявок при условии, что на приборе начала обслуживаться заявка длины x, и в системе находилось k заявок. Уравнение для $u_k(s;x)$ получается из следующих рассуждений: за время обслуживания заявки длины x с вероятностью $e^{-\lambda_k x}$ не поступит больше ни одной заявки, а с вероятностью $\lambda_k e^{-\lambda_k t}dt$ на интервале времени $[t,t+dt]$ может поступить группа размером $i \geqslant 1$. В первом случае ПЛС равно e^{-sx}, а во втором зависит от размера поступающей группы и того, произошла смена заявки на приборе или нет (и в каждом случае необходимо дождаться окончания обслуживания исходной заявки длины $x-t$ и i новых заявок). Рассматривая все

[53] Здесь производная понимается как производная Радона–Никодима.

возможные события и воспользовавшись свойствами ПЛС, получаем

$$u_k(s;x) = e^{-(\lambda_k+s)x} + \\ + \sum_{i=1}^{\infty} \int_0^x \lambda_k e^{-(\lambda_k+s)t}\, dt \int\limits_{y_1,\ldots,y_i>0} \ldots \int \left(w(y_1,x-t) u_k(s;x-t) \prod_{j=1}^{i} u_{k+i+1-j}(s;y_j) + \right. \\ \left. + \int\limits_{y_1,\ldots,y_i>0} \ldots \int \overline{w}(y_1,x-t) u_{k+i}(s;x-t) \prod_{j=1}^{i} u_{k+i-j}(s;y_j) \right) B_i(dy_1,\ldots,dy_i). \quad (1.81)$$

Найдем ПЛС $\omega_{k1}(s;x)$ стационарного распределения времени ожидания начала обслуживания заявки длины x при условии, что она поступила в группе размера $k \geqslant 1$ и была на первом месте в группе. Ее время ожидания равно нулю, если она застала систему свободной и если она, застав на приборе заявку длины y, заняла ее место. Если же она застала в системе n, $n \geqslant 1$, заявок, на приборе — заявку длины y и не заняла ее место, то время ожидания совпадает с ПЗ, открываемого заявкой длины y, когда в системе находится $(n+k)$ заявка т. е. $u_{n+k}(s;y)$. В терминах ПЛС имеем

$$\omega_{k1}(s;x) = P_0 + \sum_{n=1}^{\infty} \int_0^{\infty} p_n(y) \left(w(x,y) + \overline{w}(x,y) u_{n+k}(s;y) \right) dy, \ k \geqslant 1.$$

Перейдем к ПЛС времени ожидания начала обслуживания заявки длины x, поступившей в группе из k, $k \geqslant 2$, заявок и занимающей в группе i-е место $(2 \leqslant i \leqslant k)$. В случае поступления в пустую систему время ожидания совпадает с суммарной длительностью $(i-1)$-го ПЗ, первый из которых открывается заявкой длины x_1, второй — x_2 и т. д. и в терминах ПЛС равно $u_k(s;x_1) \ldots u_2(s;x_{i-1})$. Длительности соответствующих ПЗ необходимо добавить к времени ожидания, когда поступающая группа застает систему занятой. В итоге, вводя обозначение $\tilde{u}_{nk}(s;x_1,\ldots,x_{i-1}) = u_{n+k}(s;x_1) \ldots u_{n+2}(s;x_{i-1})$, выражение для ПЛС стационарного распределения времени ожидания начала обслуживания $\omega_{ki}(s;x_1,\ldots,x_{i-1},x)$ заявки длины x, поступившей в группе из k заявок и занимающей в группе i-е место, можно записать так:

$$\omega_{ki}(s;x_1,\ldots,x_{i-1},x) = P_0 \tilde{u}_{0k}(s;x_1,\ldots,x_{i-1}) + \\ + \sum_{n=1}^{\infty} \int_0^{\infty} p_n(y) \left(w(x_1,y) \tilde{u}_{nk}(s;x_1,\ldots,x_{i-1}) + \right. \\ \left. + \overline{w}(x_1,y) u_{n+k}(s;y) \tilde{u}_{n-1,k}(s;x_1,\ldots,x_{i-1}) \right) dy, k \geqslant 2, \ 2 \leqslant i \leqslant k.$$

Воспользовавшись формулой полной вероятности находим ПЛС $\omega_{ki}(s;x)$ условного стационарного распределения времени ожидания начала обслуживания заявки длины x при условии, что она поступила в группе из $k \geqslant 2$ заявок и оказалась на i-м месте:

$$\omega_{ki}(s;x) = \int_0^\infty \dots \int_0^\infty \omega_{ki}(s;x_1,\dots,x_{i-1},x)\bar{B}(dx_1,\dots,dx_{i-1};k,i,x),$$

усредняя которое по распределению $\hat{B}(k,i;x)$, получаем (1.80).

□

Очевидно, ПЛС $\omega(s)$ стационарного распределения времени ожидания начала обслуживания произвольной заявки получается путем усреднения (1.80) по распределению длины заявки т. е.

$$\omega(s) = \int_0^\infty \omega(s;x)d\hat{B}(x)(\hat{B}(\infty))^{-1}. \tag{1.82}$$

Приведенные в доказательстве *Теоремы 12* рассуждения могут быть полезны для нахождения в терминах преобразований и других стационарных временных характеристик; в частности — ПЛС условного и безусловного стационарного распределения времени пребывания заявки в системе (соответственно $\varphi(s;x)$ и $\varphi(s)$). Обозначим через $\varphi_{ki}(s;x_1,\dots,x_{i-1},x)$, $k \geqslant 1$, $1 \leqslant i \leqslant k$, ПЛС стационарного распределения времени пребывания в системе заявки длины x, поступившей в группе из k заявок и занимающей в группе i-е место. При $i = 1$ аргумент x_0 опускается, т. е. $\varphi_{k1}(s;x_0,x) = \varphi_{k1}(s;x)$. Тогда по формуле полной вероятности имеем:

$$\begin{aligned}\varphi_{ki}(s;x_1,\dots,x_{i-1},x) &= P_0\tilde{u}_{0k}(s;x_1,\dots,x_{i-1})u_1(s;x)+\\ &+\sum_{n=1}^{\infty}\int_0^\infty p_n(y)\Big(w(x_1,y)\tilde{u}_{nk}(s;x_1,\dots,x_{i-1})u_{n+1}(s;x)+\\ &+\overline{w}(x_1,y)u_{n+k}(s;y)\tilde{u}_{n-1,k}(s;x_1,\dots,x_{i-1})u_n(s;x)\Big)dy,\ k \geqslant 1,\ 1 \leqslant i \leqslant k.\end{aligned}$$

Переход к ПЛС $\varphi(s;x)$ и $\varphi(s)$ осуществляется по формулам (1.80)–(1.82). Дифференцируя полученные формулы необходимое число раз можно получать моменты соответствующих характеристик.

Теорема 13. *Необходимое и достаточное условие существования стационарного режима системы* $M_k^{[X_i]} \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,PP}$ *имеет вид* $-u'(0) < \infty$, *где*

$$u(s) = \sum_{k=1}^{\infty} \int\limits_{y_1,\ldots,y_k>0} \ldots \int \prod_{n=1}^{k} u_n(s; y_{k-n+1}) B_k(dy_1, \ldots, dy_k). \tag{1.83}$$

Доказательство. Нетрудно видеть, что (1.83) есть ПЛС периода занятости системы. Поэтому, опираясь на известный результат из общей теории цепей Маркова о том, что для неприводимой и непериодической цепи Маркова необходимым и достаточным условием существования (собственного) предельного распределения является конечность среднего времени возвращения в некоторое состояние и применяя его к состоянию 0 (т. е. когда общее число заявок в системе равно нулю), приходим к утверждению теоремы.

□

Потенциал полученных выше теоретических результатов раскрывается при рассмотрении частных случаев входящего потока. Действительно, рассмотрим групповой пуассоновский поток постоянной интенсивности, в котором длины заявок в поступающей группе не зависят друг от друга и от размера группы т. е.

$$\lambda_k = \lambda, \;\; k \geqslant 0, \quad B_k(x_1, \ldots, x_k) = c_k B(x_1) \ldots B(x_k), \;\; k \geqslant 1, \tag{1.84}$$

где $B(x)$ — ф.р. распределения времени обслуживания одной заявки на приборе, $c_k \geqslant 0$ и $\sum_{k=1}^{\infty} c_k = 1$. Необходимым и достаточным условием существования стационарного режима[54] является $\lambda \overline{c} \overline{b} < 1$, где $\overline{b} = \int_0^{\infty} x b(x) dx$ — средняя длина поступающей заявки, а $\overline{c} = \sum_{k=1}^{\infty} k c_k$ — средний размер поступающей группы заявок.

Определим теперь ПФ[55]

$$H^*(z) = \sum_{n=0}^{\infty} z^n P_n = P_0 + H(z), \quad h(z,x) = \sum_{n=1}^{\infty} z^n p_n(x), \quad C(z) = \sum_{n=1}^{\infty} z^n c_n.$$

[54] Этот результат также следует из сравнения суммарной работы в рассматриваемой системе и классической системе $M/G/1$ с групповым входящим потоком и обслуживанием в порядке поступления.

[55] Предполагая, что ряды в определениях $H(z)$ и $h(z,x)$ сходятся при $0 < z \leqslant 1$.

Умножив уравнение (1.69) на z, а (1.70) — на z^n, просуммировав и проинтегрировав с учетом граничного условия $h(z,\infty)=0$, получаем уравнение[56]

$$h(z,x)=\lambda P_0(1-B(x))\frac{z(1-C(z))}{1-z}+$$
$$+\lambda(1-B(x))H(z)\left(C(z)+c_1+\frac{z^2-C(z)}{z(1-z)}\right)-$$
$$-\lambda(1-C(z))\left(\int_x^\infty\int_0^\infty \overline{w}(t,y)h(z,y)dy\,dB(t)-\int_0^\infty\int_x^\infty \overline{w}(t,y)h(z,y)dy\,dB(t)\right)+$$
$$+\lambda\frac{C(z)-c_1z}{z}\int_x^\infty\int_0^\infty h(z,y)\left(\overline{w}(t,y)+\int_0^\infty w(u,y)dB(u)\right)dy\,dB(t). \quad (1.85)$$

Полученное соотношение позволяет хоть и численно но, не прибегая к суммированию бесконечного ряда, находить моменты стационарного распределения общего числа заявок в системе. Ограничимся описанием алгоритма расчета математического ожидания. Обозначим через ν сл.в., распределенную как общее число заявок в системе в стационарном режиме. Проинтегрируем[57] (1.85) по x от 0 до ∞ и найдем $H(z)$. Продифференцировав выражение для $(1-z)H(z)$ два раза и положив $z=1$, получим формулу для расчета среднего числа заявок $\mathsf{E}\nu$ в системе с двумя неизвестными: $h(1,x)$ и $h'(1,x)=\partial h(z,x)/\partial z|_{z=1}$. Их нахождение осуществляется в два этапа. Сначала выписывается выражение для $H(1)$, затем, подставив $z=1$ и найденное выражение для $H(1)$ в (1.85), получается интегральное уравнение для $h(1,x)$, численное решение которого можно найти, например, итерационным методом[58]. Таким же образом, но предварительно продифференцировав (1.85) по z, находится и уравнение для $h'(1,x)$. Используя метод из [318] (см. также [314]) можно показать, что необходимым условием существования стационарного среднего $\mathsf{E}N$ числа заявок в системе является

$$\overline{c}\int_0^\infty\int_0^\infty\int_x^\infty \overline{w}(t,y)(1-B(y))dy\,dB(t)dx<\infty. \quad (1.86)$$

Для (1.86) достаточно конечности среднего размера группы и существования у распределения времени обслуживания второго момента. Перейдем к

[56] В случае ординарного потока ($c_1\equiv 1$) из (1.85) следует ПФ числа заявок в системе, исследованной в [318;319].

[57] Предполагается, что операции дифференцирования, которые будут применены ниже, законны.

[58] См. комментарии к *Теореме 1*.

временным характеристикам. Поскольку в условиях (1.84) значения $u_n(s;x)$, найденные в *Теореме 12*, не зависят от n (т. е. $u_n(s;x) = u(s;x)$), то из (1.81) следует, что $u(s;x) = e^{-(\lambda+s-\lambda C(u(s)))x}$. Здесь $u(s)$ является корнем уравнения[59] $u(s) = \beta\left(\lambda + s - \lambda C(u(s))\right)$. Далее, поскольку при фиксированном i все $u_n(s;x_i)$ равны между собой, то $\omega_{ki}(s;x_1,\ldots,x_{i-1},x) = \omega_{k1}(s;x_1)u(s;x_1)\ldots u(s;x_{i-1})$. В итоге, несложными преобразованиями формула (1.80) для ПЛС $\omega(s;x)$ стационарного распределения времени ожидания начала обслуживания заявки длины x приводится к виду

$$\omega(s;x) = \frac{1}{\overline{c}}\left(\omega^*(s;x) + \frac{u(s) - C(u(s))}{u(s)(1-u(s))}\int_0^\infty \omega^*(s;y)u(s,y)dB(y)\right),$$

где

$$\omega^*(s;x) = P_0 + \int_0^\infty h(1,y)\left(w(x,y) + \overline{w}(x,y)u(s;y)\right)dy.$$

Вспоминая, что время пребывания заявки в системе складывается из времени ожидания начала обслуживания и времени пребывания на приборе, ПЛС $\varphi(s;x)$ стационарного распределения времени пребывания в системе заявки длины x равно $\varphi(s;x) = \omega(s;x)u(s;x)$. Полученные формулы уже пригодны для расчета моментов; что же касается их обращения, то здесь справедливо уже сделанное ранее замечание (см. стр. 49).

В завершение этого параграфа рассмотрим систему с так называемыми фоновыми заявками[60], раскрывающую потенциал разработанных теоретических результатов в направлении, отличном от избранного выше. В систему $M^{[X]}\,|\,GI\,|\,1\,|\,n\,|\,\mathsf{LIFO\,PP}$ $(n \leqslant \infty)$ поступают заявки двух типов. Заявки первого типа поступают группами в соответствии с пуассоновским потоком с параметром λ. Как и ранее, c_k будет обозначать вероятность наличия k заявок в группе, а $\overline{c} = \sum_{k=1}^\infty kc_k$ — средний размер группы. Заявки второго типа поступают из накопителя бесконечной емкости, и их длины независимы с ф. р. $G(x)$, имеющей плотность $g(x) = G'(x)$, и средним значением $\overline{g} = \int_0^\infty xg(x)dx < \infty$. Заявки первого типа имеют относительный приоритет перед заявками второго типа, т. е. поступление на прибор заявки второго типа происходит только в том случае, если в системе отсутствуют заявки первого типа. Прерывание обслуживания заявок второго типа не допускается. Общее число заявок первого типа в системе ограничено числом n, $n \leqslant \infty$. При $n < \infty$ считается, что поступающая группа

[59] Отметим, что ПЛС $u^*(s)$ длительности ПЗ системы в условиях (1.84) удовлетворяет уравнению $u^*(s) = C(\beta(\lambda + s - \lambda u^*(s)))$.

[60] По поводу заявок такого типа см., например, [139; 140].

теряется целиком, если в момент поступления для хотя бы одной из заявок в группе не хватает места в очереди.

Будем считать, что в случае двух потоков дисциплина LIFO PP работает следующим образом:

- если в момент поступления группы заявок первого типа в системе обслуживается заявка первого типа, то длина x первой из поступающей группы заявки сравнивается с остаточной длиной y заявки, находящейся на приборе. С вероятностью $w(y,x)$ поступающая группа занимает первые места в очереди, а заявки, находившиеся в очереди до поступления группы, становятся за ними с учетом порядка. С дополнительной вероятностью $\overline{w}(y,x) = 1 - w(y,x)$ первая заявка из поступающей группы становится на прибор, остальные заявки из поступающей группы занимают первые места в очереди, заявка с прибора встает за ними. Остальные заявки, находившиеся в очереди до поступления новой группы, становятся после этой заявки с сохранением порядка;
- если группа заявок первого типа в момент поступления застает на приборе заявку второго типа, то длина x первой из поступающей группы заявки сравнивается с длиной y заявки, стоящей на первом месте в очереди. С вероятностью $v(y,x)$ заявка длины y остается на первом месте в очереди, поступающая группа занимает места в очереди начиная со второго, а остальные заявки, имевшиеся в очереди до поступления новой группы, становятся за ними с сохранением порядка. С дополнительной вероятностью $\overline{v}(y,x) = 1 - v(y,x)$ поступающая группа заявок занимает первые места в очереди, заявка длины y становится за поступившей группой заявок, а заявки, находившиеся в очереди до момента поступления группы, становятся за ней с учетом порядка.

Останавливаясь только на случае $n = \infty$, положим:

- $q_k(t,x)$, $k \geqslant 0$, — стационарная плотность вероятности того, что на приборе обслуживается заявка второго типа длины t и в очереди находятся k заявок первого типа, причем заявка, стоящая в очереди первой, имеет длину x;
- $p_k(x,y)$, $k \geqslant 1$, — стационарная плотность вероятности того, что на приборе обслуживается заявка первого типа длины x и в очереди находятся $k-1$ заявок первого типа, причем заявка, стоящая в очереди первой, имеет длину y.

Обозначим через $P_k = \int_0^\infty \int_0^\infty p_k(x,y)dydx$, $k \geqslant 1$, стационарную вероятность того, что на приборе обслуживается заявка первого типа и в очереди находятся $k-1$ заявок первого типа. Соответственно, $Q_k = \int_0^\infty \int_0^\infty q_k(t,x)dxdt$, $k \geqslant 0$ есть стационарная вероятность того, что в системе находятся k заявок первого типа и обслуживается заявка второго типа. Определим ПФ[61]

$$P(s,x) = \sum_{k=1}^{\infty} p_k(x)s^k, \tag{1.87}$$

$$Q(s,t,x) = \sum_{k=1}^{\infty} q_k(t,x)s^k, \tag{1.88}$$

$$Q^*(s,t,z,x) = \sum_{k=1}^{\infty} q^*_{k+1}(t,z,x)s^k, \tag{1.89}$$

где $p_k(x) = \int_0^\infty p_k(x,y)dy$. Используя методы исследования, изложенные в этом параграфе и в [132; 298], были изучены[62] аналогичные характеристики для описанной системы с фоновыми заявками. В частности показано, что стационарные вероятности Q_k, $k \geqslant 0$, определяются как коэффициенты при s^k разложения в ряд по степеням s ПФ $Q(s) + Q_0$, задаваемой формулой

$$Q(s) + Q_0 = Q\frac{1 - \gamma(\lambda - \lambda C(s))}{\lambda - \lambda C(s)},$$

где $\gamma(s)$ — ПЛ плотности g в точке s, Q — (нормировочная) постоянная, а стационарные вероятности P_k, $k \geqslant 1$, — как коэффициенты при s^k разложения в ряд по степеням s ПФ $P(s) = \int_0^\infty P(s,x)dx$, где $P(s,x)$ удовлетворяет уравнению

$$-\frac{dP(s,x)}{dx} = -\lambda(1-c_1)P(s,x) + \lambda c_1 b(x)P(s) + Q(s,0,x) + \int_0^\infty Q^*(s,0,u,x)du +$$

$$+\lambda\left(C(s) - c_1\right)\int_0^\infty \left(b(u)w(x,u)P(s,x) + b(x)\overline{w}(u,x)P(s,u)\right)du +$$

$$+\lambda\left(\frac{C(s)}{s} - c_1\right)\int_0^\infty dz \int_0^\infty \left(b(u)w(z,u)P(s,z)\delta(u-x) + b(z)b(x)\overline{w}(u,z)P(s,u)\right)du,$$

[61] Предполагая, что $\lambda\overline{c}\overline{b} < 1$ и ряды в определениях сходятся при $0 < s \leqslant 1$.

[62] В предположении, что введенные плотности существуют, непрерывны и ограничены.

в котором $Q(s,t,x) = \sum_{i=1}^{\infty} A_i(x) q_i(t) \left(C(s)\right)^i$, A_i и q_i — некоторые рекурретным образом вычисляемые функции, и

$$Q^*(s,t,z,x) = Q\lambda \left(\frac{C(s)}{s} - c_1\right) b(z)b(x) \int_t^{\infty} e^{\lambda u} du \int_u^{\infty} g(y) e^{-\lambda y} dy +$$

$$+ \lambda c_1 \int_t^{\infty} \left(b(x)v(z,x)Q(s,u,z) + b(z)\overline{v}(x,z)Q(s,u,x)\right) du +$$

$$+\lambda \left(\frac{C(s)}{s} - c_1\right) \int_t^{\infty} du \int_0^{\infty} \left(b(y)v(z,y)Q(s,u,z)\delta(y-x) + b(z)b(x)\overline{v}(y,z)Q(s,u,y)\right) dy.$$

Постоянная Q определяется из условия нормировки из соотношения $Q_0 + P(1) + Q(1) = 1$. Из результатов [298] также следует, что стационарные временные характеристики могут быть найдены не только при описанном выше варианте дисциплины LIFO PP, но также и когда порядок постановки в очередь заявок первого типа в момент обслуживания заявки второго типа произвольный[63].

[63] Вместе с тем, в связи с изученными в этом параграфе СМО остаются и открытые теоретические вопросы. Например, специального исследования заслуживает случай, когда длины заявок принимают только конечное число значений. Не изучен вариант более общего группового входящего потока, когда в каждой поступающей группе могут находиться подгруппы заявок одинаковой длины. Наконец, представляет интерес обобщение разработанного метода на случай нескольких типов (с различными функциями распределения длин) заявок и других правил начала обслуживания (например, N-политики [408]).

Глава 2. Получение оценок стационарных характеристик частично наблюдаемых стохастических систем обслуживания на основе информации о прогнозных временах обслуживания

2.1 Предварительные замечания

В этой главе будем называть *частично наблюдаемой* всякую СМО, для которой выполнены следующие условия:

- для каждой поступающей заявки становится известным некоторое положительное число, которое назовем остаточным *прогнозным временем обслуживания* и которое имеет смысл работы, которую, как ожидается, необходимо совершить прибору для завершения обработки заявки[1];
- фактическое время обслуживания заявки, т. е. работа, которую в действительности необходимо совершить прибору для завершения ее обработки, фиксируется в момент поступления заявки в систему, однако *ненаблюдаемо* и не совпадает с указанным для нее прогнозным временем обслуживания;
- специальное планирование обслуживания (если оно в системе предусмотрено[2]) осуществляется только на основе остаточных прогнозных времен обслуживания.

Таким образом, говоря о вероятностно–временных характеристиках частично наблюдаемых (в указанном выше смысле) СМО, необходимо отличать их прогнозные значения, от фактических. При этом для задач практики значение имеют, вообще говоря, лишь последние[3].

[1]Отметим, что остаточное прогнозное время обслуживания уменьшается только при нахождении заявки на приборе и, вообще говоря, может стать отрицательным. Кроме того, обслуживание заявки может окончиться раньше, чем обнулится остаточное прогнозное время обслуживания.

[2]Например, оно предусмотрено при дисциплине SRPT и не предусмотрено при дисциплине FIFO. Известно и планирование другого рода; см. [409].

[3]Здесь уместно подчеркнуть связь рассматриваемого вопроса с практикой. Несовпадение того времени выполнения заявки, которое указывается в момент ее поступления и используется планировщиком очереди, с тем временем, которое по факту заняло ее обслуживание, является неотъемлемой чертой некоторых современных технических систем (см., например, [36; 410–414]). В частности, как отмечается специалистами межведомственного суперкомпьютерного центра РАН [415], эта особен-

В этой главе теоретически обосновывается новый метод уточнения оценок (сверху) фактических значений стационарных характеристик частично наблюдаемых СМО, исходные параметры которых удовлетворяют определенным, продиктованным практикой условиям[4]. Он заключается в следующем. Для имеющейся частично наблюдаемой системы сначала фиксируется интересующая характеристика, стационарное распределение которой существует, и вычисляется ее значение[5]. Затем, исходя из имеющейся информации о частично наблюдаемой системе, выбирается СМО с некоторой разновидностью дисциплины LIFO GPP, в которой значение искомой (или, возможно, другой) характеристики лучше рассчитанного прогнозного значения и близко к (неизвестному!) фактическому.

Ясно, что приблизиться к фактическому значению, не зная его, можно не всегда. Условие, при котором предложенный метод позволяет это сделать, формулируется ниже следующим образом: в случае ненаблюдаемости системы известного класса (далее он обозначается $\mathfrak{M}$) возможно получение оценок (сверху) фактических значений некоторых ее стационарных характеристик, если она принадлежит его определенному подмножеству (далее оно обозначается $\mathfrak{M}^*$). Задача исчерпывающего описания $\mathfrak{M}^*$, т. е. нахождение границ применимости метода, остается нерешенной. Однако некоторые аналитические соображения (см. *Теорему 18*) подсказывают условия[6], при выполнении которых метод, по-видимому, работает для многих частично наблюдаемых СМО. Вычислительные эксперименты подтверждают этот вывод.

Прежде чем переходить к основному содержанию главы, напомним несколько определений.

ность типична для современных суперкомпьютерных систем коллективного пользования и снижает эффективность расписаний запусков заявок.

[4]Примером одного из таких условий является принадлежность прогнозных времен обслуживания классу сл. в. с убывающей функцией интенсивности (УФИ). Напомним, что принадлежность классу УФИ означает, что вероятность окончания обслуживания на промежутке $[t,t+x]$ при условии, что обслуживание заявки не окончилось до момента t, не возрастает с ростом t при фиксированном x.

[5]Это значение, являясь лишь прогнозным, может быть как больше, так и меньше фактического.

[6]Это условия на загрузку системы и на распределение(я) прогнозных времен обслуживания.

Определение[7]. Для двух сл. в. X и Y с ф. р. $F(x)$ и $G(x)$ выполняется соотношение $X \overset{d}{\leqslant} Y$, если для всех вещественных x выполняется неравенство $F(x) \geqslant G(x)$.

В литературе встречаются и другие обозначения для $\overset{d}{\leqslant}$; например, $\leqslant_{st}$, $\leqslant_{d}$, $\overset{(1)}{\leqslant}$. Обычно случайную величину X, в случае выполнения $X \overset{d}{\leqslant} Y$, называют стохастически меньшей, чем Y.

Определение[8]. Положительная сл. в. X с ф. р. $F(x)$ принадлежит классу ГНСХИ (гармоничное новое в среднем хуже использованного), если для всех $x \geqslant 0$

$$\int_x^\infty (1 - F(u))du \geqslant \mathsf{E}X\, e^{-\frac{x}{\mathsf{E}X}}.$$

Определение[9]. Положительная сл. в. X с ф. р. $F(x)$ принадлежит классу $\mathscr{L}$, если для всех $s \geqslant 0$

$$\int_0^\infty e^{-su}(1 - F(u))du \geqslant \frac{\mathsf{E}X}{1 + s\mathsf{E}X}. \tag{2.1}$$

Если же в (2.1) выполняется обратное (но опять же нестрогое) неравенство, то будем говорить, что сл. в. X принадлежит классу $\bar{\mathscr{L}}$. Класс $\bar{\mathscr{L}}$ строго "больше" класса ГНСХИ (см. [419, С. 617]) и включает экспоненциально распределенные сл. в. в качестве граничных (для них выполняется точное равенство). Напоним, что, если сл. в. X принадлежит классу $\bar{\mathscr{L}}$, то ее коэффициент вариации не меньше единицы (см., например, [420;421]); при этом уже третий момент может не существовать[10] (см. [424, Example 2.1]).

Определение[11]. Будем говорить, что положительная сл. в. X имеет лог–симметричное распределение с параметром $\sigma > 0$, если $X = e^Y$ и

[7] См., например, [416, С. 16].

[8] См., например, [417;418] или [60, С. 103].

[9] См., например, [419].

[10] Таким образом, сл. в. X может быть "тяжелохвостой" в следующем смысле: если $B(x)$ ее ф. р., то $\lim_{x\to\infty} e^{\varepsilon x}(1 - B(x)) = \infty$ при всех $\varepsilon > 0$ (см., например, определение 2.4 и раздел 2.2 в [422], и [423]).

[11] См., например, [425].

сл. в. Y имеет симметричное (относительно нуля) распределение с плотностью $g(x) = \alpha\left(\frac{x^2}{\sqrt{\sigma}}\right)$, $x \in (-\infty, \infty)$, где α — некоторая положительная при $x > 0$ функция, для которой $\int_0^\infty \sqrt{u}\,\alpha(u)du = 1$.

Из этого определения следует, что сл. в. X и $1/X$ одинаково распределены и $\mathsf{E}X^r \geqslant 1$, если момент r-го порядка существует. Отметим также, что ряд известных распределений относятся к классу лог–симметричных: логнормальное[12], лог–лапласовское, Бирнбаума–Сандерса и др. (см., например, [426, С. 199]).

Обозначим через $\mathfrak{M}$ множество, состоящее из всех возможных СМО типа

$$\Sigma \,|\, B_r(x), r \in \mathcal{R} \,|\, c \,|\, n \,|\, \mathcal{U}, \tag{2.2}$$

где Σ — суммарный входящий поток, $B_r(x)$ — распределение длины заявки типа r, $\mathcal{R}$ — конечное множество типов заявок, c — число идентичных приборов, n — емкость очереди, $\mathcal{U}$ — дисциплина выбора из очереди и предоставления обслуживания. Считается, что обслуживающие приборы не порятятся и способны немедленно после окончания обслуживания одной заявки приступить к обслуживанию следующей. Кроме того, допускается прерывание обслуживания и механизм прерывания не меняет длины заявки. Таким образом, в $\mathfrak{M}$ содержатся только “консервативные” СМО.

Далее, говоря о вероятностно–временных характеристиках, будем иногда явно указывать их зависимость от ф. р. длин заявок; например, если сл. в. Q — длина очереди в СМО (2.2), то сл. в. $Q_{B_1,\ldots,B_{|\mathcal{R}|}}$ имеет тот же смысл.

Обозначим через $\mathfrak{M}^*$ подмножество $\mathfrak{M}$, для элементов которого выполняются следующие условия:

– найдется вероятностно–временная характеристика[13], стационарное распределение которой существует, и известны достаточные условия его существования. Пусть X — сл. в., имеющая это распределение;
– найдутся такие два набора $\{B_r(x), r \in \mathcal{R}\}$ и $\{\hat{B}_r(x), r \in \mathcal{R}\}$ ф. р. длин заявок, что справедливо неравенство

$$X_{B_1,\ldots,B_{|\mathcal{R}|}} \overset{d}{\leqslant} X_{\hat{B}_1,\ldots,\hat{B}_{|\mathcal{R}|}};$$

[12] С параметрами 0 и σ, если положить $\alpha(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}u}$.

[13] Например, длина очереди, время ожидания начала обслуживания, время пребывания заявки в системе и т. п.

– найдется вероятностно–временная характеристика, скажем Y, функционирующей в стационарном режие СМО с тем же входящим потоком, набором ф. р. $\{\hat{B}_r(x), r \in \mathcal{R}\}$, возможно другим числом приборов и емкостью очереди, и некоторым вариантом дисциплины $\mathsf{LIFO\,GPP}$, для которой выполняются соотношения

$$X_{B_1,\ldots,B_{|\mathcal{R}|}} \overset{d}{\leqslant} Y_{\hat{B}_1,\ldots,\hat{B}_{|\mathcal{R}|}} \overset{d}{\leqslant} X_{\hat{B}_1,\ldots,\hat{B}_{|\mathcal{R}|}}. \tag{2.3}$$

Принадлежность СМО множеству $\mathfrak{M}^*$ фактически означает, что, в случае ее частичной наблюдаемости, с помощью описанного в начале параграфа метода можно получать оценки (сверху) фактических значений ее стационарных характеристик.

2.2 Оценки для систем с дисциплиной справедливого разделения процессора

Покажем, что $\mathfrak{M}^*$ — непустое множество. Рассмотрим систему $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ с интенсивностью входящего потока λ, в которой длины заявок $\hat{S}$ имеют абсолютно непрерывное[14] распределение $\hat{B}(x)$ (с плотностью $\hat{b}(x)$) и конечное среднее $\mathsf{E}\hat{S}$. Пусть сл. в. N^{PS} имеет распределение, совпадающее со стационарным распределением общего числа заявок в этой СМО. Как известно [75, С. 61], оно существует при $\lambda\mathsf{E}\hat{S} < 1$. Теперь рассмотрим систему $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,GPP}$ с тем же входящим потоком, той же ф. р. длин заявок $\hat{B}(x)$ и дисциплиной $\mathsf{LIFO\,GPP}$, в которой $D(x,y|u,v) = \hat{B}(x)\hat{B}(y)$, а остальные определяющие дисциплину функции тождественно равны нулю. Сл. в., имеющую стационарное распределение общего числа заявок в этой СМО, которое существует (согласно *Теореме 3*) при $\frac{1}{2} < \hat{\beta}(\lambda) = \int_0^\infty e^{-\lambda u} d\hat{B}(u) < 1$, обозначим через $N^{\mathsf{LIFO\,Re}}$.

Далее до конца параграфа (если явно не указано иное) через $B(x)$ и $b(x)$ обозначаются соответственно ф. р. и плотность сл. в. S, через $l(x)$ и $g(x)$ — плотности соответственно лог–симметричной сл. в. $X = e^Y$ и сл. в. Y.

[14] Это условие необходимо, поскольку дальнейшие рассуждения основываются на результатах параграфа 1.2, полученных в предположении, что распределение длин заявок является абсолютно непрерывным. Обобщение на случай, когда отсутствует плотность возможно, но требует других рассуждений.

Теорема 14[15]**.** *Если сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$, то справедливо неравенство*

$$N_{\hat{B}}^{\mathsf{LIFO\,Re}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{PS}}. \tag{2.4}$$

Если дополнительно известно, что сл. в. $\hat{S}$ представима в виде произведения[16] *$S \cdot X$, причем сл. в. S и X независимы и имеют соответсвенно экспоненциальное и лог–симметричное распределения, то выполняются соотношения*

$$N_{B}^{\mathsf{PS}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{LIFO\,Re}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{PS}}. \tag{2.5}$$

Доказательство. Начнем с первого утверждения теоремы. Стационарные распределения общего числа заявок в обеих системах являются геометрическими. Для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, как известно (см., например, [75, С. 61]), имеет место формула

$$\mathsf{P}\{N_{\hat{B}}^{\mathsf{PS}} = k\} = \left(1 - \lambda \mathsf{E}\hat{S}\right)\left(\lambda \mathsf{E}\hat{S}\right)^{k}, \quad k = 0,1,\ldots,$$

а для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$, согласно *Теореме 4*, — формула

$$\mathsf{P}\{N_{\hat{B}}^{\mathsf{LIFO\,Re}} = k\} = \left(2 - \frac{1}{\hat{\beta}(\lambda)}\right)\left(\frac{1 - \hat{\beta}(\lambda)}{\hat{\beta}(\lambda)}\right)^{k}, \quad k = 0,1,\ldots$$

Отсюда следует, что $N_{\hat{B}}^{\mathsf{LIFO\,Re}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{PS}}$ тогда и только тогда, когда $\hat{\beta}(\lambda) \geqslant \frac{1}{1+\lambda \mathsf{E}\hat{S}}$. Вспоминая (2.1), убеждаемся в справедливости последнего неравенства.

Перейдем к доказательству второго утверждения. Для этого рассмотрим подробнее вид ПЛС $\hat{\beta}(\lambda)$, когда $\hat{S} = S \cdot X$. С учетом того, что плотность сл. в. X была обозначена выше через $l(x)$, имеем

$$\hat{\beta}(\lambda) = \mathsf{E}\left(e^{-\lambda SX}\right) = \int_0^\infty \mathsf{E}\left(e^{-\lambda uS}\right) l(u)du = \int_0^\infty \frac{1/\mathsf{E}S}{1/\mathsf{E}S + \lambda u} l(u)du.$$

[15] Отчасти эта теорема справедлива и для однолинейной СМО в дискретном времени с геометрическим входящим потоком и дисциплиной циклического обслуживания (см. [295]).

[16] В том, что эта сл. в. принадлежит классу $\bar{\mathscr{L}}$, можно убедиться, воспользовавшись неравенством Йенсена [427]:

$$\mathsf{E}e^{-s\hat{S}} = \mathsf{E}e^{-sSX} = \int_0^\infty \mathsf{E}\left(e^{-suX}\right) b(u)du \geqslant \int_0^\infty e^{-su\mathsf{E}X} b(u)du = \frac{1}{\mathsf{E}S}\int_0^\infty e^{-su\mathsf{E}X} e^{-\frac{u}{\mathsf{E}S}} du = \frac{1}{1 + s\mathsf{E}\hat{S}}.$$

Положим $\ln(u) = v$; тогда предыдущая формула, в которой надо положить $u = e^v$ и $du = e^v dv$, дает

$$\hat{\beta}(\lambda) = \int_{-\infty}^{\infty} \frac{1}{1 + \mathsf{E}Se^v}\, g(v)dv,$$

или, с учетом того, что функция g является четной, —

$$\begin{aligned}\hat{\beta}(\lambda) &= \int_0^{\infty} \left(\frac{1}{1+\lambda\mathsf{E}Se^{-v}} + \frac{1}{1+\lambda\mathsf{E}Se^{v}}\right) g(v)dv = \\ &= \int_0^{\infty} \left(\frac{2+\lambda\mathsf{E}S(e^v+e^{-v})}{1+(\lambda\mathsf{E}S)^2+\lambda\mathsf{E}S(e^{-v}+e^{v})}\right) g(v)dv = \\ &= \int_0^{\infty} \left(\frac{2+2\lambda\mathsf{E}S\cosh(v)}{1+(\lambda\mathsf{E}S)^2+2\lambda\mathsf{E}S\cosh(v)}\right) g(v)dv.\end{aligned}$$

Здесь cosh обозначает гиперболический косинус, т. е. $\cosh(x) = \frac{1}{2}(e^{-x} + e^x)$. Заметим теперь, что предыдущую формулу можно переписать в виде

$$\begin{aligned}\hat{\beta}(\lambda) &= \frac{2}{1+\lambda\mathsf{E}S} \int_0^{\infty} \left(\frac{(1+\lambda\mathsf{E}S\cosh(v))(1+\lambda\mathsf{E}S)}{1+(\lambda\mathsf{E}S)^2+2\lambda\mathsf{E}S\cosh(v)}\right) g(v)dv = \\ &= \frac{2}{1+\lambda\mathsf{E}S} \int_0^{\infty} \underbrace{\left(\frac{1+\lambda\mathsf{E}S+\cosh(v)(\lambda\mathsf{E}S+(\lambda\mathsf{E}S)^2)}{1+(\lambda\mathsf{E}S)^2+2\lambda\mathsf{E}S\cosh(v)}\right)}_{=h(v)} g(v)dv = \\ &= \frac{2}{1+\lambda\mathsf{E}S} \int_0^{\infty} h(v)g(v)dv.\end{aligned}$$

Покажем, что $0 \leqslant h(x) \leqslant 1$ при $x \geqslant 0$. Неотрицательность функции h очевидна. Далее, поскольку $\cosh(x) \geqslant 1$ при всех $x \geqslant 0$, $0 < \lambda\mathsf{E}S < 1$ и $\lambda\mathsf{E}S - (\lambda\mathsf{E}S)^2 > 0$, имеем

$$\lambda\mathsf{E}S - (\lambda\mathsf{E}S)^2 \leqslant \cosh(x)\left(\lambda\mathsf{E}S - (\lambda\mathsf{E}S)^2\right).$$

Прибавляя к левой и правой частям неравенства $1 + \lambda\mathsf{E}S\cosh(x)$, получаем

$$1 + \lambda\mathsf{E}S + \cosh(x)\left(\lambda\mathsf{E}S + (\lambda\mathsf{E}S)^2\right) \leqslant 1 + (\lambda\mathsf{E}S)^2 + 2\lambda\mathsf{E}S\cosh(x),$$

т. е. числитель определяющей функцию h дроби не превосходит знаменателя. Вспоминая теперь, что $2\int_0^{\infty} g(u)du = 1$, имеем

$$\hat{\beta}(\lambda) = \frac{2}{1+\lambda\mathsf{E}S} \int_0^{\infty} h(v)g(v)dv \leqslant \frac{2}{1+\lambda\mathsf{E}S} \int_0^{\infty} g(v)dv = \frac{1}{1+\lambda\mathsf{E}S},$$

что вместе с (2.4) доказывает (2.5).

□

Из доказательства[17] теоремы видно, что для выполнения неравенств (2.5) нет необходимости требовать, чтобы для сл. в. $\hat{S}$ выполнялось (2.1) при всех $s \geqslant 0$, а достаточно проверить (2.1) при $0 \leqslant s < \mathsf{E}\hat{S}$.

Полученный результат можно проинтерпретировать следующим образом. Пусть имеется СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, в которой времена обслуживания заявок становятся известными в момент поступления, причем их ф. р. известна; обозначим ее через $\hat{B}(x) = \mathsf{P}\{\hat{S} < x\}$. Кроме того известно, что эти времена содержат случайную ошибку X и фактическое[18] время обслуживания S каждой заявки связано с планируемым временем $\hat{S}$ соотношением $\hat{S} = S \cdot X$[19]. *Теорема 14* показывает, что достаточно знать распределения S и X (и не требуется

[17]Стоит отметить, что путь доказательства второго утверждения теоремы был найден не сразу. Так безрезультатными оказались попытки применения методов асимптотического (при $\sigma \to 0$) анализа [428] и известных общих неравенств, в том числе и для ПЛ (см., например, [429–431]).

[18]Т. е. то время, которое она действительно проведет на приборе.

[19]Это предположение, являющееся существенным для большинства результатов параграфа, взято из практики. Известен ряд исследований (см., например, работы [413; 414] и ссылки в них), в которых достаточно убедительно показывается, что в некоторых современных технических системах времена обслуживания заданий/заявок/запросов содержат мультипликативную ошибку. В теоретических же исследованиях оно — не редкость (см., например, [432; 433]). Небезынтересно проследить на обсуждаемом примере к чему может привести отказ от предположения, что $\hat{S}$ содержит мультипликативную ошибку, в пользу предположения о наличии несистематической аддитивной ошибки (помехи), принятого во многих областях (см., например, [434, С. 596], [435, С. 241–244], [20; 436].). Пусть $\hat{S} = S + X$, где $\mathsf{P}\{S > 2\} = 1$ и случайная ошибка X имеет соответствующее усеченное нормальное распределение с нулевым средним (см., например, [437, С. 434]). Возвращаясь в начало доказательства *Теоремы 14*, видим, что, поскольку $\mathsf{E}\hat{S} = \mathsf{E}S + 0$, то наличие несистематической ошибки вообще никак не влияет на распределение $\mathsf{P}\{N_{\hat{B}}^{\mathsf{PS}} = k\}$, и все вероятностные характеристики очереди будут в точности равны фактическим. Допуская некоторую вольность речи, можно сказать, что дисциплина справедливого разделения процессора фильтрует случайную ошибку. Такое положение дел, конечно же, объясняется ее известным свойством (т. е. инвариантностью стационарного распределения общего числа заявок в системе относительно вида распределения времени обслуживания при фиксированном среднем), и имеет место для некоторых других специальных дисциплин обслуживания (например, LIFO с прерыванием, LIFO с прерыванием наикратчайшей заявкой). Аддитивная модель обладает и другим серьезным недостатком: в рамках изучаемой проблемы ей трудно придать наблюдаемые на практике черты. Например, сл. в. $\hat{S}$, будучи временем обслуживания, всегда положительна. Таким образом, сл. в. X должна принимать либо только неотрицательные значения, либо такие (положительные и отрицательные) значения, при которых сумма $\hat{S} = S + X$ всегда положительна. Оба эти ограничения представляются неестественными. А такое требование (обсуждаемое в [413; 414]), как “ошибка в большую сторону также вероятна, как и в меньшую”, по-видимому, вообще не реализуемо в рамках аддитивной модели; в мультипликативном же случае для его выполнения достаточно предположить, что сл. в. X имеет, например, логнормальное распределение. Дальнейшие результаты в связи с обсуждаемыми здесь вопросами можно найти в [296], а критику мультипликативной модели — в [30; 438].

знать их параметры!), и проверить принадлежность[20] сл. в. $\hat{S}$ классу $\bar{\mathscr{L}}$, чтобы иметь возможность уточнить оценку фактического распределения очереди. Новую, более точную оценку[21] дает СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с той же интенсивностью входящего потока и той же ф. р. времени обслуживания $\hat{B}(x)$. Необходимо отметить особо тот способ, с помощью которого посредством новой СМО достигается результат: каждая (поступающая в непустую систему) заявка назначает новое остаточное время обслуживания заявке на приборе, причем независимо от всей предыстории функционирования системы. Поскольку сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$, эти воздействия[22] укорачивают время обслуживания[23] и, в итоге, каждая заявка уходит с прибора раньше, чем предписывает ее прогноное время выполнения.

Следствием *Теоремы 14* (и отношения $\overset{d}{\leqslant}$) является упорядоченность моментов любого порядка стационарных распределений общего числа заявок в рассмотренных СМО; в частности[24],

$$\mathsf{E}\left(N_B^{\mathsf{PS}}\right)^r \leqslant \mathsf{E}\left(N_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)^r \leqslant \mathsf{E}\left(N_{\hat{B}}^{\mathsf{PS}}\right)^r, \quad 0 \leqslant r < \infty. \tag{2.6}$$

Пусть сл. в. V^{PS} имеет распределение, совпадающее со стационарным распределением времени пребывания заявки в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, а сл. в. $V^{\mathsf{LIFO\,Re}}$ — распределение той же характеристики, но в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$. Положив $r = 1$ в (2.6) и воспользовавшись законом

[20] Отметим, например, что принадлженость распределений длительностей выполнения работ в процессоре в системах разделения времени классу УФИ (и, значит, классу $\bar{\mathscr{L}}$) подчеркивалась еще в [439] и [75, С. 88]; изучение свойств распределений различных характеристик нагрузки продолжается и поныне, причем сейчас это направление исследований набирают все больший вес (см., например, [64; 440; 441], [354, Part VI] и [442, Section 6]).

[21] Отметим, что область стационарности новой СМО шире, чем исходной. Поэтому она позволяет получить оценки при таких значениях загрузки, которые ранее были вне рассмотрения.

[22] Пользуясь устоявшейся терминологией (см., например, [443]), эти воздействия можно назвать шоковыми. Для математических моделей с шоковыми воздействиями известно много результатов (см., например, [444–454] и ссылки в них). Однако, ввиду принципиальных отличий рассматриваемых моделей от шоковых (например, не происходит накопление шоковых воздействий, не случается критических превышений уровней и т. п.), уже известные результаты не удается приспособить для получения ответов на интересующие вопросы.

[23] Или удлинняет, если $\hat{S}$ принадлежит классу $\mathscr{L}$. Здесь интересно указать на связь, отмеченную в [455], между таким процессом обслуживания и характеризацией экспоненциального распределения.

[24] Все моменты существуют при $\lambda\mathsf{E}\hat{S} < 1$. При $\lambda\mathsf{E}\hat{S} \geqslant 1$, но $\frac{1}{2} < \hat{\beta}(\lambda) < 1$ не существуют моменты сл. в. $N_{\hat{B}}^{\mathsf{PS}}$. При $0 < \hat{\beta}(\lambda) < \frac{1}{2}$ и $\lambda\mathsf{E}S < 1$ не существуют также и моменты сл. в. $N_{\hat{B}}^{\mathsf{LIFO\,Re}}$. Наконец при $\lambda\mathsf{E}S \geqslant 1$ никакие моменты не существуют.

Литтла[25], обнаруживаем, что (в условиях *Теоремы 14*) упорядочены и стационарные средние времена пребывания заявок в системах[26], т. е.

$$\mathsf{E}\left(V_B^{\mathsf{PS}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right). \tag{2.7}$$

Однако для моментов более высоких порядков систему неравенств получить уже не удается.

Теорема 15. *Если сл. в. $\hat{S}$ представима в виде произведения $S \cdot X$, причем сл. в. S и X независимы и имеют соответсвенно экспоненциальное и лог–симметричное распределения, то справедливо неравенство*

$$\mathsf{Var}\left(V_B^{\mathsf{PS}}\right) \leqslant \min\left(\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right), \mathsf{Var}\left(V_{\hat{B}}^{\mathsf{PS}}\right)\right). \tag{2.8}$$

Доказательство. Напомним (см., например, [75, С. 81]), что дисперсия времени пребывания заявки в стационарной СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, равна

$$\mathsf{Var}\left(V_B^{\mathsf{PS}}\right) = \frac{(\mathsf{E}S)^2}{(1-\lambda\mathsf{E}S)^2}\frac{2+\lambda\mathsf{E}S}{2-\lambda\mathsf{E}S} = \frac{1}{\lambda^2}\cdot\frac{(1-y)^2(1+y)}{(2y-1)^2(3y-1)} = \frac{1}{\lambda^2}\cdot f_1(y),$$

где введено обозначение $y = \frac{1}{1+\lambda\mathsf{E}S}$. Заметим, что $\frac{1}{2} < y < 1$. Для нахождения формулы для дисперсии $\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ воспользуемся результатами *Теоремы 4*. Имеем

$$\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) = \varphi''(0) - \left(\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)\right)^2 = -\frac{(1-\hat{\beta}(\lambda))^2}{\lambda^2(2\hat{\beta}(\lambda)-1)^2} + \\ + \frac{2\left(1-3\hat{\beta}(\lambda)+3\hat{\beta}(\lambda)^2\right)}{\lambda^2\hat{\beta}(\lambda)^2(2\hat{\beta}(\lambda)-1)^3}\left(\hat{\beta}(\lambda)^2(1-\hat{\beta}(\lambda)) + \lambda(2\hat{\beta}(\lambda)-1)\hat{\beta}'(\lambda)\right), \tag{2.9}$$

[25]Который, как известно, справедлив для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, и, как следует из доказанных формул (1.24) и (1.25), имеет место и для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$.

[26]Известные интервальные оценки для $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$ при пуассоновском входящем потоке ничего не дают (см. [456, Theorem 3.2] и [457, Section 1]). Если же входящий поток другой (а, как будет видно из дальнейшего, метод "работает" и в этом случае), то вопрос использования нижней границы соответствующего интервала вместо $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$ (как и обоснование самого метода!) остается невыясненным.

где $\varphi''(0)$ — вторая производная по s в точке $s=0$ ПЛС $\varphi(s)$ стационарного распределения времени пребывания заявки в системе, найденного в (1.16), а $\hat{\beta}'(\lambda)$ — первая производная ПЛС $\hat{\beta}(s)$ по s в точке $s=\lambda$.

Чтобы получить нижнюю оценку для (2.9), необходимо найти границы изменения $\hat{\beta}'(\lambda)$. Покажем, что

$$-\frac{1}{4\lambda} \leqslant \hat{\beta}'(\lambda) \leqslant 0. \tag{2.10}$$

Поскольку первый и второй моменты сл. в. $V_{\hat{B}}^{\mathsf{LIFO\,Re}}$ существуют, то существует и дисперсия. Следующая цепочка равенств не требует других пояснений:

$$\begin{aligned}\hat{\beta}'(\lambda) &= \frac{d}{d\lambda}\int_{-\infty}^{\infty}\frac{1}{1+\lambda \mathsf{E}Se^{u}}\,g(u)du = \\ &= \int_{-\infty}^{\infty}\frac{-\mathsf{E}Se^{u}}{(1+\lambda\mathsf{E}Se^{u})^2}\,g(u)du = \\ &= -\int_{0}^{\infty}\left(\frac{\mathsf{E}Se^{u}}{(1+\lambda\mathsf{E}Se^{u})^2}+\frac{\mathsf{E}Se^{-u}}{(1+\lambda\mathsf{E}Se^{-u})^2}\right)g(u)du.\end{aligned} \tag{2.11}$$

Запишем двойное неравенство $0 \leqslant \frac{x}{(1+x)^2} \leqslant \frac{1}{4}$, справедливое при $x \geqslant 0$. Положив в нем сначала $x=\lambda\mathsf{E}Se^{u}$, а затем $x=\lambda\mathsf{E}Se^{-u}$, устанавливаем, что правый сомножитель в подынтегральном выражении (2.11) неотрицателен и не превосходит $\frac{1}{2\lambda}$. Вспоминая, что $\int_0^{\infty} g(u)du = \frac{1}{2}$, из (2.11) получаем (2.10).

Вернемся к (2.9) и заметим, что при $\frac{1}{2} < \hat{\beta}(\lambda) < 1$ дробь во втором слагаемом в правой части положительна. Следовательно нижняя оценка для дисперсии $\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ получится, если вместо $\hat{\beta}'(\lambda)$ подставить ее наименьшее значение т. е. $-\frac{1}{4\lambda}$. Вводя для сокращения записи обозначение $x=\hat{\beta}(\lambda)$, из (2.9) получаем

$$\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) \geqslant \frac{1}{\lambda^2}\cdot\frac{1-5x+3x^2+18x^3-34x^4+16x^5}{2x^2(2x-1)^3} = \frac{1}{\lambda^2}\cdot f_2(x).$$

Покажем, что $\mathsf{Var}\left(V_B^{\mathsf{PS}}\right) \leqslant \mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$. Для этого достаточно убедиться в том, что при[27] $\frac{1}{2} < x \leqslant y < 1$ имеет место неравенство $f_1(y) \leqslant f_2(x)$. Но $f_1(y)$ убывает при $\frac{1}{2} < y < 1$ т. к.

$$f_1'(y) = \frac{(1-y)(3-8y+y^2)}{(1-3y)^2(2y-1)^3} < 0,$$

[27] Неравенство $x \leqslant y$ следует из *Теоремы 14*.

и, кроме того, $f_1(x) < f_2(x)$ при $\frac{1}{2} < x < 1$, поскольку

$$f_2(x) - f_1(x) = \frac{1 - 8x + 20x^2 + 3x^3 - 86x^4 + 124x^5 - 52x^6}{2x^2(2x-1)^3(3x-1)} > 0.$$

Для завершения доказательства (2.8) осталось воспользоваться рассуждениями, использованными в [75, С. 87], для вывода свойства монотонности дисперсии в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, из которых следует, что $\mathsf{Var}\left(V_B^{\mathsf{PS}}\right) \leqslant \mathsf{Var}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$.

□

Ввиду громоздкости явного выражения для $\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$, дальнейшее уточнение правой части (2.8), т. е. установление соотношения между дисперсиями $\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ и $\mathsf{Var}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$, затруднено.

Интересно остановиться на другой, популярной в классических СМО временной характеристике — незаконченной работе. Обозначим через R^{PS} и $R^{\mathsf{LIFO\,Re}}$ сл. в. с распределениями, совпадающими со стационарными распределениями незаконченной работы соответственно в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ и СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$. Отметим, что выражение для $\mathsf{E}\left(R_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ было получено в (1.26); а формула для $\mathsf{E}\left(R_{\hat{B}}^{\mathsf{PS}}\right)$ может быть найдена, например, из соответствующего ПЛС в [75, С. 85]. Как известно (см., например, [416, С. 38–39]), если бы выполнялось соотношение $R_{\hat{B}}^{\mathsf{LIFO\,Re}} \overset{d}{\leqslant} R_{\hat{B}}^{\mathsf{PS}}$, то должно было выполняться соответствующее неравенство как для ПЛС, так и, конечно же, для средних т. е. $\mathsf{E}\left(R_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) \leqslant \mathsf{E}\left(R_{\hat{B}}^{\mathsf{PS}}\right)$. Однако можно привести примеры[28], когда

$$\mathsf{E}\left(R_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) - \mathsf{E}\left(R_{\hat{B}}^{\mathsf{PS}}\right) = \left(\frac{\mathsf{E}\hat{S}\hat{\beta}(\lambda)}{2\hat{\beta}(\lambda) - 1} - \frac{1 - \hat{\beta}(\lambda)}{\lambda\hat{\beta}(\lambda)}\right) - \frac{\lambda\mathsf{E}\hat{S}^2}{2(1 - \lambda\mathsf{E}\hat{S})} > 0.$$

Таким образом, в условиях *Теоремы 14* сл. в. $R_{\hat{B}}^{\mathsf{LIFO\,Re}}$ не является[29] во всем диапазоне загрузки $0 < \lambda\mathsf{E}\hat{S} < 1$ ни стохастически меньшей, ни стохастически большей, чем сл. в. $R_{\hat{B}}^{\mathsf{PS}}$.

[28] Например, $\hat{S} = S \cdot X$, S имеет экспоненциальное распределение с параметром 1, X — логнормальное распределение с параметрами 0 и $\sigma = 0.7$, и $\lambda = 0.1$.

[29] Известен ряд работ (см., например, [458] и обзор в [459]), в которых система $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ применяется для моделирования (генерации выборки из) стационарных распределений характеристик других, более сложных СМО (например, вектора остаточных длин обслуживания в системе $M\,|\,GI\,|\,c\,|\,\infty\,|\,\mathsf{PS}$, находящейся в стационарном режиме). Методы, которые позволяют это делать (упомянем, в частности, метод "Dominated CFTP" [459, Раздел 2.3.4]) и регенеративное моделирование (см. [459, Раздел 2.4.1] и [323; 355; 460–462]), требуют наличия системы, потраекторно мажорирующей исходную (например, $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$ мажорирует $M\,|\,GI\,|\,c\,|\,\infty\,|\,\mathsf{FIFO}$ с точки зрения процесса незаконченной работы в каждый момент времени). Возникает вопрос: возможно ли

Вернемся к неравенствам для средних времен. Как было доказано в *Теореме 4* среднее время пребывания в стационарной СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ совпадает со средней длиной ее периода занятости (ср. (1.23) и (1.25)). Таким образом, (2.7) не изменится, если вместо $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ — среднего времени пребывания подставить $\mathsf{E}\left(U_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ — среднюю длину периода занятости[30], т. е.

$$\mathsf{E}\left(V_B^{\mathsf{PS}}\right) \leqslant \mathsf{E}\left(U_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right). \tag{2.12}$$

Как показывает следующая теорема, к этой системе неравенств (и к системе (2.7)) можно добавить еще одно, дающее нетривиальную оценку снизу для значения $\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$. Напомним, что $\hat{b}(x)$ есть значение плотности распределения сл. в. $\hat{S}$ в точке x.

Теорема 16. *Если сл. в. $\hat{S}$ представима в виде произведения $S \cdot X$, причем сл. в. S и X независимы и имеют соответсвенно экспоненциальное и лог–симметричное распределения, то справедливы неравенства*

$$0 < \frac{1}{\hat{b}(0) - \lambda} \leqslant \mathsf{E}\left(V_B^{\mathsf{PS}}\right). \tag{2.13}$$

Доказательство. Как было доказано в *Теореме 4*, в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с интенсивностью входящего потока λ и ф. р. длины заявки $\hat{B}(x)$ ПЛС распределения времени пребывания заявки на приборе (обозначим его $\psi^{(1)}(s)$),

применение СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ для решения такой же задачи? По-видимому, ответ на этот вопрос отрицательный. Обозначим через $\chi(t)$ величину незаконченной работы в системе в момент t. Интуитивно ясно, что при одинаковых начальных условиях (ввиду изменения остаточного времени обслуживания заявки на приборе при каждом поступлении заявки) неравенство $\chi^{\mathsf{PS}}(t) \leqslant \chi^{\mathsf{LIFO\,Re}}(t)$ не может выполняться при всех $t > 0$. Так же обстоит дело, если рассмотреть процессы общего числа заявок в системах. Таким образом, (с точки зрения упомянутых двух характеристик) система $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ не является мажорантой (и минорантой) для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$.

[30] Отсюда автоматически получаются и оценки для средней длины периода занятости, поскольку, как известно (см., например, [354, С. 487]), для стационарной СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ средняя длина периода занятости равна среднему времени пребывания в системе произвольной заявки. Отметим, что этот результат не следуют из известных оценок для средних значений периода занятости и простоя классических систем $GI\,|\,GI\,|\,1\,|\,\infty\,|\,\cdot$ (см., например, [416, С. 112]). Неравенства (2.12) указывают на еще одно важное обстоятельство: оценкой интересующей характеристики (здесь это $\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$) может служить значение совершенно другой характеристики (здесь это средняя длина периода занятости). Прояснить эту ситуацию (хотя бы частично) помогает рассмотрение СМО с несколькими типами заявок (см. стр. 122).

рассчитывается по формуле (см. (1.17))

$$\psi^{(1)}(s) = \frac{\hat{\beta}(\lambda + s)(\lambda + s)}{s + \lambda\hat{\beta}(\lambda + s)}.$$

Из теоремы Бохнера–Хинчина (см., например, [463, С. 228]) следует[31], что $\psi^{(1)}(s)$ служит преобразованием Лапласа–Стилтьеса некоторой ф.р. Обозначим ее через $\hat{B}^{(2)}(x)$. Рассмотрим теперь СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с той же интенсивностью входящего потока λ и ф. р. длины заявки $\hat{B}^{(2)}(x)$. Повторяя рассуждения *Теоремы 4*, находим, что в этой новой системе ПЛС стационарного распределения времени пребывания заявки на приборе (обозначим его $\psi^{(2)}(s)$) существует при $\hat{\beta}(2\lambda) \in (\frac{1}{3},1)$ и равно

$$\psi^{(2)}(s) = \frac{\hat{\beta}(2\lambda + s)(2\lambda + s)}{s + \lambda\hat{\beta}(2\lambda + s)},$$

а для стационарного среднего $\mathsf{E}\left(V_{\hat{B}^{(2)}}^{\mathsf{LIFO\,Re}}\right)$ времени пребывания заявки в этой системе имеет место формула

$$\mathsf{E}\left(V_{\hat{B}^{(2)}}^{\mathsf{LIFO\,Re}}\right) = \frac{\hat{\beta}(2\lambda) - 1}{\lambda - \lambda\hat{\beta}(2\lambda) - 2\lambda\hat{\beta}(2\lambda)}.$$

Как ранее ПЛС $\psi^{(1)}(s)$, теперь и $\psi^{(2)}(s)$ служит преобразованием Лапласа–Стилтьеса некоторой ф. р. Обозначим ее через $\hat{B}^{(3)}(x)$ и опять рассмотрим СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ с той же интенсивностью входящего потока λ, и ф. р. длины заявки $\hat{B}^{(3)}(x)$, и т.д. Путем несложных, но утомительных преобразований, можно установить, что стационарное среднее время пребывания заявки в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$, рассматриваемой на n-м шаге, существует при $\hat{\beta}(n\lambda) \in (\frac{1}{n+1},1)$ и равно

$$\mathsf{E}\left(V_{\hat{B}^{(n)}}^{\mathsf{LIFO\,Re}}\right) = \frac{\hat{\beta}(n\lambda) - 1}{\lambda - \lambda\hat{\beta}(n\lambda) - n\lambda\hat{\beta}(n\lambda)},\ n \geqslant 1. \tag{2.14}$$

Устремим в последнем равенстве n к бесконечности. Поскольку $\lim_{x\to 0}\hat{b}(x) = \hat{b}(0)$ существует, из тауберовой теоремы (см., например, [320, С. 211]) следует, что $\lim_{n\to\infty} n\lambda\hat{\beta}(n\lambda) = \hat{b}(0)$. Учитывая, что $\lim_{n\to\infty}\hat{\beta}(n\lambda) = 0$, приходим к заключению, что описанные выше шаги, повторенные неограниченное число

[31] Впрочем можно поступить и по-другому, заметив, что время пребывания заявки на приборе есть сумма (случайного числа) независимых в совокупности неотрицательных сл. в.

раз, приводят к СМО, в которой стационарное среднее время пребывания заявки равно

$$\lim_{n\to\infty} \mathsf{E}\left(V^{\mathsf{LIFO\,Re}}_{\hat{B}^{(n)}}\right) = \frac{1}{\hat{b}(0)-\lambda}.$$

Выписывая теперь явный вид плотности $\hat{b}(x)$

$$\hat{b}(x) = \frac{1}{\mathsf{E}S}\int\limits_0^\infty e^{-\frac{x}{u\mathsf{E}S}}\frac{l(u)}{u}du,\ x \geqslant 0,$$

подставляя $x = 0$ и вспоминая, что сл. в. X и $1/X$ одинаково распределены, и $\mathsf{E}X \geqslant 1$, получаем

$$\hat{b}(0) = \frac{\mathsf{E}(1/X)}{\mathsf{E}S} \geqslant \frac{1}{\mathsf{E}S} > \lambda.$$

Для завершения доказательства осталось вспомнить, что $\mathsf{E}\left(V^{\mathsf{PS}}_B\right) = \frac{\mathsf{E}S}{1-\lambda\mathsf{E}S}$. Поэтому правое неравенство в (2.13) равносильно неравенству $\hat{b}(0)\mathsf{E}S > 1$, которое, как только что было показано, выполняется.

□

Если собрать вместе (2.7) и (2.13), то получается цепочка неравенств

$$0 < \frac{1}{\hat{b}(0)-\lambda} \leqslant \frac{\mathsf{E}S}{1-\lambda\mathsf{E}S} \leqslant \frac{1-\hat{\beta}(\lambda)}{\lambda(2\hat{\beta}(\lambda)-1)} \leqslant \frac{\mathsf{E}\hat{S}}{1-\lambda\mathsf{E}\hat{S}} < \infty,$$

из которой следует, что

$$\frac{1}{\hat{b}(0)} \leqslant \mathsf{E}S \leqslant \frac{1-\hat{\beta}(\lambda)}{\lambda\hat{\beta}(\lambda)}. \tag{2.15}$$

Вернемся к предложенной выше интерпретации результатов *Теоремы 14* (см. стр. 113), в соответствии с которой сл. в. S есть фактическое время обслуживания заявки на приборе. Неравенства[32] (2.15) показывают, во-первых, что

[32]Правое неравенство в (2.15) позволяет уточнить оценку для дисперсии, полученную в *Теореме 15*. Из свойства монотонности дисперсии времени пребывания в стационарной СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ (см. [75, С. 87]) следует, что замена распределения сл. в. $\hat{S}$ с $\hat{B}(x)$ на экспоненциальное (со средним равным правой части (2.15)), не увеличивает значение дисперсии. Поэтому правая часть двойного неравенства

$$\mathsf{Var}\left(V^{\mathsf{PS}}_B\right) \leqslant \left(\frac{1-\hat{\beta}(\lambda)}{\lambda(2\hat{\beta}(\lambda)-1)}\right)^2 \frac{1+\hat{\beta}(\lambda)}{3\hat{\beta}(\lambda)-1} \leqslant \mathsf{Var}\left(V^{\mathsf{PS}}_{\hat{B}}\right)$$

справедлива, а левая устанавливается непосредственной проверкой.

распределение планируемого времени обслуживания $\hat{S}$ содержит информацию о (неизвестном!) фактическом среднем времени обслуживания, и, во-вторых, что часть этой информации можно из этого распределения извлечь. Точность полученных оценок установить невозможно. Заметим лишь, что при отсутствии случайной ошибки в (2.15) выполняются точные равенства. Кроме того, предложенная оценка сверху лучше тривиальной оценки $\mathsf{E}\hat{S}$.

Условие, что СМО должна принадлежать множеству $\mathfrak{M}^*$ для того, чтобы с помощью предложенного метода можно быть уточнять оценки фактических значений ее стационарных характеристик, является довольно ограничительным. Хотя оно и приводит к интересным теоретическим выводам (см., например, (2.6)), но трудно рассчитывать на то, что $\mathfrak{M}^*$ велико настолько, чтобы метод можно было признать полезным для задач практики. Исправить ситуацию можно, но за счет большей части теоретических результатов. Например, соотношение (2.3), являющееся критерием принадлежности СМО множеству $\mathfrak{M}^*$, можно заменить двойным неравенством для средних значений[33]. Подобная замена по-видимому не может позволить продвинуться далеко в теоретическом плане, но увеличивает мощность $\mathfrak{M}^*$. Чтобы наглядно продемострировать последнее обстоятельство сделаем эту замену и предъявим еще один элемент из $\mathfrak{M}^*$. Рассмотрим систему $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{PS}$, на вход которой поступает $r > 1$ независимых пуассоновских потоков заявок различных типов интенсивности λ_i, $1 \leqslant i \leqslant r$. Положим $\lambda = \sum_{i=1}^{r} \lambda_i$. Длины $\hat{S}_i$ поступающих заявок — независимые в совокупности одинаково распределенные абсолютно непрерывные сл. в. с ф. р. $\hat{B}_i(x) = \mathsf{P}\{\hat{S}_i < x\}$ и конечным средним $\mathsf{E}\hat{S}_i$. Пусть сл. в. V^{PS} имеет распределение, совпадающее со стационарным распределением времени пребывания произвольной заявки в такой СМО. Как известно [75, С. 70], при при $\sum_{i=1}^{r} \lambda_i \mathsf{E}\hat{S}_i < 1$ имеет место равенство

$$\mathsf{E}\left(V^{\mathsf{PS}}_{\hat{B}_1,\ldots,\hat{B}_r}\right) = \frac{1}{\lambda} \frac{\sum\limits_{i=1}^{r} \lambda_i \mathsf{E}\hat{S}_i}{1 - \sum\limits_{i=1}^{r} \lambda_i \mathsf{E}\hat{S}_i}.$$

Возьмем теперь (исследованную в парарафе 1.3; см. стр. 69) систему $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{LIFO\,Re}$ с теми же входящими потоками, тем же набором ф. р.

[33] Например, для стационарных средних времен пребывания заявок в системе или других моментов.

длин заявок $\{\hat{B}_i(x), 1 \leqslant i \leqslant r\}$, и дисциплиной $\mathsf{LIFO\,Re}$. Сл. в., имеющую стационарное распределение периода занятости этой СМО, среднее значение которой конечно (согласно *Теореме 9*) при $0 < \sum_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{1-\hat{\beta}_i(\lambda)}{\hat{\beta}_i(\lambda)} < 1$, обозначим через $U^{\mathsf{LIFO\,Re}}_{\hat{B}_1,\ldots,\hat{B}_r}$. Напомним, что $\hat{\beta}_i(\lambda) = \int_0^\infty e^{-\lambda u} d\hat{B}_i(u)$. Согласно (1.51), средняя длина периода занятости вычисляется по формуле

$$\mathsf{E}\left(U^{\mathsf{LIFO\,Re}}_{\hat{B}_1,\ldots,\hat{B}_r}\right) = \frac{1}{\lambda} \frac{\sum\limits_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{1-\hat{\beta}_i(\lambda)}{\hat{\beta}_i(\lambda)}}{1 - \sum\limits_{i=1}^{r} \frac{\lambda_i}{\lambda} \frac{1-\hat{\beta}_i(\lambda)}{\hat{\beta}_i(\lambda)}}.$$

Теперь, путем простого сравнения дробей[34], нетрудно установить справедливость следующего утверждения, являющегося более слабым аналогом *Теоремы 14*: если при каждом $1 \leqslant i \leqslant r$ сл. в. $\hat{S}_i$ представима в виде произведения $S_i \cdot X_i$, причем сл. в. S_i и X_i независимы и имеют соответсвенно экспоненциальное и лог–симметричное распределения, то во всей области стационарности выполняется двойное неравенство

$$\mathsf{E}\left(V^{\mathsf{PS}}_{B_1,\ldots,B_r}\right) \leqslant \mathsf{E}\left(U^{\mathsf{LIFO\,Re}}_{\hat{B}_1,\ldots,\hat{B}_r}\right) \leqslant \mathsf{E}\left(V^{\mathsf{PS}}_{\hat{B}_1,\ldots,\hat{B}_r}\right). \tag{2.16}$$

Соотношения (2.16) указывают на важное обстоятельство: чтобы предложенный метод уточнения оценок фактических значений стационарных характеристик частично наблюдаемых СМО дал содержательный результат может потребоваться найти не только подходящий вариант дисциплины $\mathsf{LIFO\,GPP}$, но и правильный оценивающий показатель. Для системы $M \,|\, GI \,|\, 1 \,|\, \infty \,|\, \mathsf{PS}$ с несколькими типами заявок нужной дисциплиной оказалась дисциплина $\mathsf{LIFO\,Re}$ (см. стр. 69), а правильным показателем[35] — стационарная средняя длина ПЗ.

[34] В самом деле, для каждой сл. в. $\hat{S}_i$ в отдельности выполняется *Теорема 14* а, значит, и неравенства

$$\frac{1}{1+\lambda \mathsf{E}\hat{S}_i} \leqslant \hat{\beta}_i(\lambda) \leqslant \frac{1}{1+\lambda \mathsf{E} S_i}.$$

для ПЛС $\beta_i(\lambda)$.

[35] Оказалось не стационарное среднее время пребывания в системе произвольной заявки (см. формулу (1.43)). Это было бы (судя по полученным ранее результатам для СМО с одним типом заявок) вполне ожидаемым. Однако обратимся к другой характеристике — стационарному среднему времени пребывания в системе заявки типа i. Для СМО $M_r \,|\, GI_r \,|\, 1 \,|\, \infty \,|\, \mathsf{PS}$, как известно, оно равно

$$\frac{\mathsf{E}\hat{S}_i}{1 - \sum\limits_{j=1}^{r} \lambda_j \mathsf{E}\hat{S}_j} = \mathsf{E}\left(V^{\mathsf{PS}}_{\hat{B}_1,\ldots,\hat{B}_r}(i)\right);$$

С точки зрения практики, основным недостатком полученных результатов является предположение о видах распределений сл. в. S[36] и X. В следующих двух теоремах показано, что отказаться[37] от этого предположения можно, но в результате возникают ограничения другого рода.

Теорема 17. *Пусть сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$ и представима в виде произведения $S \cdot X$, где сл. в. S и X независимы, и $\mathsf{E}X \geqslant 1$. Тогда существует такое $n_0 \in [0,1]$, что при $n \in [0,n_0]$ справедливы неравенства*

$$\mathsf{E}\left(V_B^{\mathsf{PS}}\right) \leqslant \frac{\mathsf{E}\hat{S}}{1-\lambda\mathsf{E}\hat{S}} - n\lambda\frac{\mathsf{E}\hat{S}^2 - 2(\mathsf{E}\hat{S})^2}{(1-\lambda\mathsf{E}\hat{S})^2} \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right). \tag{2.17}$$

Доказательство. Поскольку сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$, то квадрат ее коэффициента вариации не меньше единицы, т. е. $\frac{\mathsf{E}\hat{S}^2-(\mathsf{E}\hat{S})^2}{(\mathsf{E}\hat{S})^2} \geqslant 1$. Вспоминая, что $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right) = \frac{\mathsf{E}\hat{S}}{1-\lambda\mathsf{E}\hat{S}}$, убеждаемся, что второе неравенство в (2.17) справедливо.

для СМО $M_r\,|\,GI_r\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ оно было найдено в (1.53) и равно

$$\frac{1-\hat{\beta}_i(\lambda)}{\lambda\hat{\beta}_i(\lambda)} + \frac{1}{\lambda}\frac{1}{1-\sum\limits_{j=1}^{r}\frac{\lambda_j}{\lambda}\frac{1-\hat{\beta}_j(\lambda)}{\hat{\beta}_i(\lambda)}}\sum_{j=1}^{r}\frac{\lambda_j}{\lambda}\frac{(1-\hat{\beta}_j(\lambda))^2}{\hat{\beta}_j^2(\lambda)} = \mathsf{E}\left(V_{\hat{B}_1,\ldots,\hat{B}_r}^{\mathsf{LIFO\,Re}}(i)\right).$$

С помощью полученных выше оценок нетрудно установить, что $\mathsf{E}\left(V_{\hat{B}_1,\ldots,\hat{B}_r}^{\mathsf{LIFO\,Re}}(i)\right) \leqslant \mathsf{E}\left(V_{\hat{B}_1,\ldots,\hat{B}_r}^{\mathsf{PS}}(i)\right)$ тогда и только тогда, когда $\sum_{j=1}^{r}\lambda_j\mathsf{E}\hat{S}_j(\mathsf{E}\hat{S}_j - \mathsf{E}\hat{S}_i) \leqslant 0$. Не ограничивая общности рассуждений можно считать, что типы потоков занумерованы в порядке возрастания значений средней длины заявки. Поэтому предыдущее неравенство точно выполняется при $i = r$ (т. е. для потока с самой большой средней длиной заявки), и не выполняется при $i = 1$ (т. е. для потока с самой маленькой средней длиной заявки). Другое же соотношение $\mathsf{E}\left(V_{B_1,\ldots,B_r}^{\mathsf{PS}}(i)\right) \leqslant \mathsf{E}\left(V_{\hat{B}_1,\ldots,\hat{B}_r}^{\mathsf{LIFO\,Re}}(i)\right)$ без введения дополнительных предположений проверить, по-видимому, невозможно: оно выполняется для всех тех i, при которых $\sum_{j=1}^{r}\lambda_j\mathsf{E}S_j(\mathsf{E}S_j - \mathsf{E}S_i) \geqslant 0$. Однако в отличие от неравенства для $\mathsf{E}\hat{S}_i$, здесь значения $\mathsf{E}S_1, \ldots, \mathsf{E}S_r$ неизвестны.

[36]Например, попытки заменить экспоненциальное распределение сл. в. S хотя бы на "почти" экспоненциальное (см. [464]) не дали результатов.

[37]Речь здесь идет, в том числе, и об отказе от эспоненциальности сл. в. S. Для классических СМО в этом направлении известно большое число результатов (см., например, [255; 457; 465–468], [416, Глава 5], [205, Глава 2]). Однако (по крайней мере стандартные) приемы неприменимы для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{LIFO\,Re}$ ввиду ее неконсервативности.

Поскольку $\mathsf{E}\left(V_B^{\mathsf{PS}}\right) = \frac{\mathsf{E}S}{1-\lambda\mathsf{E}S}$, левое неравенство в (2.17) равносильно неравенству

$$\underbrace{\mathsf{E}S(\mathsf{E}X-1)}_{\geqslant 0} - n\lambda \underbrace{\frac{1-\lambda\mathsf{E}S}{1-\lambda\mathsf{E}\hat{S}}\left(\mathsf{E}\hat{S}^2 - 2(\mathsf{E}\hat{S})^2\right)}_{\geqslant 0} \geqslant 0.$$

Оно выполняется при $n=0$ и, поскольку первое слагаемое в левой части не зависит от n, то существует[38] такое (зависящее от $\mathsf{E}S$, $\mathsf{E}\hat{S}^2$ и $\mathsf{E}X$) число $n_0 \in [0,1]$, что неравенство остается справедливым при всех $n \in [0,n_0]$.

□

К выражению в центральной части (2.17) приводят следующие соображения. Предположим, что формула (2.14) справедлива и для всех действительных $n \in [0,1)$. Далее будет неудобно использовать громоздкое обозначение $\mathsf{E}\left(V_{\hat{B}^{(n)}}^{\mathsf{LIFO\,Re}}\right)$ для правой части (2.14). Поэтому положим

$$f(n) = \mathsf{E}\left(V_{\hat{B}^{(n)}}^{\mathsf{LIFO\,Re}}\right) = \frac{\hat{\beta}(n\lambda)-1}{\lambda - \lambda\hat{\beta}(n\lambda) - n\lambda\hat{\beta}(n\lambda)},\ n \in [0,1].$$

По формуле Тейлора в окрестности точки $n=0$ имеем

$$f(n) = f(0) + nf'(0) + o(n). \tag{2.18}$$

Из того, что сл. в. $\hat{S}$ принадлежит классу $\mathscr{L}$ следует, что $\mathsf{E}\hat{S}^2 < \infty$. Поэтому существуют первая и вторая производные функции $\hat{\beta}$ (причем $(-1)^k\hat{\beta}^{(k)}(0) = \mathsf{E}\hat{S}^k$, $k=1,2$). Воспользовавшись правилом Лопиталя, находим

$$f(0) = \lim_{n\to+0} f(n) = \lim_{n\to+0} \frac{\lambda\hat{\beta}'(n\lambda)}{-\lambda^2\hat{\beta}'(n\lambda) - \lambda\hat{\beta}(n\lambda) - n\lambda^2\hat{\beta}'(n\lambda)} = \frac{\mathsf{E}\hat{S}}{1-\lambda\mathsf{E}\hat{S}}.$$

Выпишем теперь формулу для производной $f'(n)$:

$$\begin{aligned} f'(n) = \frac{1}{(\lambda-\lambda\hat{\beta}(n\lambda)-n\lambda\hat{\beta}(n\lambda))^2} \times \Big(&-\lambda\hat{\beta}'(n\lambda)\left(n\lambda\hat{\beta}(n\lambda)+\lambda\hat{\beta}(n\lambda)-\lambda\right) - \\ &-(1-\hat{\beta}(n\lambda))\left(n\lambda^2\hat{\beta}'(n\lambda)+\lambda^2\hat{\beta}'(n\lambda)+\lambda\hat{\beta}(n\lambda)\right)\Big) = \\ &= \lambda\frac{-n\lambda\hat{\beta}'(n\lambda) - (1-\hat{\beta}(n\lambda))\hat{\beta}(n\lambda)}{(\lambda-\lambda\hat{\beta}(n\lambda)-n\lambda\hat{\beta}(n\lambda))^2}. \end{aligned} \tag{2.19}$$

[38] Пусть, например, $n = \frac{1-\lambda\mathsf{E}\hat{S}}{1-\lambda\mathsf{E}S}$; $0 < n \leqslant 1$ поскольку $\mathsf{E}S \leqslant \mathsf{E}\hat{S}$. Тогда левая часть неравенства равна $\mathsf{E}\hat{S} - \mathsf{E}S - \lambda(\mathsf{E}\hat{S}^2 - 2(\mathsf{E}\hat{S})^2)$ и при большом втором моменте $\mathsf{E}\hat{S}^2$ может быть отрицательной.

Учитывая, что $\lambda \mathsf{E}\hat{S} < 1$ (чтобы существовала правая часть (2.17)) и $-\hat{\beta}'(n\lambda) \leqslant \mathsf{E}\hat{S}$[39], для числителя дроби имеем следующую оценку:

$$-n\lambda\hat{\beta}'(n\lambda) - (1-\hat{\beta}(n\lambda))\hat{\beta}(n\lambda) \leqslant n - (1-\hat{\beta}(n\lambda))\hat{\beta}(n\lambda).$$

Поскольку для существования $f(n)$ должно выполняться условие $\hat{\beta}(n\lambda) \in (\frac{1}{1+n},1)$, то $((1-\hat{\beta}(n\lambda))\hat{\beta}(n\lambda))'_n > 0$ и $((1-\hat{\beta}(n\lambda))\hat{\beta}(n\lambda))''_n < 0$ при $n \in [0,1]$. Значит найдется такое $n_0 \in [0,1]$, что при $n \in [0,n_0]$ числитель (2.19) отрицателен, и, значит, функция f убывает. Воспользовавшись дважды правилом Лопиталя, находим

$$f'(0) = \lim_{n\to+0} f'(n) = \lambda\frac{2(\mathsf{E}\hat{S})^2 - \mathsf{E}\hat{S}^2}{(1-\lambda\mathsf{E}\hat{S})^2}.$$

Подставляя $f(0)$ и $f'(0)$ в (2.18), и отбрасывая остаточный член, приходим к искомому выражению.

Теорема 17 расширяет область применения предложенного метода уточнения оценок фактических значений стационарных характеристик частично наблюдаемых СМО. В ней показано, что ограничения на тип распределений сл. в. S и X, накладываемые *Теоремой 14*, могут быть (иногда) сняты, и новые оценки остаются справедливыми во всей области стационарности. Вместе с тем, в них появляется новый параметр, значение которого неизвестно и не может быть найдено в рамках сделанных предположений[40].

Если же отказаться от требования, чтобы новые оценки были справедливы во всей области стационарности, то, как показывает следующая теорема, можно получить более сильный результат, чем (2.17).

Теорема 18. *Пусть сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$ и представима в виде произведения $S \cdot X$, где сл. в. S и X независимы, и $\mathsf{E}X \geqslant 1$. Тогда существует такое $\lambda_0 \geqslant 0$, что при всех $0 \leqslant \lambda \leqslant \lambda_0$ выполняются соотношения*

$$N_B^{\mathsf{PS}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{LIFO\,Re}} \overset{d}{\leqslant} N_{\hat{B}}^{\mathsf{PS}}. \tag{2.20}$$

[39]Это немедленно следует из того, что $-\hat{\beta}'(n\lambda) = \int_0^\infty ue^{-n\lambda u}d\hat{B}(u)$.

[40]Тем не менее, польза для практики из результата (2.17) может быть извлечена. Например, когда априори известно, что времена обслуживания заявок сильно завышены (и такие случаи типичны; например, в [415] отмечается, что пользователи суперкомпьютерных центров коллективного пользования в среднем завышают время выполнения на 86%), то подойдет любое $n \in (0,1)$.

Доказательство. Как показано в *Теореме 14* (2.20) выполняется тогда и только тогда, когда для $\hat{\beta}(\lambda)$ справедлива двусторонняя оценка

$$\frac{1}{1+\lambda \mathsf{E}\hat{S}} \leqslant \hat{\beta}(\lambda) \leqslant \frac{1}{1+\lambda \mathsf{E}S}.$$

Поскольку сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$, левое неравенство выполняется при любом $\lambda \geqslant 0$. Иначе обстоит дело с правым неравенством. Из того, что $\lambda \mathsf{E}\hat{S} < 1$ следует, что $1 - \lambda \mathsf{E}S \leqslant \frac{1}{1+\lambda \mathsf{E}S}$. Воспользуемся неравенством для преобразований Лапласа неотрицательных сл. в., полученным в [429]. Из формулы (18) в [429] имеем:

$$\hat{\beta}(\lambda) \leqslant 1 - \frac{\lambda(\mathsf{E}\hat{S})^2}{\lambda \mathsf{E}\hat{S}^2 + \mathsf{E}\hat{S}}, \ \lambda \geqslant 0.$$

Нетрудно видеть, что $1 - \frac{\lambda(\mathsf{E}\hat{S})^2}{\lambda \mathsf{E}\hat{S}^2+\mathsf{E}\hat{S}} \leqslant 1 - \lambda \mathsf{E}S$ тогда и только тогда, когда

$$\lambda \leqslant \frac{\mathsf{E}\hat{S}(\mathsf{E}X - 1)}{\mathsf{E}\hat{S}^2}.$$

Правая часть неравенства неотрицательна и является искомым значением λ_0.

□

И снова вернемся к предложенной выше интерпретации результатов *Теоремы 14* (см. стр. 113), в соответствии с которой фактические времена обслуживания S содержат случайную ошибку X. Поскольку по условию *Теоремы 18* сл. в. $\hat{S} = S \cdot X$ принадлежит классу $\bar{\mathscr{L}}$, то квадрат ее коэффициента вариации не меньше единицы и, значит, $\frac{\mathsf{E}\hat{S}^2}{\mathsf{E}\hat{S}} \geqslant 2\mathsf{E}\hat{S}$. Но тогда справедливо неравенство

$$\lambda_0 = \frac{\mathsf{E}\hat{S}(\mathsf{E}X - 1)}{\mathsf{E}\hat{S}^2} \leqslant \frac{1}{\mathsf{E}S}\left(\frac{1}{2} - \frac{1}{2\mathsf{E}X}\right),$$

которое помогает прояснить физический смысл *Теоремы 18*. Если прогнозные времена обслуживания $\hat{S}$ сильно завышены, то, по крайней мере когда (ненаблюдаемая!) система загружена меньше чем наполовину, можно рассчитывать на получение более точных оценок фактического распределения общего числа заявок в системе. Если же прогнозные времена обслуживания близки фактическими, то (скорее всего) такой возможности нет.

Как и в случае с *Теоремой 17*, польза для практики от *Теоремы 18* ограничена, поскольку значение λ_0 невозможно вычислить. Однако, ограничившись областью малой загрузки, можно (в известных случаях; см. сноску на стр. 125) быть почти уверенным, что (2.20) выполняются.

2.3 Дополнения

1. В множество $\mathfrak{M}^*$ входят СМО не только с дисциплиной справедливого разделения процессора, но и с другими (в том числе и классическими) дисциплинами обслуживания. Например, неравенства для первых моментов (2.7) остаются справедливыми, если вместо дисциплины PS взять FIFO[41]. Ключевую роль здесь играет предположение об экспоненциальности сл. в. S. Если же отказаться от него и заменить распределение $B(x) = \mathsf{P}\{S < x\}$ на другое, скажем на распределение Вейбулла

$$b(x) = B'(x) = k\Gamma(1+k^{-1})(x\Gamma(1+k^{-1}))^{k-1}e^{-(x\Gamma(1+k^{-1}))^k},\ k > 0,\ x \geqslant 0,$$

то при $k \neq 1$ неравенства (2.7) более не выполняются. Причина этого кроется в том, что дисциплина LIFO Re никак не учитывает изменившиеся свойства распределения $\hat{B}(x)$ (например, что хвост распределения стал тяжелее, если $k < 1$). Как показывает следующий пример, существуют случаи, когда выбор (вместо LIFO Re) другой дисциплины обслуживания позволяет исправить ситуацию. Пусть $k = 0.5$, а сл. в. X имеет логнормальное распределение с параметрами 0 и 0.5. Графически (см. рисунок ниже) можно убедиться, что сл. в. $\hat{S} = S \cdot X$ принадлежит классу $\bar{\mathscr{L}}$.

Значения, приведенные в следующей таблице (см. таблица 1), свидетельствуют о том, что неравенства $\mathsf{E}\left(V_B^{\mathsf{FIFO}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{FIFO}}\right)$ не выполняются[42] ни при каком значении λ. Изменим дисциплину LIFO Re следующим образом: каждая поступающая в непустую систему заявка назначает новую остаточную длину заявке на приборе, если и только если текущая остаточная длина последней не меньше некоторого заранее фиксированного числа. Такую пороговую дисциплину с порогом $\theta \geqslant 0$ будем обозначать LIFO Re(θ). Очевидно, LIFO Re(θ)

[41] Или LIFO или Random. Однако, например, для дисциплины SRPT это уже не так.

[42] Оценка $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ оказывается заниженной и связано это с тем, что назначение новых остаточных длин обслуживающимся заявкам происходит слишком часто для данного распределения $\hat{B}(x)$.

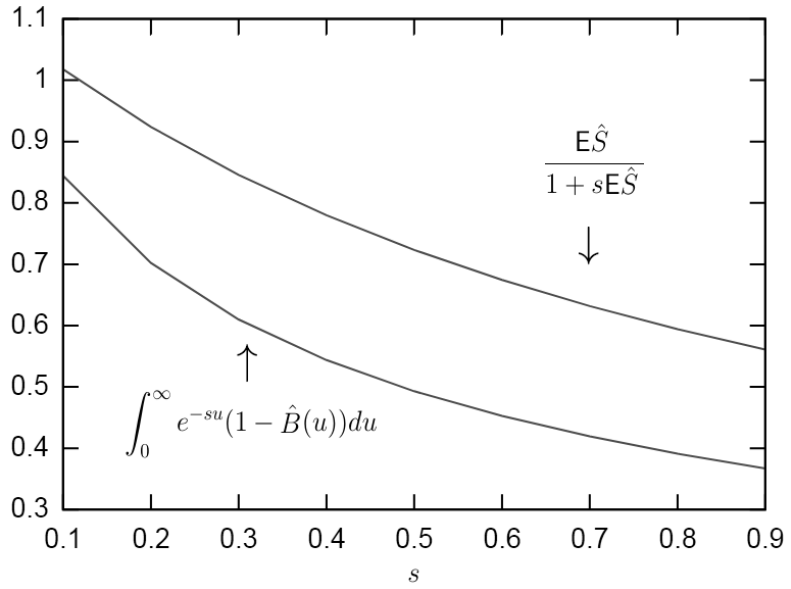


Рисунок 1 — Левая и правая части неравенства (2.1) при $s < \frac{1}{\mathsf{E}\hat{S}}$

есть одна из разновидностей дисциплины $\mathsf{LIFO\,GPP}$, причем[43]

$$d(x,y|u,v) = \begin{cases} \hat{b}(x)\delta(y-v), & u > \theta, v < \theta, \\ \hat{b}(y)\delta(x-u), & u < \theta, v > \theta, \\ \delta(x-u)\delta(y-v), & u < \theta, v < \theta, \\ \hat{b}(x)\hat{b}(y), & u \geqslant \theta, v \geqslant \theta, \end{cases}$$

а остальные определяющие дисциплину $\mathsf{LIFO\,GPP}$ функции тождественно равны нулю. Возвращаясь к таблице 1 видим, что, варьируя значения порога θ, можно добиться того, чтобы (при всех λ) выполнялись неравенства[44]

$$\mathsf{E}\left(V_B^{\mathsf{FIFO}}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}(\theta)}\right) \leqslant \mathsf{E}\left(V_{\hat{B}}^{\mathsf{FIFO}}\right).$$

Открытым (и, по-видимому, неразрешимым без дополнительных предположений) остается вопрос вычисления значения θ без информации о параметрах распределений сл. в. S и X.

2. Предложенный метод уточнения оценок фактических значений стационарных характеристик частично наблюдаемых СМО может давать содержательные результаты не только при пуассоновских входящих потоках.

[43] Напомним, что всюду δ обозначает дельта–функцию Дирака.

[44] Значения порога для указанных в таблице значений $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}(\theta)}\right)$ равно 25; оно было найдено численно по точным формулам параграфа 1.1.

Таблица 1 — Стационарные средние времена пребывания в СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$ при различных интенсивностях входящего потока λ: $\mathsf{E}\left(V_B^{\mathsf{FIFO}}\right)$ — фактическое, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{FIFO}}\right)$ — прогнозное, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ — оценка с дисциплиной $\mathsf{LIFO\,Re}$, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}(\theta)}\right)$ — оценка с дисциплиной $\mathsf{LIFO\,Re}(\theta)$

λ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\mathsf{E}\left(V_B^{\mathsf{FIFO}}\right)$	1.333	1.750	2.286	3	4	5.5	8	13	28
$\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$	1.056	1.001	0.983	0.982	0.991	1.009	1.034	1.066	1.105
$\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}(\theta)}\right)$	1.47	1.90	2.49	3.28	4.43	6.16	9.4	16.9	54
$\mathsf{E}\left(V_{\hat{B}}^{\mathsf{FIFO}}\right)$	1.688	2.406	3.370	4.734	6.811	10.359	17.794	43.256	∞

Рассмотрим СМО $D\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$, в которой времена меджу поступлениями заявок имеют вырожденное распределение (скажем, в точке λ^{-1}). Пусть $B(x) = \mathsf{P}\{S < x\} = 1 - e^{-x}$, сл. в. X распределена логнормально с параметрами 0 и 0.7, а сл. в. $\hat{S}$ имеет распределение $\hat{B}(x) = \mathsf{P}\{S \cdot X < x\}$. Из таблицы 2, в которой приведены значения[45] средних времен пребывания $\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$ при интенсивностях входящего потока λ, равномерно покрывающих область стационарности, видно, что предложенный метод обеспечивает выполнение (по крайней мере) неравенств для первых моментов (2.7).

Таблица 2 — Стационарные средние времена пребывания в СМО $D\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{PS}$ при различной загрузке λ: $\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$ — фактическое, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$ — прогнозное, $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ — оценка

λ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
$\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$	1	1.007	1.043	1.120	1.255	1.480	1.876	2.692	5.179
$\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$	1.264	1.254	1.275	1.347	1.474	1.684	2.046	2.710	4.287
$\mathsf{E}\left(V_{\hat{B}}^{\mathsf{PS}}\right)$	1.295	1.383	1.561	1.884	2.470	3.735	8.254	∞	∞

Отметим, что значения $\mathsf{E}\left(V_B^{\mathsf{PS}}\right)$ были рассчитаны по известным точным формулам (см. [75, С. 102]). Что касается значений в последних двух строках таблицы 2, то они были получены с помощью имитационного моделирования.

[45] Отметим, что в последовательности значений $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ нет (ожидавшейся) монотонности.

Вопрос о возможности аналитического расчета $\mathsf{E}\left(V_{\hat{B}}^{\mathsf{LIFO\,Re}}\right)$ и аналитического обоснования наблюдаемого эффекта остается невыясненным[46].

3. Прием, позволивший получить оценки для рассмотренных частично наблюдаемых СМО, по сути заключается в следующем изменении правила обслуживания заявок: назначить обслуживающейся заявке новую остаточную длину (независимо от всей предыстории функционирования системы), когда в систему поступает заявка. Как показывают вычислительные эксперименты, внесение такого изменения в любую модель частично наблюдаемой стохастической системы позволяет получать содержательные результаты. Приведем лишь один пример. Обратимся к модели вычислительного кластера из [469] (см. рисунок 2).

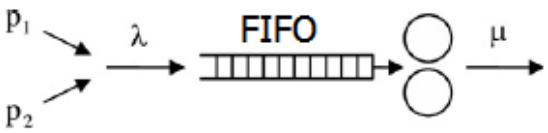


Рисунок 2 — Модель вычислительного кластера с двумя процессорами (p_i — вероятность того, что заявке для выполнения требуется i процессоров); рисунок взят из [469, Fig. 1]

Она представляет собой два идентичных процессора (единичной производительности), обслуживающих заявки (в порядке поступления) из единственной очереди неограниченной емкости. Заявки поступают в очередь по пуассоновскому закону (с параметром λ), их длины $\hat{S}_1, \hat{S}_2, \ldots$ (или, по-другому, времена вычисления; см. [470]) становятся известны в момент поступления, являются независимыми сл. в. и имеют одинаковое распределение $\hat{B}(x) = \mathsf{P}\{\hat{S} < x\}$. Для выполнения каждой заявке с вероятностью $p_1 \in (0{,}1)$ требуется один процессор, а с дополнительной вероятностью — два; конкретное число становится известным в момент поступления заявки в систему и не изменяется вплоть до момента ее ухода. Предполагается, что занятие (и освобождение) двух процессоров происходит одновременно. Заявка поступает на обслуживание, во-первых, когда подошла ее очередь и, во-вторых, когда требуемое ей число процессоров

[46] Хотя, например, условие стационарности хорошо понятно (см. [93, Theorem 4.2]) и имеет вид $\mathsf{E}H(T) \geqslant 2$; здесь $H(t)$ — функция восстановления простого процесса восстановления, в котором длительности восстановления имеют ф. р. $\hat{B}(x)$, а сл. в. T — длина интервала между поступлениями двух соседних заявок.

свободно. Таким образом, система является неконсервативной: процессор может простаивать, когда в системе есть незаконченная работа. Пусть параметры модели выбраны так, что существует[47] стационарный режим; обозначим стационарное среднее время пребывания через $\mathsf{E}V_{\hat{B}}$. Предположим, что времена вычисления $\hat{S}$ в среднем завышены[48], и фактическое время вычисления S каждой заявки связано с временем $\hat{S}$, заявленным при поступлении, соотношением $\hat{S} = S \cdot X$ (и, значит, $\mathsf{E}X > 1$). Условимся обозначать неизвестную ф. р. сл. в. S через $B(x)$. Если в исходной модели заменить $\hat{B}(x)$ на $B(x)$, то очевидно стационарный режим будет существовать; обозначим (неизвестное) стационарное среднее время пребывания через $\mathsf{E}V_B$. Что можно сказать о значении $\mathsf{E}V_B$ кроме того, что оно (вероятно[49]) не превосходит $\mathsf{E}V_{\hat{B}}$? Если сл. в. $\hat{S}$ принадлежит классу $\bar{\mathscr{L}}$, то, видоизменяя (как описано выше) правило обслуживания, можно получить нечто большее. Пусть теперь каждая поступающая заявка, прежде чем занять последнее место в очереди, прерывает обслуживание и назначает заявкам на процессорах новое остаточное время выполнения из распределения $\hat{B}(x)$. Обозначим стационарное среднее время пребывания в этой модели через $\mathsf{E}V_{\hat{B}}^{\mathsf{Re}}$. Оказывается, что (по крайней мере) в области малой загрузки могут выполняться неравенства

$$\mathsf{E}V_B < \mathsf{E}V_{\hat{B}}^{\mathsf{Re}} < \mathsf{E}V_{\hat{B}}. \tag{2.21}$$

Действительно, пусть сл. в. S имеет экспоненциальное распределение с параметром 1, а загрузка системы равна 0.05. График $\mathsf{E}V_B$, как функции от интенсивности входящего потока λ[50], представлен в [469] (см. Fig. 9b), причем $\lambda \in (0{,}0.1)$. Предположим теперь, что сл. в. X имеет логнормальное распределение с параметрами 0 и $\sigma = 0.25$. Как уже не раз было показано выше сл. в. $\hat{S} = S \cdot X$ принадлежит классу $\bar{\mathscr{L}}$.

На рисунке 3, изображены кривые зависимостей $\mathsf{E}V_B$, $\mathsf{E}V_{\hat{B}}^{\mathsf{Re}}$ и $\mathsf{E}V_{\hat{B}}$ от $\lambda \in (0{,}0.1)$. Отмеченные значения $\mathsf{E}V_{\hat{B}}^{\mathsf{Re}}$ и $\mathsf{E}V_{\hat{B}}$ были получены путем имитационного моделирования. Из рисунка видно, что сформулированное выше

[47] Отметим, что проблема нахождения условий стационарности для подобных моделей (как вычислительных кластеров, так и систем социального обслуживания (см. [471; 472])) остается по большей части открытой (см. [442, Section 2] и [473]). Также обстоит дело и с проблемой нахождения характеристик их производительности (см., например, [474], а также [442, Section 2]).

[48] См. сноску на стр. 107.

[49] Что, вообще говоря, необходимо доказывать.

[50] Отметим, что значения загрузки системы, среднего времени вычисления и интенсивности входящего потока однозначно определяют значение p_1 (см. соотношение (111) в [469]).

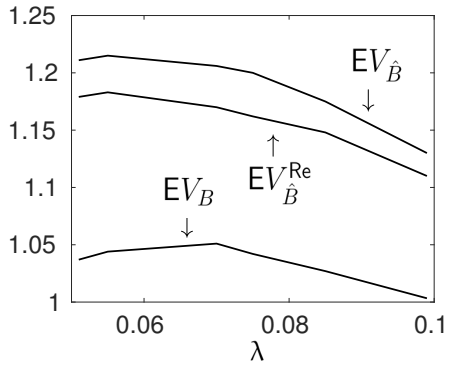


Рисунок 3 — Стационарные средние времена пребывания при малой загрузке: $\mathsf{E}V_B$ — фактическое, $\mathsf{E}V_{\hat{B}}$ — прогнозное, $\mathsf{E}V_{\hat{B}}^{\mathsf{Re}}$ — оценка

утверждение справедливо: (2.21) выполняется при всех $\lambda \in (0{,}0.1)$. Теоретическое обоснование этого экспериментального факта пока не найдено.

Глава 3. Алгоритмы управления для частично наблюдаемых стохастических систем с параллельным обслуживанием

В этой и следующей главах диссертации внимание сосредоточено на задаче централизованного оптимального управления входящими потоками в частично наблюдаемых стохастических системах с параллельным обслуживанием[1]. Типичная система представляют собой совокупность параллельно и независимо друг от друга работающих обслуживающих ресурсов, которые выполняют задания, направляемые на них единственным диспетчером. При этом диспетчер, осуществляя выбор ресурса для выполнения очередного задания, не имеет возможности отложить решение. Ему также недоступна какая-либо динамическая информация о состоянии ресурсов; при этом, однако, в его распоряжении имеется определенная априорная информация о системе.

3.1 Аналитический подход. Алгоритмы управления при прямом порядке обслуживания в однопроцессорных серверах

Рассмотрим систему в непрерывном времени, состоящую из $M \geqslant 2$ параллельно работающих серверов, в которую поступает рекуррентный поток заданий. Индексируя серверы числами, начиная с единицы, будем обозначать производительность сервера m через $v^{(m)}$. Примем, что $0 < v^{(m)} < \infty$, $1 \leqslant m \leqslant M$, причем не все $v^{(m)}$ равны между собой. Задания поступают в систему по одному, причем интервалы между поступлениями образуют последовательность независимых случайных величин с распределением $F(x)$, средним $\int_0^\infty x dF(x) = \lambda^{-1}$ и коэффициентом вариации $C_F < \infty$. Каждое задание имеет случайный объем (размер), причем его распределение является непрерывным[2] и может зависеть от порядкового номера задания. Будем обозначать распределение размера S_n задания, поступившего в систему n-м по счету, через $B_n(x)$,

[1] Как уже говорилось во Введении, примером таких систем являются системы добровольных вычислений (volunteer computing) [41, Section 2.3].

[2] Это предположение не связано с сутью дела и сделано для упрощения изложения. Отказ от него повлечет некоторые (несущественные) изменения в представленные ниже выкладки.

предполагая[3], что коэффициент вариации $C_{B_n} < \infty$. В каждом сервере имеется очередь неограниченной емкости для хранения заданий и один процессор для обработки, причем выбор на обслуживание происходит в соответствии с дисицплиной FIFO. Серверы работают независимо, без обмена заданиями и являются абсолютно надежными.

Каждое поступившее задание должно быть немедленно направлено на один из серверов. Пусть $0 \leqslant t_1 < \cdots < t_n < \ldots$ — последовательность моментов поступления заданий в систему. Решение (действие), принимаемое диспетчером (в автоматическом или ручном режиме), в момент t_n относительно поступившего задания обозначим через y_n. Очевидно, $y_n \in \{1, \ldots, M\}$. Пусть V_n — время, проведенное в системе заданием, поступившим в момент t_n и обслуженным согласно правилу y_n. Цель диспетчера — минимизировать стационарное среднее $\mathsf{E}V$ время пребывания задания в системе, определяемое как

$$\mathsf{E}V = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \mathsf{E}_y V_n, \tag{3.1}$$

где E_y — интегрирование по мере, порождаемой последовательностью y. При принятии решения относительно задания, поступающего в момент времени t_{n+1}, помимо информации о дисциплине обслуживания в серверах и их производительностях, диспетчеру известны только лишь

- предыстория принятых решений $y_1, \ldots, y_n$, включая моменты времени $t_1, \ldots, t_n$, в которые эти решения принимались, и
- распределения $B_1(x), \ldots, B_n(x)$, $B_{n+1}(x)$ размеров поступивших заданий.

Какая-либо динамическая информация о состоянии системы (например, о числе заданий в серверах, об остаточных временах обслуживания, о размерах заданий и др.) диспетчеру недоступна. Как уже упоминалось во Введении, к настоящему времени в научной литературе известно два способа для достижения поставленной цели: использовать либо рандомизированную, либо программную стратегию[4]. В этом параграфе описывается новый подход к порождению диспетчеризаций, являющихся наилучшими из известных во всем диапазоне изменений значений исходных параметров системы.

[3]Наложенные на все распределения ограничения связаны с рассматриваемым целевым функционалом и дисциплиной обслуживания, и могут быть ослаблены (или усилены) при их пересмотре.

[4]См. подробное описание этих стратегий начиная, со стр. 20.

Основная идея нового подхода состоит в использовании всей доступной предыстории наблюдаемых компонент. Точнее говоря, помимо самих решений $y_1,y_2,\ldots$, предлагается использовать информацию о моментах времени $t_1,t_2,\ldots$, в которые эти решения принимались. Положим

$$y_{n+1} = \mathrm{argmin}_{1\leqslant m\leqslant M}\left(\mathsf{E}W_{n+1}^{(m)} + \frac{\mathsf{E}S_{n+1}}{v^{(m)}}\right),\ n\geqslant 0, \tag{3.2}$$

где $W_{n+1}^{(m)}$ — время, необходимое для выполнения всех заданий, имеющихся в сервере m в момент t_{n+1}, без учета задания, поступившего в этот момент. Напомним, что неоднозначность при нахождении минимума разрешается в пользу самого быстрого сервера и, если их несколько, — равновероятным выбором. Число y_{n+1} служит номером сервера, на который направляется задание, поступившее в момент t_{n+1}. Диспетчеризация[5] $y = \{y_1,y_2,\ldots\}$, которую далее условимся кодировать AA, основываются на предыстории принятых решений и моментах поступления заданий.

Заметим, что математические ожидания $\mathsf{E}W_n^{(m)}$ в (3.2) являются условными и зависят от распределений размеров первых n заданий и моментов их поступлений. Для расчета $W_n^{(m)}$ воспользуемся рекурсией Линдли, согласно которой

$$W_{n+1}^{(m)} = \max\left(0, W_n^{(m)} + \frac{S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} - \tau_n\right),\ n\geqslant 1, \tag{3.3}$$

где $\mathbf{1}_{(A)}$ — индикатор множества A, а $\tau_n = t_{n+1} - t_n$ — время между поступлением $(n+1)$-го и n-го задания.

Для расчета $\mathsf{E}W_n^{(m)}$ поступим следующим образом. Проведем формальное "квантование" каждого из распределений $B_1(x)$, $B_2(x)$, ... с фиксированным шагом $0<\Delta\ll 1$. Обозначая дискретный аналог сл. в. S_n через $\tilde{S}_n$, положим[6]

$$\mathsf{P}\left\{\tilde{S}_n = k\Delta\right\} = \mathsf{P}\left\{S_n < (k+0.5)\,\Delta\right\} - \mathsf{P}\left\{S_n < (k-0.5)\,\Delta\right\} =$$
$$= B_n((k+0.5)\Delta) - B_n((k-0.5)\Delta) = s_n(k),\ k = 0,1,\ldots \tag{3.4}$$

[5] Диспетчеризация (3.2) "подражает" известной стратегии LWL, используя вместо точных значений незаконченной работы в каждом сервере их средние значения, посчитанные по участку траектории, начиная с первого задания.

[6] Как, например, в [475]. Но можно воспользоваться и другими способами дискретной аппроксимации (см., например, [416, С. 181] и [476]).

Перейдя от непрерывных распределений размеров заданий к дискретным, естественным образом получается и "квантование" распределения сл. в. $W_n^{(m)}$. Обозначая ее дискретный аналог через $\tilde{W}_n^{(m)}$, из (3.3) из формулы полной вероятности следуют соотношения для вероятностей $w_n^{(m)}(k) = \mathsf{P}\left\{\tilde{W}_n^{(m)} = k\Delta\right\}$, $n \geqslant 2,\ k = 0,1,\ldots$: при $m = y_n$ имеем

$$w_{n+1}^{(m)}(0) = \sum_{i=0}^{\left\lfloor \frac{\tau_n v^{(m)}}{\Delta} \right\rfloor} \sum_{j=0}^{\left\lfloor \frac{\tau_n}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor} w_n^{(m)}(j) s_n(i), \tag{3.5}$$

$$w_{n+1}^{(m)}(k) = \sum_{i=0}^{kv^{(m)} + \left\lfloor \frac{\tau_n v^{(m)}}{\Delta} \right\rfloor} w_n^{(m)}\left(k + \left\lfloor \frac{\tau_n}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor\right) s_n(i),\ k = 0,1,\ldots; \tag{3.6}$$

при $m \neq y_n$ получаем

$$w_{n+1}^{(m)}(0) = \sum_{i=0}^{\left\lfloor \frac{\tau_n}{\Delta} \right\rfloor} w_n^{(m)}(i), \tag{3.7}$$

$$w_{n+1}^{(m)}(k) = w_n^{(m)}\left(k + \left\lfloor \frac{\tau_n}{\Delta} \right\rfloor\right),\ k = 0,1,\ldots. \tag{3.8}$$

Здесь и далее $\lfloor \cdot \rfloor$ обозначает округление вниз. Задав начальное состояние системы и распределения $\left\{w_1^{(m)}(k) = \mathsf{P}\{\tilde{W}_1^{(m)} = k\Delta\},\ k = 0,1,\ldots\right\}$, $1 \leqslant m \leqslant M$, по соотношениям (3.5)–(3.8) теоретически можно рассчитать значения любых (как угодно далеких) вероятностей $w_n^{(m)}(k)$.

Воспользовавшись равенством (8.91) в [353, Раздел 8.4]), получим:

$$\max\left(0, W_n^{(m)} + \frac{S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} - \tau_n\right) - \max\left(0, -W_n^{(m)} - \frac{S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} + \tau_n\right) = \\ = W_n^{(m)} + \frac{S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} - \tau_n.$$

Отсюда, взяв математическое ожидание от обеих частей, находим

$$\mathsf{E}W_{n+1}^{(m)} = \mathsf{E}\left(\max\left(0, -W_n^{(m)} - \frac{S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} + \tau_n\right)\right) + \mathsf{E}W_n^{(m)} + \frac{\mathsf{E}S_n}{v^{(m)}}\mathbf{1}_{(m=y_n)} - \tau_n.$$

Выписывая теперь явный вид первого слагаемого в правой части, различая случаи $\mathbf{1}_{(m=y_n)} = 0$ и $\mathbf{1}_{(m=y_n)} = 1$, приходим к рекуррентной формуле для приближенного расчета при $n \geqslant 1$ математических ожиданий, входящих в (3.2):

$$\mathsf{E}W_{n+1}^{(m)} \approx \begin{cases} \mathsf{E}W_n^{(m)} + \frac{\mathsf{E}S_n}{v^{(m)}} - \tau_n + \\ + \sum\limits_{i=0}^{\left\lfloor \frac{\tau_n v^{(m)}}{\Delta} \right\rfloor} \sum\limits_{j=0}^{\left\lfloor \frac{\tau_n}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor} \left(\tau_n - j\Delta - \frac{i\Delta}{v^{(m)}}\right) w_n^{(m)}(j) s_n(i), & \mathbf{1}_{(m=y_n)} = 1, \\ \mathsf{E}W_n^{(m)} - \tau_n + \sum\limits_{i=0}^{\left\lfloor \frac{\tau_n}{\Delta} \right\rfloor} (\tau_n - i\Delta) w_n^{(m)}(i), & \mathbf{1}_{(m=y_n)} = 0. \end{cases} \tag{3.9}$$

Теперь, задавшись некоторым[7] значением Δ и зафиксировав управление y_1 для первого по счету задания, можно воспользоваться (3.2) для нахождения управлений для всех последующих заданий. Формальное описание соответствующего алгоритма для задания, поступившего в момент t_{n+1}, представлено ниже. Помимо исходных значений M, $v^{(1)}, \ldots, v^{(M)}$, выбранного Δ и t_{n+1} входными данными являются: дискретизованное распределение $B_n(x)$, среднее значение $\mathsf{E}S_{n+1}$ размера $(n+1)$-го задания, средние значения $\mathsf{E}W_n^{(1)}, \ldots, \mathsf{E}W_n^{(M)}$ незаконченной работы в каждом из серверов в момент t_n, управления $y_1, \ldots, y_n$ и моменты $t_1, \ldots, t_n$, в которые эти управления применялись. Выходные данные — это номер сервера y_{n+1}, на который следует отправить поступившее в момент t_{n+1} задание и средние значения $\mathsf{E}W_{n+1}^{(1)}, \ldots, \mathsf{E}W_{n+1}^{(M)}$.

[7] Подходящее значение этого единственного неизвестного параметра приходится искать в каждой задаче методом проб и ошибок. При этом необходимо учитывать, что значение Δ должно быть меньше среднего времени между поступлениями заданий, а также меньше среднего времени обслуживания любого задания на любом из процессоров.

Алгоритм I. Псевдокод алгоритма выбора управления для задания, поступившего в момент t_{n+1}, $n \geqslant 1$

---

for $l = 2 \to n$ **do**
  for $k = 0 \to \sum_{j=l-1}^{n} \left\lfloor \frac{\tau_j v^{(y_j)}}{\Delta} \right\rfloor$ **do**
    $s_l(k) = B_l((k+0.5)\Delta) - B_l((k-0.5)\Delta)$
for $l = 2 \to n$ **do**
  if $m = y_l$ **then**
    $w_l^{(m)}(0) = \sum\limits_{i=0}^{\left\lfloor \frac{\tau_{l-1} v^{(m)}}{\Delta} \right\rfloor} \sum\limits_{j=0}^{\left\lfloor \frac{\tau_{l-1}}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor} w_{l-1}^{(m)}(j) s_l(i)$
  else
    $w_l^{(m)}(0) = \sum_{i=0}^{\left\lfloor \frac{\tau_{l-1}}{\Delta} \right\rfloor} w_{l-1}^{(m)}(i)$
  for $k = 1 \to \sum_{j=l}^{n+1} \left\lfloor \frac{\tau_j}{\Delta} \right\rfloor$ **do**
    if $m = y_l$ **then**
      $w_l^{(m)}(k) = \sum\limits_{i=0}^{kv^{(m)} + \left\lfloor \frac{\tau_{l-1} v^{(m)}}{\Delta} \right\rfloor} w_{l-1}^{(m)}\left(k + \left\lfloor \frac{\tau_{l-1}}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor\right) s_l(i)$
    else
      $w_l^{(m)}(k) = w_{l-1}^{(m)}(k + \left\lfloor \frac{\tau_{l-1}}{\Delta} \right\rfloor)$
for $m = 1 \to M$ **do**
  if $m = y_n$ **then**
    $\mathsf{E}W_{n+1}^{(m)} = \mathsf{E}W_n^{(m)} + \frac{\mathsf{E}S_n}{v^{(m)}} - \tau_n + \sum\limits_{i=0}^{\left\lfloor \frac{\tau_n v^{(m)}}{\Delta} \right\rfloor} \sum\limits_{j=0}^{\left\lfloor \frac{\tau_n}{\Delta} - \frac{i}{v^{(m)}} \right\rfloor} \left(\tau_n - \frac{(i+j)\Delta}{v^{(m)}}\right) w_n^{(m)}(j) s_n(i)$
  else
    $\mathsf{E}W_{n+1}^{(m)} = \mathsf{E}W_n^{(m)} - \tau_n + \sum\limits_{i=0}^{\left\lfloor \frac{\tau_n}{\Delta} \right\rfloor} (\tau_n - i\Delta) w_n^{(m)}(i)$
$y_{n+1} = \operatorname{argmin}_{1 \leqslant m \leqslant M} \left(\mathsf{E}W_{n+1}^{(m)} + \frac{\mathsf{E}S_{n+1}}{v^{(m)}}\right)$
return $y_{n+1}, \mathsf{E}W_{n+1}^{(1)}, \ldots \mathsf{E}W_{n+1}^{(M)}$

Вычислительная сложность[8] этого алгоритма растет с увеличением числа поступивших заданий. В отсутствие больших вычислительных мощностей или возможностей распараллеливания[9], время, затрачиваемое на принятие очередного решения может выйти за рамки всякого разумного представления о быстродействии. Вместе с тем, *Алгоритм I* является отправной точкой для всевозможных изменений в расчете на увеличение эффективности. Далее речь

[8] Которую можно оценить, например, в терминах необходимого числа умножений (см. [265, Section 5]).

[9] См. обсуждение на стр. 157.

пойдет об одной из наилучших найденных модификаций этого алгоритма, которая, имея заметно меньшую вычислительную сложность, полноценно реализует его основной замысел.

Зададимся некоторым $0 < \Delta \ll 1$ и обозначим через $\{\tilde{s}_{n,m}(k), k = 0,1,\dots\}$ распределение на $\mathcal{X}_\Delta = \{0, \Delta, 2\Delta, \dots\}$, аппроксимирующее распределение сл. в. $S_n^{(m)} = S_n/v^{(m)}$ т. е.

$$\tilde{s}_{n,m}(k) = B_n\Big((k+0.5)v^{(m)}\Delta\Big) - B_n\Big((k-0.5)v^{(m)}\Delta\Big),\ k = 0,1,\dots.$$

Для заранее заданного $\alpha_{n,m} \in (0,1)$ положим $K_{n,m} = \min(k : \sum_{i=0}^{k} \tilde{s}_{n,m}(i) \geqslant \alpha_{n,m})$. Будем считать, что сл. в. $S_n^{(m)}$ может принимать только конечное число значений[10] из множества $\{0,\dots,K_{n,m},K_{n,m}+1\}$. Для этого назначим максимальному значению вероятность, равную $\Big(1 - \sum_{k=0}^{K_{n,m}} \tilde{s}_{n,m}(k)\Big)$. Обозначим теперь через $\{w_n^{(m)}(k), k = 0,1,\dots\}$ и $\{\tilde{w}_n^{(m)}(k), k = 0,1,\dots\}$ распределения на $\mathcal{X}_\Delta$, аппроксимирующие соответственно распределения сл. в. $W_n^{(m)}$ и $W_n^{(m)} + \mathbf{1}_{(m=y_n)}S_n^{(m)}$. Выберем такое $\varepsilon \in [0,1)$, что найдутся положительные константы

$$\begin{aligned} K_n^{(m)} &= \operatorname{argmax}_{k\geqslant 0}\Big(w_n^{(m)}(k) > \varepsilon\Big), \\ \widetilde{K}_n^{(m)} &= \operatorname{argmax}_{k\geqslant 0}\Big(\tilde{w}_n^{(m)}(k) > \varepsilon\Big), \\ L_n^{(m)} &= \operatorname{argmax}_{k\geqslant 0}\left(\tilde{s}_{n,m}(k) > \varepsilon\right). \end{aligned}$$

Наконец положим $\widehat{K}_n^{(m)} = \operatorname{argmin}_{k\geqslant 0}\Big(\sum_{i=0}^{k} \tilde{w}_n^{(m)}(i) \geqslant \alpha_{n,m}\Big)$. Повторяя рассуждения, которые использовались для вывода (3.5)–(3.8), но теперь с учетом того, что все сл. в. должны принимать только конечное число значений, получаем новые соотношения для распределения незаконченной работы в каждом сервере в момент поступления $(n+1)$-го задания:

$$\tilde{w}_n^{(m)}(k) = \begin{cases} \begin{cases} \sum\limits_{i=\max(0,k-L_n^{(m)})}^{\min(k,K_n^{(m)})} w_n^{(m)}(i)\tilde{s}_{n,m}(k-i), & \mathbf{1}_{(m=y_n)} = 1, \\ w_n^{(m)}(k), & \mathbf{1}_{(m=y_n)} = 0, \end{cases} & ,\ 0 \leqslant k \leqslant \widehat{K}_n^{(m)}, \\ 0, & \text{иначе}, \end{cases} \tag{3.10}$$

[10] Перейти к дискретному распределению с конечным числом значений можно и по-другому, начав с нахождения такого минимального $X > 0$, что $\int_0^X dB_n(v^{(m)}x) \geqslant \alpha_{n,m}$.

$$w_{n+1}^{(m)}(0) \;=\; \sum_{i=0}^{\min(\lfloor \frac{\tau_n}{\Delta} \rfloor, \widetilde{K}_n^{(m)})} \tilde{w}_n^{(m)}(i), \tag{3.11}$$

$$w_{n+1}^{(m)}(k) \;=\; \tilde{w}_n^{(m)} \left(k + \left\lfloor \frac{\tau_n}{\Delta} \right\rfloor\right),\ 1 \leqslant k \leqslant \widetilde{K}_n^{(m)} - \left\lfloor \frac{\tau_n}{\Delta} \right\rfloor,\ \widetilde{K}_n^{(m)} > \left\lfloor \frac{\tau_n}{\Delta} \right\rfloor. \tag{3.12}$$

Перейдя к новым аппроксимирующим распределениям, вычисляемым по (3.10)–(3.12), заменим и прежнее правило диспетчеризации (3.2) на новое:

$$y_{n+1} = \operatorname{argmin}_{1 \leqslant m \leqslant M} \left(\mathsf{E}\tilde{W}_{n+1}^{(m)} + \theta \cdot \mathsf{E}\tilde{S}_{n+1}^{(m)}\right),\ n \geqslant 0, \tag{3.13}$$

где $\theta \in [0,1]$ — наперед заданное число, и[11]

$$\mathsf{E}\tilde{W}_{n+1}^{(m)} = \sum_{k=0}^{\widehat{K}_n^{(m)}} k w_{n+1}^{(m)}(k), \quad \mathsf{E}\tilde{S}_{n+1}^{(m)} = \sum_{k=0}^{K_{n+1,m}+1} k \tilde{s}_{n+1,m}(k).$$

Формальное описание нового алгоритма выбора управления для задания, поступившего в момент t_{n+1}, представлено ниже.

[11] Для расчета $\mathsf{E}\tilde{W}_{n+1}^{(m)}$ можно воспользоваться и приведенным выше рекуррентным соотношением, которое, с учетом новых построений, имеет вид:

$$\mathsf{E}\tilde{W}_{n+1}^{(m)} = \mathsf{E}\tilde{W}_n^{(m)} + \mathbf{1}_{(m=y_n)}\mathsf{E}\tilde{S}_n^{(m)} - \tau_n + \sum_{i=0}^{\min(\lfloor \frac{\tau_n}{\Delta} \rfloor, \widetilde{K}_n^{(m)})} (\tau_n - i\Delta)\, \tilde{w}_n^{(m)}(i).$$

Алгоритм II. Псевдокод алгоритма выбора управления для задания, поступившего в момент t_{n+1}, $n \geqslant 1$

---

for $m = 1 \to M$ **do**
  for $k = 0 \to K_{n+1,m}$ **do**
    $\tilde{s}_{n+1,m}(k) = B_{n+1}\big((k+0.5)v^{(m)}\Delta\big) - B_{n+1}\big((k-0.5)v^{(m)}\Delta\big)$
  $\tilde{s}_{n+1,m}(K_{n+1,m}+1) = 1 - \sum\limits_{k=0}^{K_{n+1,m}} \tilde{s}_{n+1,m}(k)$
  if $m = y_n$ **then**
    for $k = 0 \to \widehat{K}_n^{(m)}$ **do**
      $\tilde{w}_n^{(m)}(k) = \sum\limits_{i=\max(0,k-L_n^{(m)})}^{\min(k,K_n^{(m)})} w_n^{(m)}(i)\tilde{s}_{n,m}(k-i)$
  else
    $\tilde{w}_n^{(m)}(k) = w_n^{(m)}(k)$
  $w_{n+1}^{(m)}(0) = \sum\limits_{i=0}^{\min(\lfloor \frac{\tau_n}{\Delta} \rfloor, \widetilde{K}_n^{(m)})} \tilde{w}_n^{(m)}(i)$
  if $\widetilde{K}_n^{(m)} > \lfloor \frac{\tau_n}{\Delta} \rfloor$ **then**
    for $k = 1 \to \widetilde{K}_n^{(m)} - \lfloor \frac{\tau_n}{\Delta} \rfloor$ **do**
      $w_{n+1}^{(m)}(k) = \tilde{w}_n^{(m)}\left(k + \lfloor \frac{\tau_n}{\Delta} \rfloor\right)$
  $\mathsf{E}\tilde{W}_{n+1}^{(m)} = \sum\limits_{k=0}^{\widehat{K}_n^{(m)}} k w_{n+1}^{(m)}(k)$
  $\mathsf{E}\tilde{S}_{n+1}^{(m)} = \sum\limits_{k=0}^{K_{n+1,m}+1} k\tilde{s}_{n+1,m}(k)$
$y_{n+1} = \operatorname{argmin}_{1 \leqslant m \leqslant M} \left(\mathsf{E}\tilde{W}_{n+1}^{(m)} + \theta\, \mathsf{E}\tilde{S}_{n+1}^{(m)}\right)$
return $y_{n+1}, w_{n+1}^{(m)}(\cdot), \tilde{s}_{n+1,m}(\cdot), K_{n+1}^{(m)}, \widetilde{K}_{n+1}^{(m)}, L_{n+1}^{(m)}$

Выходными данными *Алгоритма II* являются номер сервера y_{n+1}, на который следует отправить $(n+1)$-е задание, (усеченные) распределения незаконченной работы в каждом из серверов в момент t_{n+1} и времен обслуживания, и числа $K_{n+1}^{(m)}$, $\widetilde{K}_{n+1}^{(m)}$, $L_{n+1}^{(m)}$. По сравнению с *Алгоритмом I*, дополнительными входными данными для нового алгоритма являются константы ε, $\alpha_{n,m}$, и θ. Варьирование значения ε позволяет изменять число компонент аппроксимирующих распределений. Безопасным выбором[12] является $\varepsilon = 0$: он избавляет от циклов, в которых суммируются ряды из нулевых слагаемых, что ускоряет работу алгоритма без ущерба для точности. Значения квантилей $\alpha_{n,m}$ необходимо подбирать вручную, исходя из специфики распределений сл. в. S_n. Для нахождения же постоянного коэффициента θ, который зависит, вообще говоря,

[12] Какое-либо общее правило для выбора приемлемых значений $\varepsilon \in (0,1)$ сформулировать не удается. В каждой конкретной задаче, если есть неохбодимость кратного увеличения скорости работы алгоритма (за счет точности получаемых оценок), его приходится искать методом проб и ошибок.

от исходных параметров и от выбранной целевой функции, можно привлекать специальные методы оптимизации на имитируемых траекториях[13].

Следующий параграф начинается с набора численных примеров[14], свидетельствующих о том, что для рассматриваемых частично наблюдаемых систем с параллельным обслуживанием предложенные алгоритмы диспетеризации по полной предыстории являются равномерно наилучшими. Почти во всем диапазоне изменений значений исходных параметров системы они[15] позволяют уменьшить стационарное среднее время пребывания по сравнению со всеми[16] ранее известными из научной литературы стратегиями. Когда улучшение целевой функции невозможно[17], новые алгоритмы приводят к тем же значениям, что и наилучшие из ранее извесных.

3.2 Примеры и дополнения

Начнем с простейшего примера. Пусть система состоит всего из двух серверов суммарной производительности 1, причем $v^{(1)} = 2/3$ и $v^{(2)} = 1/3$. Предположим, что входящий поток — пуассоновский с интенсивностью λ, а распределение размера заданий — экспоненциальное со средним 1. Таким образом, загрузка системы ρ совпадает с λ. В таблице 3 даны значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе

[13] Применимы адаптивные алгоритмы для управления частично наблюдаемыми марковскими цепями [4] (см. также [3]) и алгоритмы эволюционной оптимизации [477; 478]. Последние, однако, из-за особенностей задачи несильно отличаются от ручного перебора.

[14] С их помощью можно также составить некоторое представление и о вычислительной сложности *Алгоритма II*. Для получения третьей значащей цифры требовалось "пропустить" через систему порядка 10^7–10^9 заданий и, значит, столько же запусков алгоритма. При этом длительность каждого эксперимента не превосходила 10 минут современного персонального компьютера.

[15] Вероятно, идея, лежащая в их основе, может позволить улучшить любую из стандартных стационарных характеристик.

[16] Здесь предполагается, что все возможные стратегии исчерпываются в рамках двух описанных выше подходов (см. стр. 19). Но, строго говоря, это не совсем так. Например, если известно, что оптимальной является программная стратегия $(11011011101101101110)^\infty$, то существует другая стратегия, "не укладывающаяся" ни в один из двух описанных выше подходов — это рандомизированная стратегия с 2^{19} параметрами. Для ее применения необходимо оценивать $2^{19}-1$ параметров, что едва ли выполнимо. Поэтому сделанное предположение исключает лишь совсем причудливые стратегии.

[17] Что, если и случается, то либо в случае очень низкой, либо очень высокой загрузки.

при различных значениях загрузки ρ и стратегиях RND, PROG и AA. Значения параметров стратегий (см. таблица 4) при каждом значении загрузки были выбраны следующим образом: для RND — как решение задачи минимизации (5), для PROG и AA[18] — как результат оптимизации на имитируемых траекториях.

Таблица 3 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметров стратегий приведены в таблице 4

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1.76 (1.76)	2.58 (2.63)	3.77 (3.87)	6.39 (6.56)	19.4 (20)
PROG-opt	1.76 (1.76)	2.41 (2.45)	3.22 (3.87)	5.17 (5.29)	15.0 (15.17)
AA	1.738 (1.76)	2.28 (2.36)	3.13 (3.23)	5.1 (5.18)	14.92 (15.43)

Таблица 4 — Значения параметров стратегий из таблицы 3

ρ		0.1	0.3	0.5	0.7	0.9
RND-opt		1	0.855	0.784	0.701	0.676
PROG-opt		1	0.7684	0.7076	0.6825	0.6734
AA	Δ	0.1	0.25	0.5	0.3	0.5
	ε	0	0	0	0	0
	$\alpha_{n,m}$	0.99	0.95	0.95	0.9	0.5
	θ	0.85	0.67	0.46	0.39	0.39

Как видно из таблицы 3, при любом значении загрузки новый алгоритм наилучшим образом оптимизирует значение стационарного среднего времени пребывания задания в системе. Рандомизированная стратегия, которая не использует никаких наблюдений, ожидаемо является наименее эффективной из трех. Наилучшая из ранее известных — программная стратегия — уступает новому решению, хотя и незначительно[19]: максимальный относительный проигрыш составляет примерно 5%. В связи с этим интересно сразу же посмотреть

[18] В этом и всех остальных приводимых ниже примерах этого пунтка диспетчеризация AA реализовывалась по *Алгоритму II*.

[19] Этот экспериментальный факт является примечательным и еще раз подтверждает (уже известное из литературы обстоятельство (см., например, [207, С. 28–36])), что программные стратегии, несмотря на свою простоту, могут быть очень эффективными.

на то, как ранжируются стратегии при другом целевом функционале. В следующей таблице (см. таблица 5) приведены значения стационарного среднего (и стандартного отклонения) времени ожидания заданием начала обслуживания.

Таблица 5 — Значения стационарного среднего (и стандартного отклонения) времени ожидания заданием начала обслуживания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметров стратегий приведены в таблице 6

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	0.264 (0.93)	0.88 (1.91)	1.92 (3.27)	4.57 (6.16)	24.43 (29)
PROG-opt	0.12 (0.62)	0.49 (1.4)	1.27 (2.51)	3.19 (4.78)	13.04 (15)
AA	0.061 (0.47)	0.427 (1.36)	1.21 (2.51)	3.17 (4.87)	12.96 (15)

Таблица 6 — Значения параметров стратегий из таблицы 5

ρ		0.1	0.3	0.5	0.7	0.9
RND-opt		1	0.87	0.76	0.72	0.71
PROG-opt		0.74	0.7	0.7	0.68	0.67
AA	Δ	0.1	0.2	0.3	0.5	0.5
	ε	0	0	0	0	0
	$\alpha_{n,m}$	0.95	0.95	0.95	0.9	0.9
	θ	0.1	0.1	0.13	0.05	0.21

По приведенным в таблице 5 значениям видно, что смена целевого функционала не влияет качественно на сформулированное выше соотношение между тремя стратегиями, но влияет количественно. При оптимизации среднего времени ожидания начала обслуживания (в отличие от среднего времени пребывания) новый алгоритм (т. е. диспетчеризация, учитывающая предысторию) оказывается вне конкуренции: в приведенном примере при малой загрузке наблюдается относительный выигрыш почти в 50% по сравнению с наилучшей из ранее известных стратегий. Здесь уместо еще раз подчеркнуть, что указанные в таблицах 3 и 5 значения функционалов при стратегиях RND и PROG являются неулучшаемыми[20]. При этом, значения параметров нового алгорит-

[20] Для программной стратегии такое заключение делается здесь по результатам вычислительных экспериментов.

ма, указанные в таблицах 4 и 6, не являются оптимальными. Таким образом, имеется потенциальная возможность получить еще большую отдачу от его применения.

Имея теперь пример того, что в полностью марковском случае[21] новая стратегия является оптимальной, посмотрим на то, как влияют характеристики входящего потока и распределения размера заданий на соотношения между стратегиями. Рассмотрим систему с $M = 9$ серверами различной производительности. Положим[22]

$$v^{(1)} = 0.9,\ v^{(2)} = 1,\ v^{(3)} = 1.1,$$
$$v^{(4)} = 2.9,\ v^{(5)} = 3,\ v^{(6)} = 3.1,$$
$$v^{(7)} = 6.9,\ v^{(8)} = 7,\ v^{(9)} = 7.1.$$

Пусть в систему поступают однородные задания, средний размер которых равен единице. Тогда загрузка системы ρ равна $\lambda/\sum_{m=1}^{9} v^{(m)} = \lambda/33$. Предположим, что размер заданий имеет распределение $B(x) = \mathsf{P}\{|S| < x\}$, где сл. в. S может иметь

- экспоненциальное распределение, т. е. $B'(x) = e^{-x}$, $x \geqslant 0$ (коэффициент вариации $C_B = 1$),
- равномерное распределение на интервале $[-1.11,2.41]$ (коэффициент вариации $C_B = 0.45$), т. е.

$$B'(x) = \frac{1}{3.52}\left(2 \cdot \mathbf{1}_{(0<x\leqslant 1.11)} + \mathbf{1}_{(1.11<x\leqslant 2.41)}\right),\ x \in [0,2.41],$$

- нормальное распределение со средним 0.195 и дисперсией 1.533 (коэффициент вариации $C_B = 0.57$), т. е.

$$B'(x) = \frac{1}{\sqrt{3.066\pi}}\left(e^{-\frac{(x-0.195)^2}{3.066}} - e^{-\frac{(x+0.195)^2}{3.066}}\right),\ x \geqslant 0,$$

[21] И не только в случае двух серверов; по этому поводу см. [307].

[22] Основой выбора подобного размера и состава системы служат следующие рассуждения. Это полноценная многосерверная система, в которой не возникают эффекты, свойственные слишком маленьким и слишком большим системам. В частности, при $M = 2$ известна оптимальная стратегия (это программная стратегия), а при $M > 2$ оптимальная стратегия неизвестна. При очень большом M целесообразнее искать приближенные алгоритмы диспетчеризации, а не точные (отметим, что некоторые результаты в этом направлении имеются в литературе (см. [224])). Наконец, поскольку исходные параметры систем приходится выбирать искусственным образом, это усложняет задачу сравнения стратегий: при ненадлежащем выборе, значения целевых функционалов могут быть очень близкими или вовсе бессмысленными.

– гиперэкспоненциальное распределение с параметрами (0.75; 1.5,0.5) (коэффициент вариации $C_B = 1.66$), т. е.

$$B'(x) = 1.125e^{-1.5x} + 0.125e^{-0.5x},\ x \geqslant 0.$$

В качестве распределения $F(x)$ входящего потока рассмотрим

– экспоненциальное распределение, т. е. $F'(x) = \lambda e^{-\lambda x}$, $x \geqslant 0$ (коэффициент вариации $C_F = 1$),

– распределение Парето с параметрами $\alpha = 2.15$, $x_m = (\alpha - 1)/(\lambda\alpha)$ (коэффициент вариации $C_F = 1.76$), т. е.

$$F'(x) = \frac{\alpha x_m^\alpha}{x^{\alpha+1}},\ x \geqslant x_m.$$

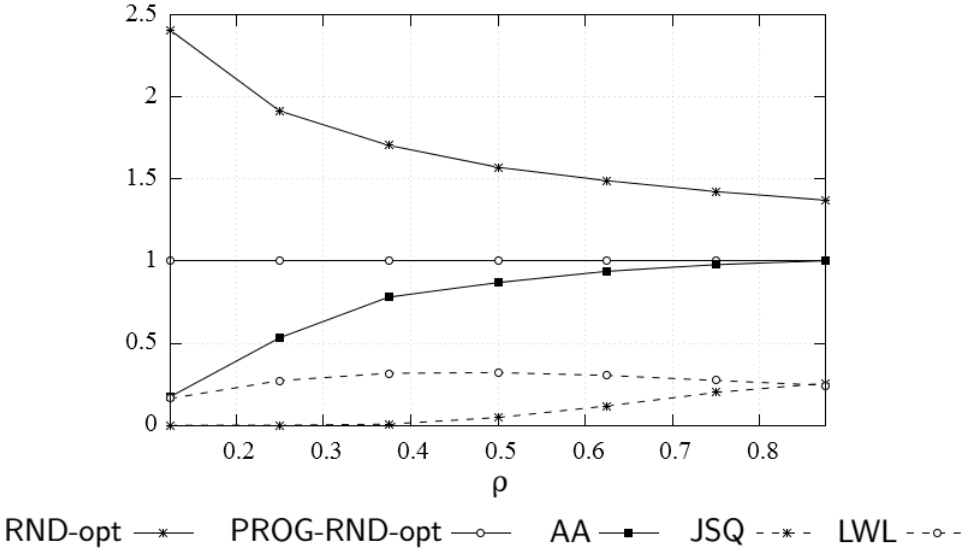
Как и в предыдущем примере, будем иметь в виду два целевых функционала: стационарное среднее время пребывания задания в системе и стационарное среднее время ожидания начала обслуживания. Кроме того, добавим к рассмотрению еще две диспетчеризации: LWL (диспетчеризация по наименьшей незаконченной работе) и JSQ (диспетчеризация по наикратчайшей очереди). Поскольку для реализации каждой из них необходимы наблюдения, интуиция подсказывает, что они должны всегда оптимизировать целевые функционалы лучше тех алгоритмов, что наблюдения не используют. Однако, как будет видно далее, это соображение (полезное хотя бы для контроля вычислений) является неверным, если во внимание принимается предложенная в диссертации диспетчеризация.

Вернемся к описанию примера. Алгоритмы LWL и JSQ являются непараметрическими. Отыскание оптимальных значений параметров для стратегий RND и PROG и AA, по существу, представляет собой отдельные задачи. Когда было возможно, для RND решалась минимизационная задача (5); в остальных случаях, значения параметров оптимизировались на имитируемых траекториях. При этом, для стратегии PROG значения параметров отождествлялись с оптимальными значениями параметров для RND. В алгоритме AA значения Δ, ε и $\alpha_{n,m}$ были зафиксированы: $\Delta = 0.1$, $\varepsilon = 0$, $\alpha_{n,m} = 0.95$ при всех n и m; постоянный коэффициент θ каждый раз подбирался на имитируемых траекториях.

Примем диспетчеризацию PROG-RND-opt за точку отсчета. На рисунках 4—11 для всех возможных комбинаций распределений входящего потока и размера заданий приведены графики зависимостей от загрузки ρ[23] значений

[23] В диапазоне (0,0.85]. Отображение значений для всего диапазона загрузки (0,1) потребовало бы изменения масштаба по осям.

относительного стационарного среднего времени пребывания задания в системе (т. е. значений $\frac{\mathsf{E}V^{\mathsf{X}}}{\mathsf{E}V^{\mathsf{PROG\text{-}RND\text{-}opt}}}$; здесь "X" — одна из стратегий LWL, JSQ, RND-opt, PROG-RND-opt или AA) и относительного стационарного среднего времени ожидания начала обслуживания.



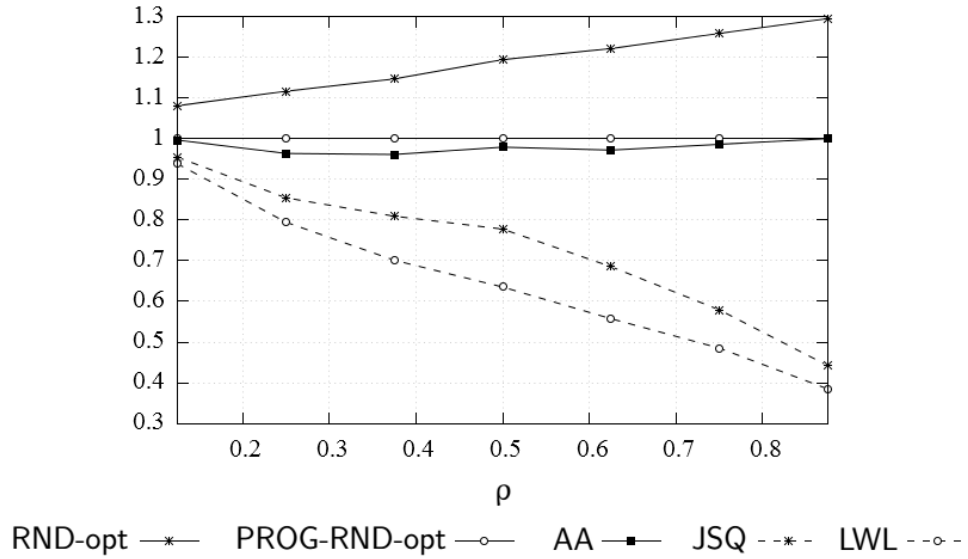


Рисунок 4 — Входящий поток — пуассоновский ($C_F = 1$), распределение размера заданий — экспоненциальное ($C_B = 1$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt

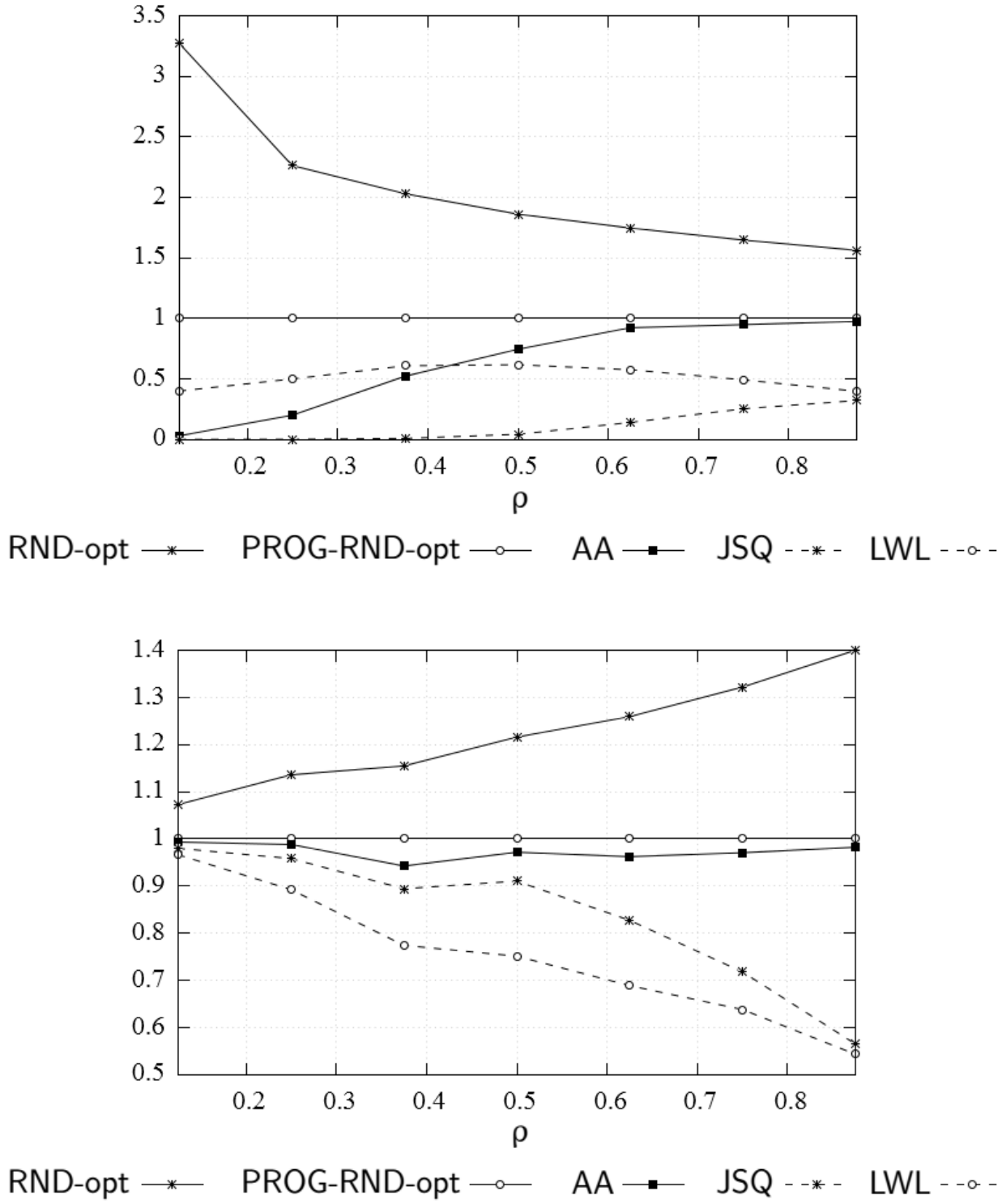


Рисунок 5 — Входящий поток — пуассоновский ($C_F = 1$), распределение размера заданий — "нормальное" ($C_B = 0.57$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt

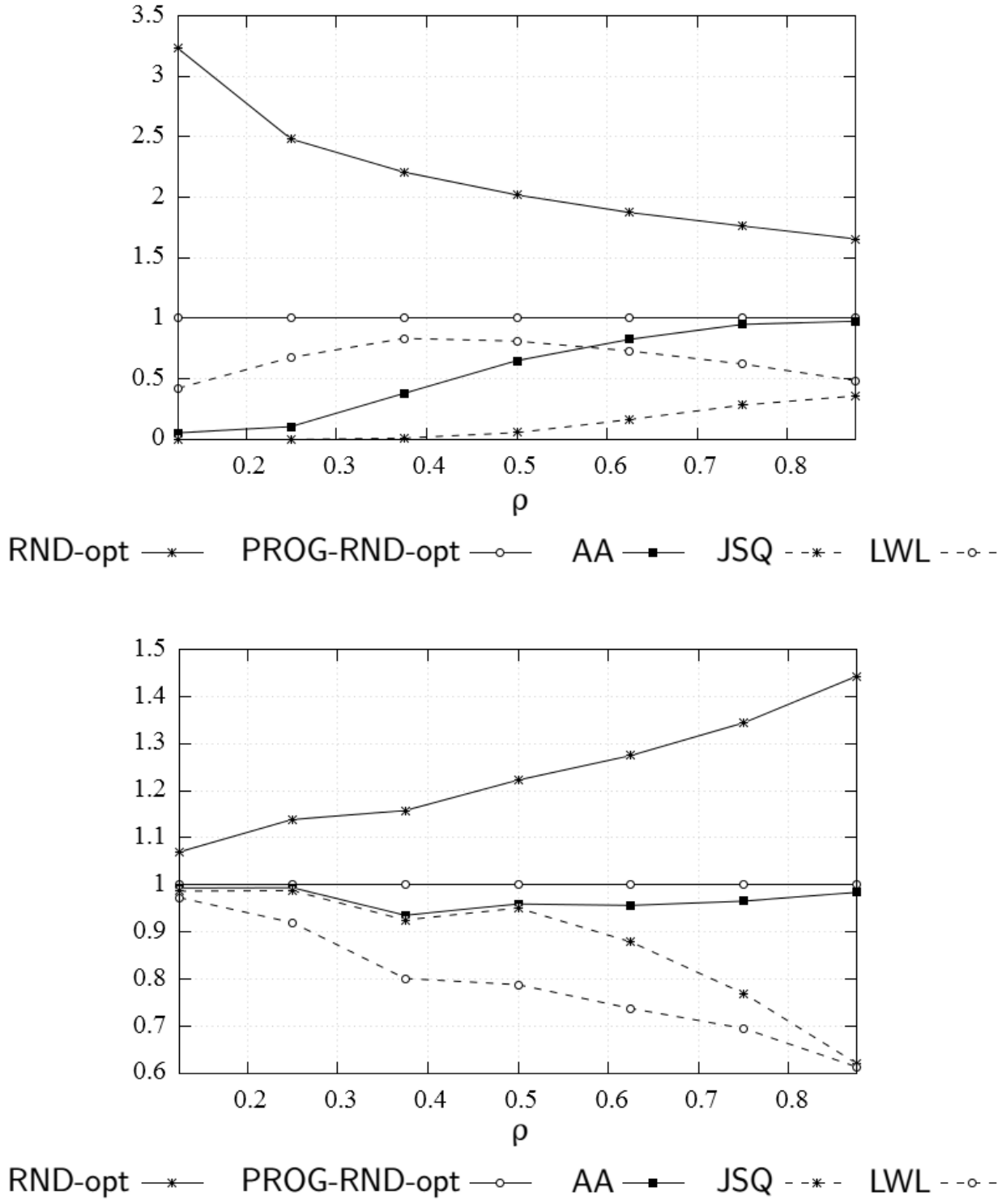


Рисунок 6 — Входящий поток — пуассоновский ($C_F = 1$), распределение размера заданий — “равномерное” ($C_B = 0.45$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt

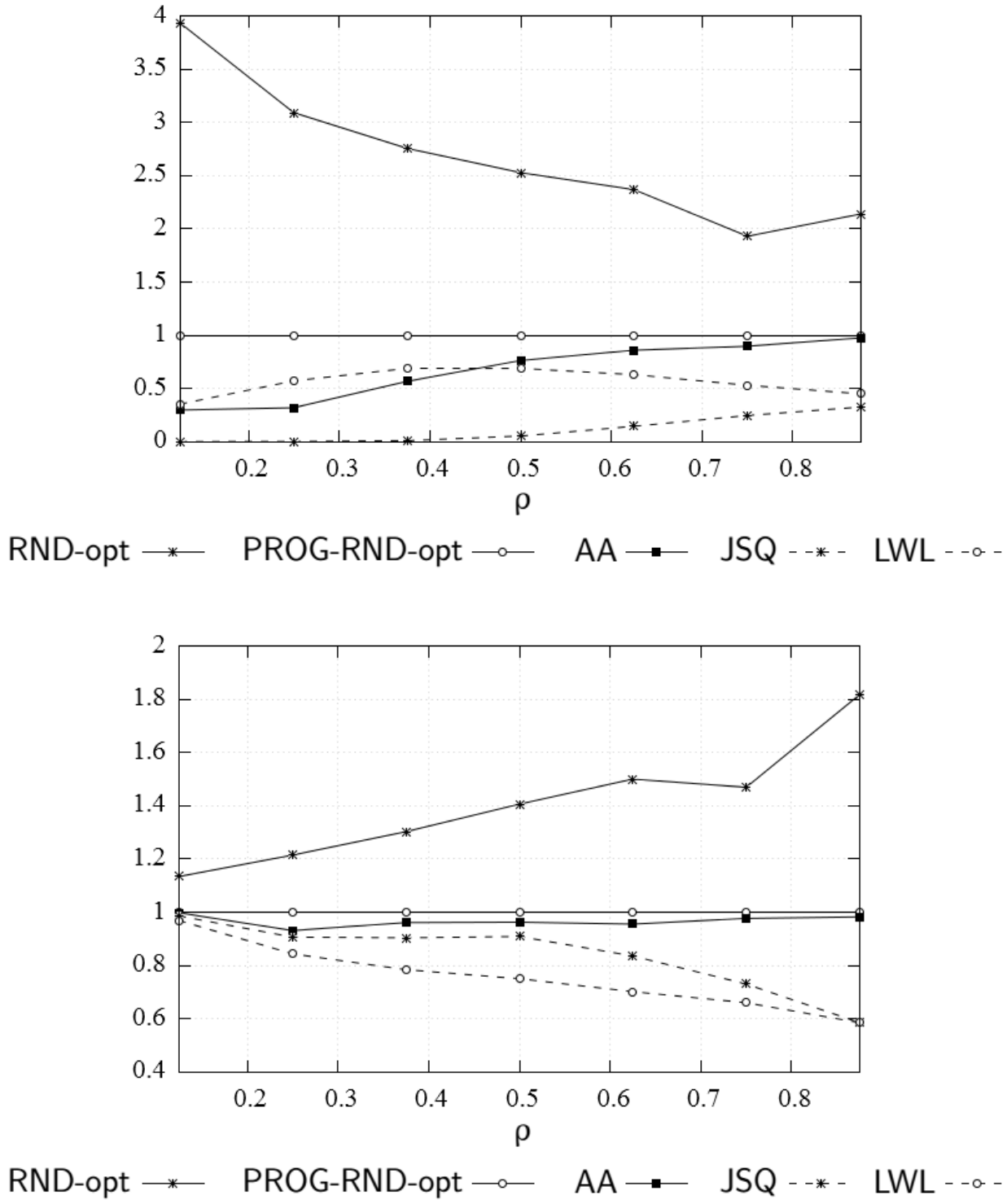
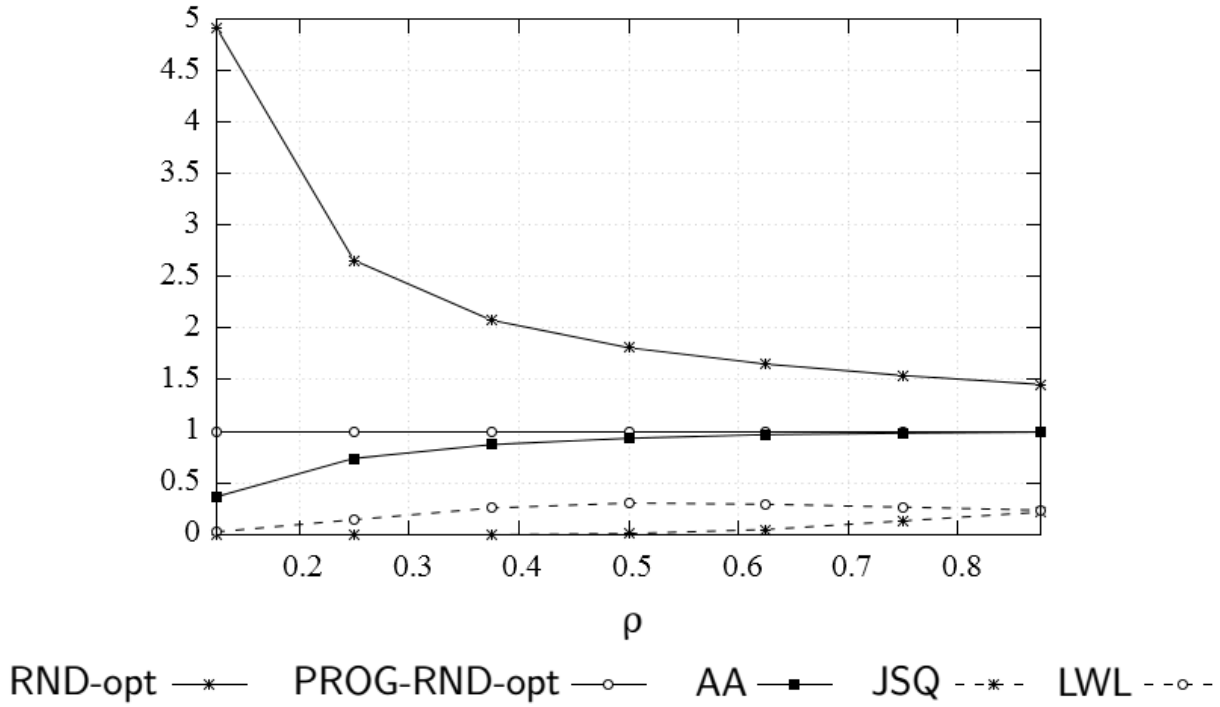


Рисунок 7 — Входящий поток — пуассоновский ($C_F = 1$), распределение размера заданий — гиперэкспоненциальное ($C_B = 1.66$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt



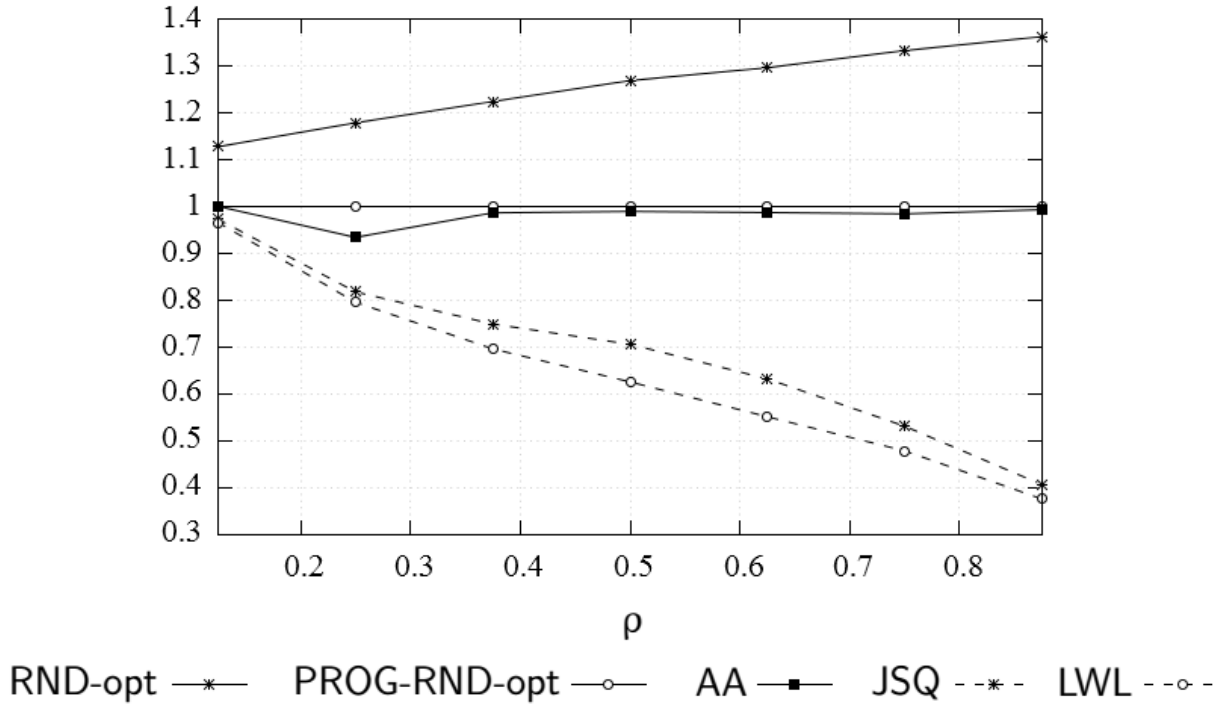
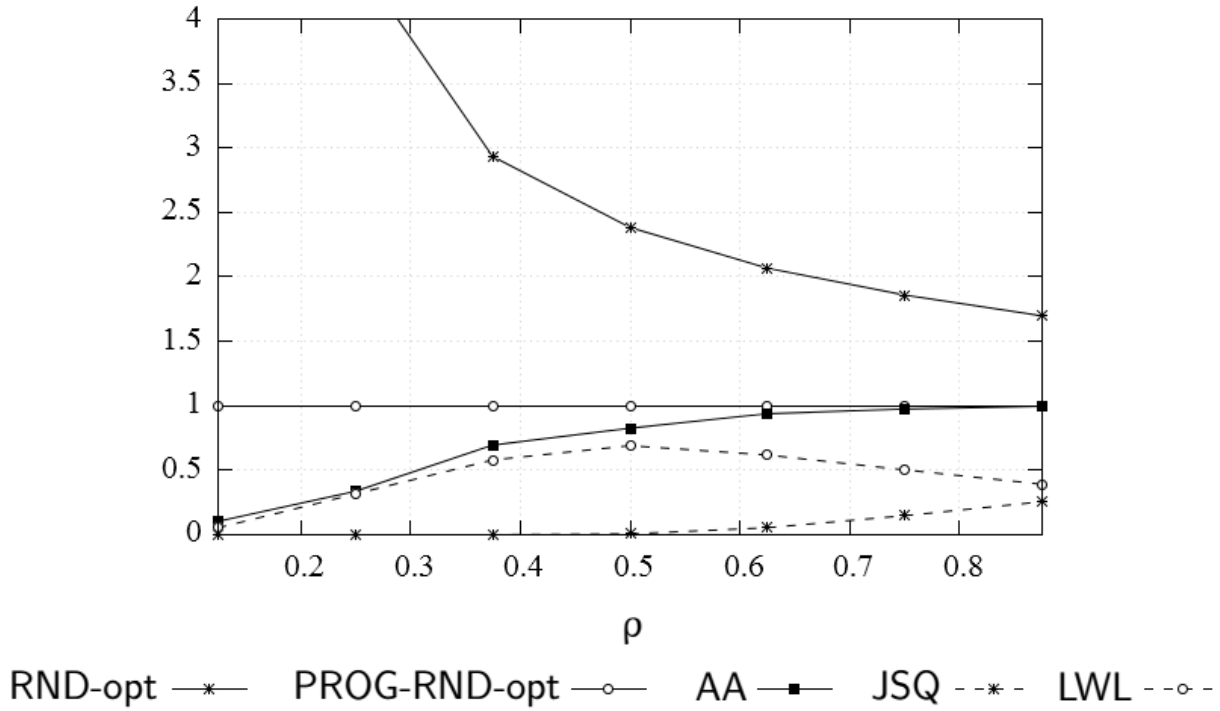


Рисунок 8 — Входящий Парето–поток ($C_F = 1.76$), распределение размера заданий — экспоненциальное ($C_B = 1$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt



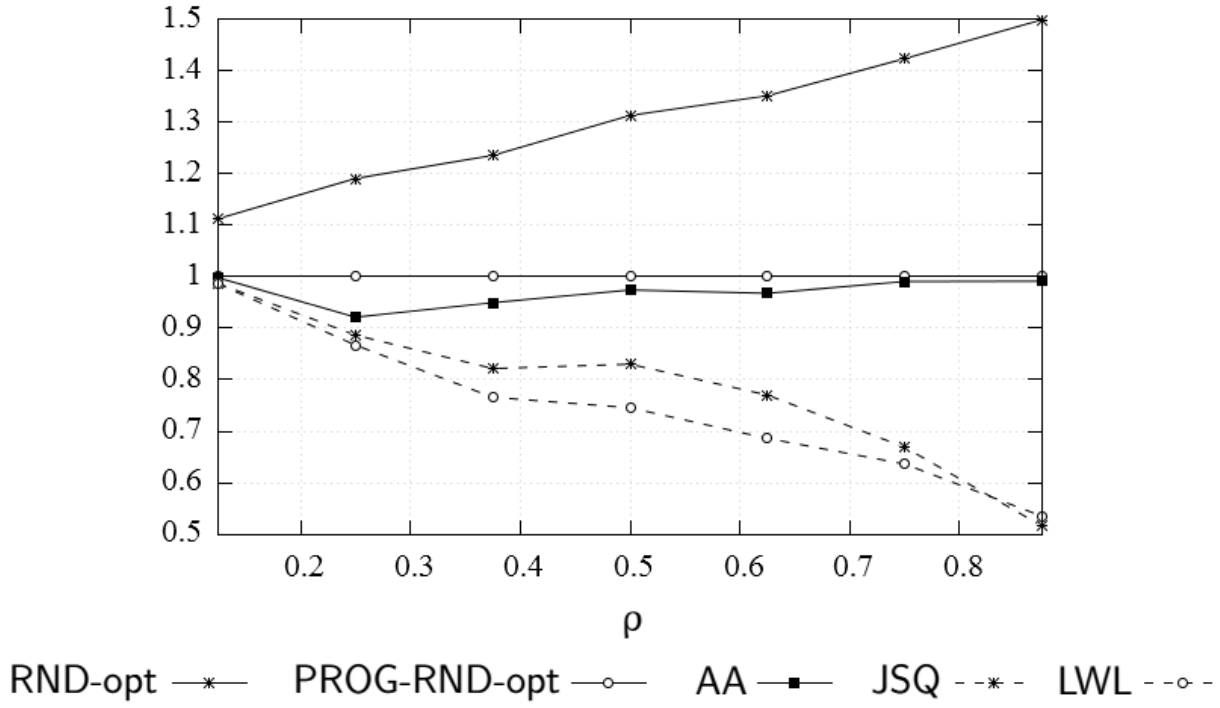
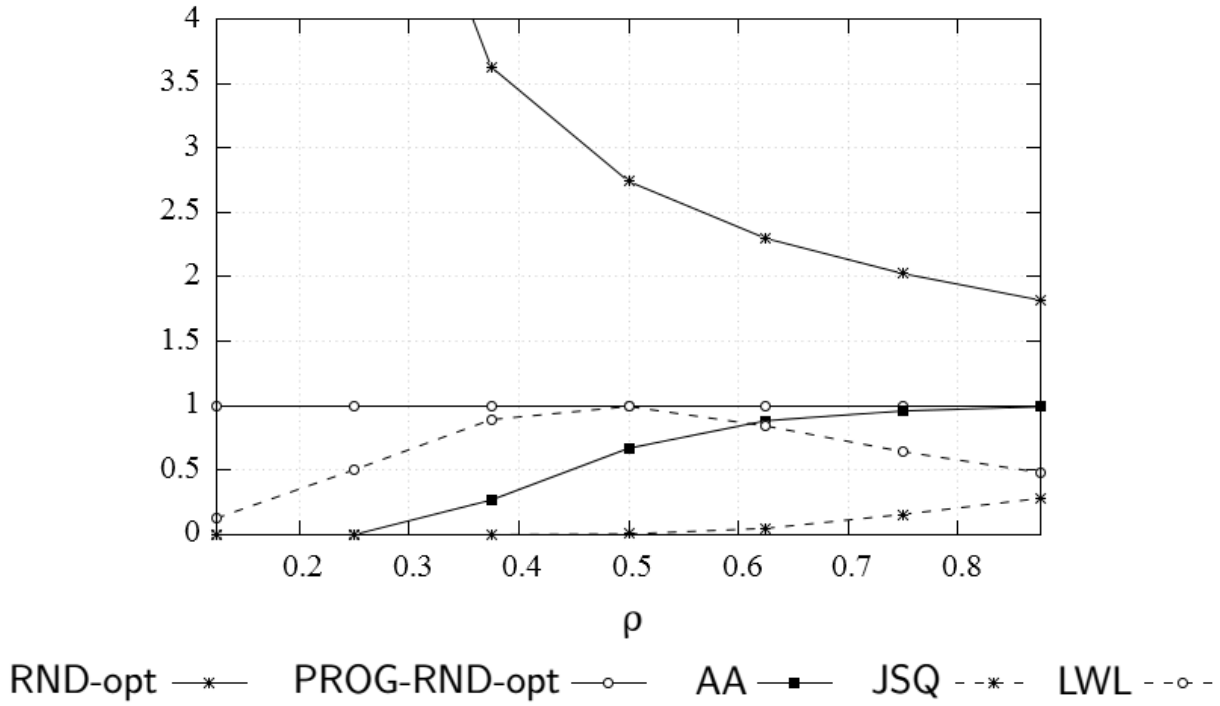


Рисунок 9 — Входящий Парето–поток ($C_F = 1.76$), распределение размера заданий — “нормальное” ($C_B = 0.57$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt



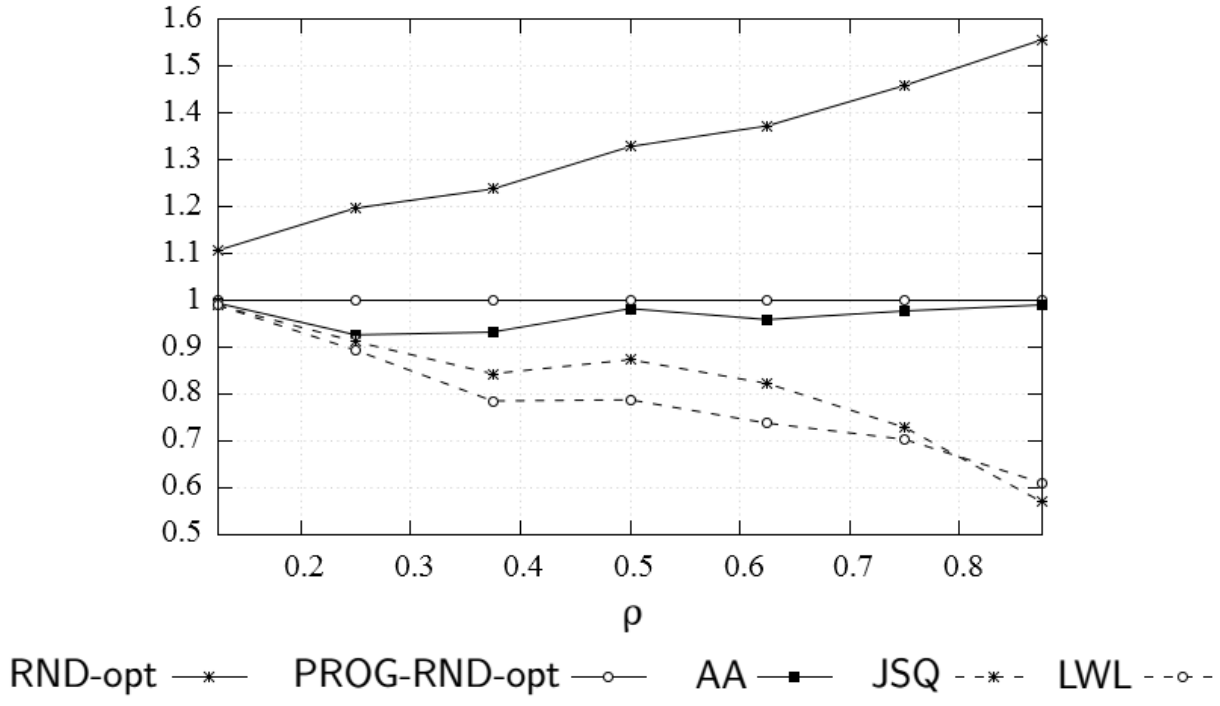
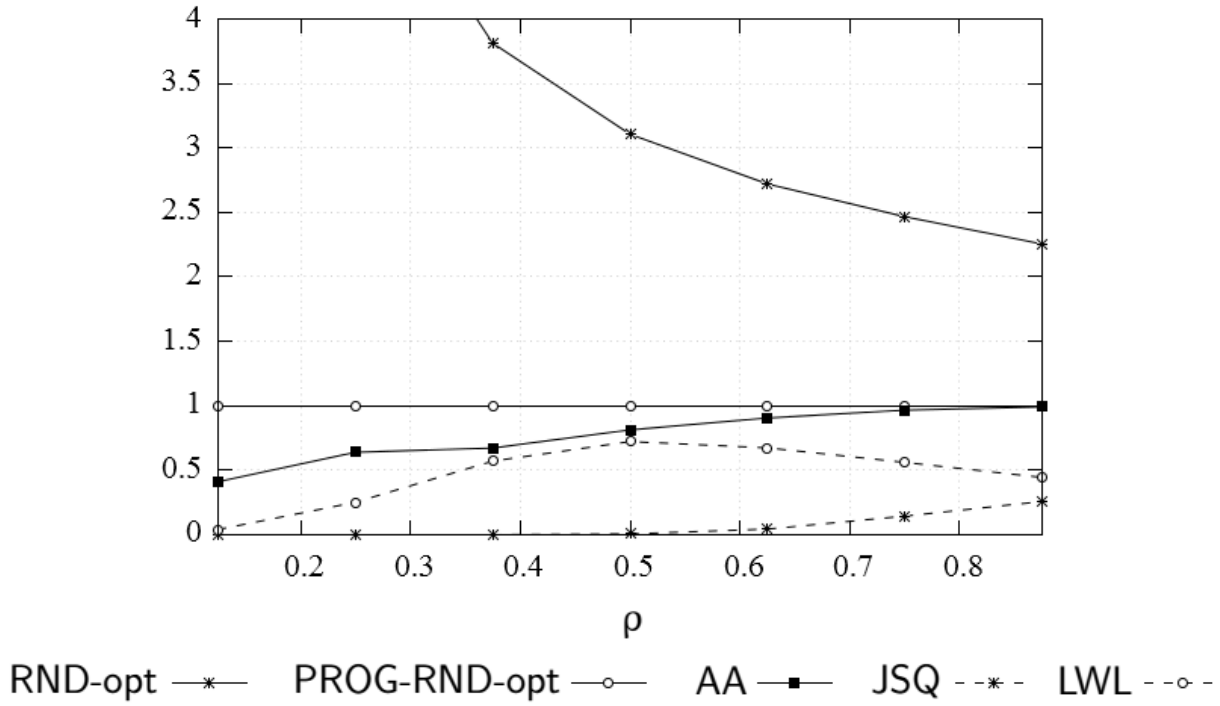


Рисунок 10 — Входящий Парето–поток ($C_F = 1.76$), распределение размера заданий — “равномерное” ($C_B = 0.45$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt



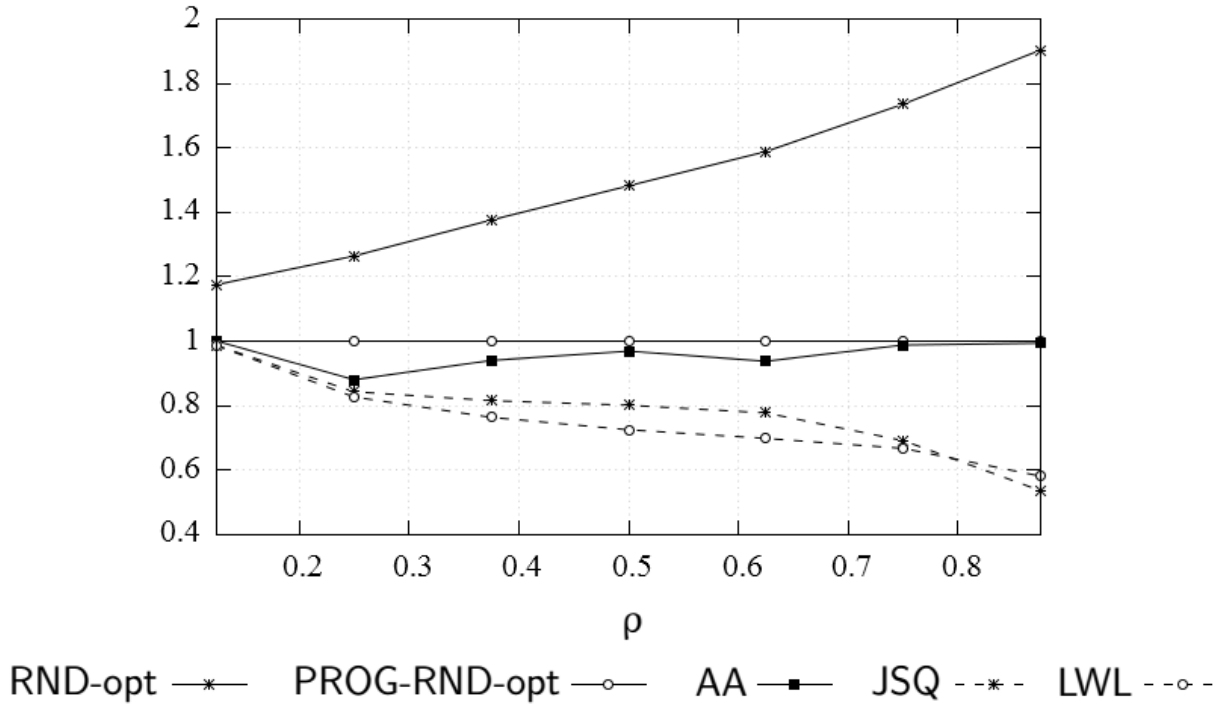


Рисунок 11 — Входящий Парето–поток ($C_F = 1.76$), распределение размера заданий — гиперэкспоненциальное ($C_B = 1.66$). Верхний рисунок — стационарное среднее время ожидания начала обслуживания. Нижний рисунок — стационарное среднее время пребывания задания в системе. Указаны значения относительно программной стратегии PROG-RND-opt

Как видно из рисунков, стратегия RND-opt, которая не использует никаких наблюдений, является наименее эффективной из всех. Новая же диспетчеризация AA (в классе диспетчеризаций, не использующих какую-либо информацию о текущем состоянии системы и размере поступающих заданий) всегда позволяет улучшить значение целевой функции. Выигрыш при разных интенсивностях зависит от свойств распределения размера задания (см. таблицу 7, в которой указаны относительные выигрыши для каждого из рассмотренных случаев).

Таблица 7 — Относительный выигрыш при стратегии AA относительно наилучшей из ранее известных стратегии PROG-RND-opt при различных значениях загрузки ρ: первое значение — стационарное среднее времени ожидания начала обслуживания, второе значение — стационарное среднее времени пребывания задания в системе (т. е. значение $100\% \times \frac{EV^{\mathsf{AA}} - EV^{\mathsf{PROG\text{-}RND\text{-}opt}}}{EV^{\mathsf{PROG\text{-}RND\text{-}opt}}}$)

ρ	0.125	0.25	0.375	0.5	0.625	0.75	0.875
рис. 4	80%/0%	50%/3.7%	22%/3.8%	13%/2.1%	6.2%/2.8%	2.1%/1.4%	0%/0%
рис. 5	97%/0.6%	80%/1.2%	48%/5.7%	26%/2.8%	7.6%/3.7%	5.1%/2.9%	2.5%/1.8%
рис. 6	94%/0.6%	89%/0.6%	62%/6.4%	35%/4%	17%/4.3%	5%/3.4%	2.5%/1.5%
рис. 7	70%/0%	68%/6.7%	43%/3.8%	24%/3.6%	14%/4.4%	10%/2.3%	2.5%/1.7%
рис. 8	63%/0%	26%/6.6%	13%/1.3%	6.8%/1%	3.3%/1.3%	1.8%/1.5%	0.8%/0.7%
рис. 9	100%/0%	66%/7.8%	30%/5%	17%/2.5%	6.4%/3.2%	2.7%/0.9%	0.5%/0.8%
рис. 10	100%/0.7%	100%/7.5%	74%/6.7%	33%/1.8%	11%/4.1%	4.2%/2.3%	0.9%/1%
рис. 11	66%/0%	36%/12%	33%/6%	19%/3.3%	9%/6.2%	3.6%/1.2%	1%/0.6%

Напомним, что в рассматриваемом примере три из четырех параметров новой стратегии AA не подвергались вообще никакой оптимизации. На этом основании можно утверждать, что возможности для улучшения результатов[24], указанных в таблице 7, не исчерпаны.

Как и графики на рисунках 4—11, значения в таблице 7 свидетельствуют о том, что с ростом загрузки преимущество новой стратегии сходит на нет. Причина этого явления, по-видимому, заключается в следующем: при высокой загрузке очереди в серверах непусты большую часть времени и новая диспетчеризация AA (т. е. правило (3.2)) оказывается недостаточно чувствительной для того, чтобы уловить разброс значений незаконченной работы в серверах, возникающий за время между двумя последовательными поступлениями. Судя по вычислительным экспериментам, в пределе при $\rho \to 1$ новая стратегия

[24] В том числе и с помощью быстрых алгоритмов (см. [479]).

доставляет такой же выигрыш[25], как и лучшая из ранее известных стратегий — программная стратегия (со значениями параметров, выбранными оптимальным образом).

Присмотримся теперь на рисунках 4—11 к графикам целевых функционалов при стратегиях LWL и JSQ. При оптимизации (в обычных условиях[26]) стационарного среднего времени пребывания задания в системе, как LWL, так и JSQ вне зависимости от загрузки предпочтительнее любой из диспетчеризаций, не использующих наблюдения за системой. Однако наблюдается следующий эффект: с уменьшением дисперсии размера задания новая стратегия AA ведет себя стабильно, тогда как эффективность JSQ (относительно AA) падает в области низкой и средней загрузок. Вычислительные эксперименты с предельным случаем (т. е. вырожденным распределением размера заданий) показывают, что новый алгоритм AA может быть лучше JSQ (но не LWL). Конкретный пример и интуитивное объяснение этого экспериментального факта будет предложено в последующих параграфах[27] (см. стр. 185), а сейчас обратимся к другому функционалу — стационарному среднему времени ожидания заданием начала обслуживания. И здесь новая диспетчеризация дает контринтуитивный результат: в четырех из восьми рассмотренных случаев (см. рисунки 5, 6, 7, 10) стратегия AA лучше основанной на наблюдениях диспетчеризации LWL по наименьшей незаконченной работе (но никогда не лучше диспетчеризации по наикратчайшей очереди JSQ). Объяснение этого эффекта, который наблюдается в различных вычислительных экспериментах, остается открытым вопросом.

Представленные, а также другие вычислительные эксперименты (с результатами которых можно ознакомиться в [304; 306; 307; 309]) свидетельствуют о том, что для рассматриваемых частично наблюдаемых стохастических систем с параллельным обслуживанием предложенный подход к управлению всегда приносит выигрыш по сравнению со всеми другими известными из научной литературы стратегиями. Величина его зависит от целевой функции и может достигать десятков процентов (см. таблица 7). Наблюдаемое преимущество, однако, дается не бесплатно. С вычислительной точки зрения алгоритмы, реализующие новый подход (см. *Алгоритм I* и *Алгоритм II*), являются намного

[25] Объяснения этому экспериментальному факту найти не удалось.

[26] Т. е., например, в отсутствие особых свойств распределения размера заданий (например, очень тяжелых хвостов).

[27] Где будет предложен иной способ реализации диспетчеризации по предыстории.

более сложными, чем алгоритмы рандомизированной и программной стратегий. Впрочем это обстоятельство не говорит однозначно в пользу последних по ряду причин:

1. Число параметров, требующих предварительной оценки для стратегий RND и PROG, на единицу меньше общего числа серверов в системе. В новой же диспетчеризации AA число параметров[28] не превосходит четырех при любом числе серверов (см. таблицы 4 и 6). Равномерно хороших процедур поиска оптимальных значений ни для какой из упомянутых стратегий к настоящему времени не разработано.

2. Несмотря на сложность алгоритмов, реализующих новый подход к диспетчеризации (см. *Алгоритм I* и *Алгоритм II*), можно[29] считать, что они реализуют управление входящим потоком в реальном времени (см. сноску на стр. 142). Кроме того, они допускают распараллеливание вычислений. Поэтому их быстродействие можно повысить при наличии соответствующей инфраструктуры[30].

3. Изложенный аналитической подход к реализации диспетчеризации по предыстории в принципе применим всегда, когда для интересующего слу-

[28]Отметим, что, судя по вычислительным экспериментам, для самого важного параметра диспетчеризации AA — коэффициента θ — в каждой задаче существует единственное значение, доставляющее минимум целевому функционалу, причем $\theta \in (0,1)$. Кроме того, по крайней мере наиболее употребительные целевые функционалы оказываются пологими в окрестности оптимального значения θ, что (при отсутствии требования сверхвысокой точности) облегчает его нахождение.

[29]Справедливости ради необходимо сказать, что границу здесь провести довольно трудно. Можно подобрать исходные параметры так, что на принятие решения о поступающем задании будет затрачиваться столь много времени, что по его истечении очереди в серверах уже окажутся пусты (и, значит, отправлять задание надо на сервер, выбранный равновероятно из наиболее производительных). Обстоятельства, при которых возникают подобные ситуации, являются скорее исключением, чем правилом. Для диспетчеризации в них необходимы иные идеи (например, [224]), которые в диссертации не обсуждаются. Отметим также, что задача диспетчеризации по предыстории, при низкой скорости работы реализующих ее алгоритмов, имеет схожие черты с известными в литературе (см. [480–485] и http://webhome.auburn.edu/~yzs0078/AoI.html) задачами управления по "стареющей" информации. Действительно, при низкой скорости диспетчеризации каждое принятое решение, вообще говоря, не отвечает текущему состоянию системы и поэтому может быть названо устаревшим. Вопрос о возможности учета этого эффекта остается невыясненным.

[30]И соответствующих алгоритмов, разработка которых (из-за особенностей процессов, возникающих в теории очередей) — вопрос открытый; хотя литература по проблематике распалаллеливания вычислений весьма обширна (укажем для примера на [486–495]).

чайного процесса с дискретным временем удается[31] выписать рекуррентные соотношения типа[32] (3.3). В частности, имея в виду рекурсию Кифера–Вольфовица, однопроцессорные сервера можно заменить на многопроцессорные. Близость программной стратегии к оптимальной в такой системе — вопрос открытый. Свойства же оптимальности нового алгоритма от структуры системы не зависят. Кроме того, судя по вычислительным экспериментам, он гарантирует[33] результаты не хуже тех, что получаются при стратегии PROG-opt (и тем более RND-opt).

Модификации и обобщения. Для минимизации стационарного среднего времени пребывания задания в системе с учетом предыстории существует путь (по крайней мере еще один), отличный от того, что был избран в алгоритме AA. А именно, в момент поступления n-го задания можно (вместо вычисления $\mathsf{E}V_n$) пересчитывать распределение случайной величины, равной номеру сервера с минимальным числом заданий. Тогда тот сервер, которому, например, отвечает мода распределения, и будет тем, на который надлежит отправить n-е задание. Судя по вычислительным экспериментам, такой образ действий, являясь, конечно, более вычислительно затратным, может (при некоторых дисциплинах обслуживания) приводить к еще большему выигрышу, в сравнении с алгоритмом AA.

В продолжение мысли, сформулированной в предыдущем абзаце, остановимся на принципиально отличном от приведенных выше примере, который показывает, что диспетчеризация по предыстории возможна не только с опорой на вычисление значений $\mathsf{E}V_n$, но и на связанные с ними величины. Рассмотрим

[31] В иных случаях (например, когда целевой функционал связан с хвостами распределений времени пребывания [496] или когда используется какая-то экзотическая дисциплина обслуживания [497]) необходимы новые приемы; речь о них пойдет в следующем параграфе диссертации и в главе 4.

[32] Объем литературы, затрагивающий этот вопрос, огромен (см., например, список литературы в диссертации [498]). Поэтому ограничимся лишь несколькими примерами. Так в [499] получены соотношения для времен ожидания неприоритетных заявок в (неклассической) приоритетной СМО с двумя типами заявок и прямым порядком обслуживания. Рекуррентные процедуры для расчета времен ожидания в некоторых тандемных системах приводятся в [500] (см. также [501] и [60, С.34–36]). Соотношения между моментами поступления и окончания обслуживания заданий в СМО $G\,|\,G\,|\,m\,|\,\infty\,|\,\mathsf{FIFO}$ даны в [502]. Наиболее общая рекурсия (для времени ожидания и пребывания) в параллельных однолинейных СМО с дисциплиной FIFO и заданиями, допускающими определенную внутреннюю структуру, приведена в [503, Theorem 2.1] (см. также [504]). Для дисциплин, отличных от FIFO, редко удается выписать соотношения для величин, важных для решаемой задачи (см., например, [505, Lemma1], где изучена СМО $SMP\,|\,M\,|\,1\,|\,\infty\,|\,\mathsf{PS}$).

[33] Математическое доказательство этого утверждения неизвестно.

систему из двух однопроцессорных серверов производительности $v^{(1)}$ и $v^{(2)}$, каждый из которых имеет очередь неограниченной емкости. Находящиеся в них задания обслуживаются в соответствии с дисциплиной справедливого разделения процессора. Задания поступают к диспетчеру по одному, по пуассоновскому потоку, причем их размер имеет экспоненциальное распределение с параметром μ. Цель диспетчера — минимизировать стационарное среднее $\mathsf{E}V$ время пребывания задания в системе т. е. (3.1). Для выбора действия y_{n+1} ($y_{n+1} \in \{1,2\}$), принимаемого в момент t_{n+1} относительно поступившего задания, (вместо (3.13)) будем руководствоваться правилом

$$y_{n+1} = \operatorname{argmin}_{1 \leqslant m \leqslant 2} \left(\theta \cdot \mathsf{E} N_{n+1}^{(m)} \right), \ n \geqslant 0, \tag{3.14}$$

где $N_{n+1}^{(m)}$ — число заданий в сервере m в момент поступления задания с номером $n+1$ (но до прибавления задания к какому-либо серверу), а $\theta \in (0,1]$ — наперед заданное число. Вычисление y_{n+1} может быть алгоритмизировано. Действительно, выберем $0 < \Delta \ll 1$ так, чтобы $\mu v^{(m)} \Delta < 1$. Тогда $\mu v^{(m)} \Delta$ можно трактовать как вероятность окончания обслуживания задания на сервере m за малое время Δ. Для сокращения записи положим $v^{(1)} = v^{(2)} = 1$ и введем следующие обозначения[34]:

$$\begin{aligned}
&q_{ij} = C_i^j \left(\frac{\mu\Delta}{i} \right)^j \left(1 - \frac{\mu\Delta}{i} \right)^{i-j}, \ i \geqslant 1, \ j = 0,1,\ldots,i, \\
&\mathbf{q}_j^{\mathrm{T}} = (q_{j,j}, q_{j,j-1}, \ldots, q_{j,1}), \ j \geqslant 2, \\
&\mathbf{0}_n^{\mathrm{T}} = (\underbrace{0,0,\ldots,0}_{n}), \ n \geqslant 1.
\end{aligned}$$

Через $\mathbf{I}_{i,i+1}$ будем обозначать прямоугольную матрицу размерности $i \times (i+1)$, составленную из единичной матрицы $\mathbf{I}_i$ размера i, дополненной справа нулевым столбцом. Наконец, определим две вспомогательные матрицы, $\mathbf{P}_n$ и $\tilde{\mathbf{P}}_n$,

[34] Здесь уместно упомянуть работу [506].

следующим образом. Пусть $\mathbf{P}_n^{(i)}$ ($\tilde{\mathbf{P}}_n^{(i)}$) — i-я строка матрицы $\mathbf{P}_n$ ($\tilde{\mathbf{P}}_n$). Тогда

$$\mathbf{P}_1 = \begin{pmatrix} 1 & 0 \\ b & 1-b \end{pmatrix} = \begin{pmatrix} \mathbf{P}_1^{(1)} \\ \mathbf{P}_1^{(2)} \end{pmatrix},$$

$$\mathbf{P}_2 = \begin{pmatrix} \mathbf{P}_1^{(1)} & 0 \\ \mathbf{P}_1^{(2)} & 0 \\ \mathbf{q}_2^{\mathrm{T}} & q_{2,0} \end{pmatrix} = \begin{pmatrix} \mathbf{P}_2^{(1)} \\ \mathbf{P}_2^{(2)} \\ \mathbf{P}_2^{(3)} \end{pmatrix}, \quad \tilde{\mathbf{P}}_2 = \begin{pmatrix} \mathbf{P}_1^{(2)} & 0 \\ \mathbf{q}_2^{\mathrm{T}} & q_{2,0} \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{P}}_2^{(1)} \\ \tilde{\mathbf{P}}_2^{(2)} \end{pmatrix},$$

$$\mathbf{P}_3 = \begin{pmatrix} \mathbf{P}_2 & \mathbf{0}_3 \\ \mathbf{q}_3^{\mathrm{T}} & q_{3,0} \end{pmatrix} = \begin{pmatrix} \mathbf{P}_3^{(1)} \\ \mathbf{P}_3^{(2)} \\ \mathbf{P}_3^{(3)} \\ \mathbf{P}_3^{(4)} \end{pmatrix}, \quad \tilde{\mathbf{P}}_3 = \begin{pmatrix} \mathbf{P}_2^{(2)} & 0 \\ \mathbf{P}_2^{(3)} & 0 \\ \mathbf{q}_3^{\mathrm{T}} & q_{3,0} \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{P}}_3^{(1)} \\ \tilde{\mathbf{P}}_3^{(2)} \\ \tilde{\mathbf{P}}_3^{(3)} \end{pmatrix}, \ \ldots$$

т. е. матрица $\mathbf{P}_n$ получается из матрицы $\mathbf{P}_{n-1}$ путем изменения размера последней: снизу добавляется одна строка $(q_{n,n},q_{n,n-1},\ldots,q_{n,1},q_{n,0})$, справа — один нулевой столбец. Вычеркиванием из матрицы $\mathbf{P}_n$ первой строки получаем матрицу $\tilde{\mathbf{P}}_n$.

Положим $\mathbf{p}_{n,m}^{\mathrm{T}} = (P_{n,m}(0),P_{n,m}(1),\ldots,P_{n,m}(n))$, где $P_{n,m}(i)$ вероятность того, что в сервере m в момент t_n находится ровно i заданий. Пусть $0 \leqslant t_1 < \cdots < t_n < \ldots$ — последовательность моментов поступления заданий в систему. Из формулы полной вероятности, с учетом введенных обозначений, имеем[35]:

$$\mathbf{p}_{1,m}^{\mathrm{T}} = (1 - \mathbf{1}_{(m=y_1)},\mathbf{1}_{(m=y_1)}),$$
$$\mathbf{p}_{2,m}^{\mathrm{T}} = \mathbf{p}_{1,m}^{\mathrm{T}} \left(\tilde{\mathbf{P}}_1 (\mathbf{P}_1)^{\frac{t_2-t_1}{\Delta}-1} \mathbf{1}_{(m=y_1)} + \mathbf{I}_2 (1 - \mathbf{1}_{(m=y_1)}) \right).$$
$$\mathbf{p}_{n,m}^{\mathrm{T}} = \mathbf{p}_{n-1,m}^{\mathrm{T}} \left(\tilde{\mathbf{P}}_{n-1} (\mathbf{P}_{n-1})^{\frac{t_n-t_{n-1}}{\Delta}-1} \mathbf{1}_{(m=y_{n-1})} + \mathbf{I}_{n-1,n} (1 - \mathbf{1}_{(m=y_{n-1})}) \right), n \geqslant 2.$$

Задавшись некоторым[36] значением Δ и зафиксировав управление y_1, алгоритм[37] выбора решения для задания, поступившего в момент t_{n+1}, $n \geqslant 1$, состоит в следующем[38]: сначала рассчитываются распределения $\mathbf{p}_{n+1,m}^{\mathrm{T}}$, $1 \leqslant m \leqslant M$,

[35] Напомним, что $\mathbf{1}_{(A)}$ — индикатор множества A.

[36] См. сноску на стр. 137.

[37] Описанный алгоритм, по-видимому, можно распространить и на случай, когда размеры заданий имеют распределение фазового типа. Такое обобщение было бы полезным (при не сверхбольшой дисперсии размера заданий), поскольку, как известно, распределением фазового типа с любой степенью точности можно приблизить (в смысле слабой сходимости ф. р.) любое распределение (см. [416, С. 181]).

[38] Отметим, что размерность векторов $\mathbf{p}_{n,m}^{\mathrm{T}}$ растет вместе с ростом n. Поэтому вычисления при реализации необходимо (на каком-то этапе) обрывать.

затем вычисляются по определению значения $\mathsf{E}N_{n+1}^{(m)}$ и, наконец, выбирается сервер по правилу (3.14). Неизвестное в (3.14) значение θ либо подбирается на имитируемых траекториях, либо полагается равным единице.

Оценка параметров. Основная цель введения постоянного коэффициента θ в алгоритмы диспетчеризации по предыстории (см. (3.13) и (3.14)) — компенсация тех изменений, которые вызываются в исходной задаче значениями всех остальных параметров алгоритмов[39]. Однако, безотносительно предназначения коэффициента θ, само его наличие[40] в структуре стратегий позволяет называть их пороговыми. Как уже упоминалось выше, оценки порога θ могут быть найдены экспериментально, т. е. на имитируемых траекториях путем ручного или умным образом огранизованного автоматизированного перебора[41]. Поскольку, судя по вычислительным экспериментам, в каждой задаче существует единственное оптимальное значение порога θ, то наличие хорошего (хотя бы в каком-то смысле) начального приближения заметно упрощает поиск. Например, за такое начальное значение можно взять оптимальное значение порога в какой-нибудь аналогичной задаче, но с полным наблюдением. Далее речь пойдет об одной из таких задач, решение которой может служить начальным приближением для значений порогов в алгоритмах диспетчеризации по полной предыстории. Составить некоторое представление о результативности этого подхода к оценке значений порогов позволяют результаты вычислительных экспериментов, речь о которых пойдет в параграфе 3.4 (см. стр. 181–182).

Итак, рассмотрим систему, состоящую из двух параллельно и независимо работающих серверов (далее — серверы 1 и 2), в которую поступает случайный поток одинаковых заданий единичного размера. В каждом сервере имеется очередь неограниченной емкости для хранения заданий, ожидающих обработки. Не ограничивая общности рассуждений, будем считать, что производительность[42] сервера 1 равна единице, а сервера 2 равна $v > 1$. Предполагается, что входной поток заданий является рекуррентным[43], и распределение $F(x) = \mathsf{P}\{\tau < x\}$ интервала τ между поступлениями имеет конечный первый

[39] Которые приходится искать методом проб и ошибок; см. сноску на стр. 137.

[40] Есть, конечно, и дополнительные основания. Например, (3.13) при $M = 2$ является не чем иным, как пороговой стратегией.

[41] См. сноску на стр. 142.

[42] Т. е. сервер 1 выполняет каждое задание за время 1, а сервер 2 — за время v^{-1}.

[43] Это предположение является существенным. Вопрос разработки подходящего для коррелированного потока решения является открытым.

момент $\int_0^\infty x dF(x) = \lambda^{-1}$. Постановка задания в одну из очередей осуществляется в момент его поступления и дальнейшие переходы между очередями не допускаются. Обслуживание заданий в серверах происходит в порядке их поступления, без прерывания. Обозначим через x текущую нагрузку в сервере 1, которая складывается из числа заданий в очереди и остаточной нагрузки непосредственно на процессоре. Аналогичную величину для сервера 2 обозначим через y. Состоянием системы назовем пару $s = (x,y)$, $s \in S = [0,\infty) \times [0,\infty)$.

Пусть в некоторый момент, когда поступает задание, система находится в состоянии s. В этот момент принимается решение относительно того, в какую очередь направляется задание. Если задание отправлено в очередь к серверу 1 (будем это характеризовать словами "выбрано действие 1"), то в момент прихода следующего задания через случайное время τ система, очевидно, окажется в состоянии

$$s' = ((x + 1 - \tau)^+, (y - v\tau)^+),$$

где для сокращения записи используется стандартное обозначение $a^+ = \max(0,a)$. Если же задание отправлено в очередь к серверу 2 (действие 2), то в момент прихода следующего задания система окажется в состоянии

$$s' = ((x - \tau)^+, (y + 1 - v\tau)^+).$$

Вероятность перехода в первом из этих случаев, обозначим ее $P_1(s'|s)$, представляет собой распределение на множестве

$$A_1(s) = \{(x',y'), x' = (x + 1 - t)^+, y' = (y - vt)^+, t \geqslant 0\}.$$

Аналогично, вероятность перехода вовтором случае, обозначим ее $P_2(s'|s)$, представляет собой распределение на множестве

$$A_2(s) = \{(x',y'), x' = (x - t)^+, y' = (y + 1 - vt)^+, t \geqslant 0\}.$$

Обе вероятности, $P_1(s'|s)$ и $P_2(s'|s)$, определяются распределением $F(x)$ случайной величины τ. Также заметим, что оба множества, $A_1(s)$ и $A_2(s)$, параметризованы одной неотрицательной вещественной переменной, то есть являются одномерными. Каждое из них составлено в общем случае из двух отрезков на плоскости, один из которых представляет собой часть прямой с угловым коэффициентом v — от точки s до первого пересечения с одной из

осей Ox или Oy, а другой — от указанной точки пересечения до начала координат.

Рассмотрим теперь систему в моменты поступления заданий, которые перенумеруем последовательно: $n = 1,2,\ldots$ Обозначим через s_n состояние системы в момент n (до принятия решения) и пусть s_0 — известное начальное состояние системы. Полагаем, что система управляется согласно пороговой стратегии с порогом $\xi \geqslant 0$. Это означает, что в состоянии s_n выбирается действие 1, если

$$s_n \in S^{\xi} = \{(x,y) : \frac{y}{v} - x > \xi\} \tag{3.15}$$

и выбирается действие 2, если

$$s_n \in \overline{S}^{\xi} = S \backslash S^{\xi} = \{(x,y) : \frac{y}{v} - x \leqslant \xi\}. \tag{3.16}$$

В этой ситуации, последовательность s_n образует марковскую цепь с переходной вероятностью

$$P_{\xi}(s'|s) = \begin{cases} P_1(s'|s), & \text{если } s \in S^{\xi}, \\ P_2(s'|s), & \text{если } s \in \overline{S}^{\xi}. \end{cases}$$

Для полноты описания необходимо еще задать начальное распределение; например, можно положить $s_1 = (0,0)$.

Предельное распределение[44], соответствующее пороговой стратегии с порогом ξ, обозначим через π_{ξ}. С каждым состоянием марковской цепи s_n связан "доход" (далее — g_n), который интерпретируется как время пребывания в системе n-го по счету задания. Если $s_n = s = (x,y)$, то, поскольку дисциплина очереди в каждом сервере — FIFO, имеем

$$g_n(s_n) = \begin{cases} x + 1, & \text{если } s_n \in S^{\xi}, \\ \frac{y+1}{v}, & \text{если } s_n \in \overline{S}^{\xi}. \end{cases}$$

Предельный средний доход (предельное среднее время пребывания задания в системе) определяется как

$$T_{\xi} = \int_S g(u)\pi_{\xi}(du).$$

[44]Предполагая, что оно существует. Для этого достаточно потребовать, чтобы загрузка системы $\lambda/(1+v)$ была меньше единицы, поскольку цель введения пороговой стратегии — минимизация среднего времени пребывания задания в системе.

Задача заключается в нахождении значения ξ, которое минимизирует T_ξ.

Построение итеративного алгоритма для нахождения приближенного значения оптимального порога ξ_{opt} основано на следующем рассуждении. Пусть система находится в состоянии $\hat{s} = (\hat{x},\hat{y})$ таком, что $\frac{\hat{y}}{v} - \hat{x} = \xi_{\text{opt}}$. Тогда описанная выше пороговая стратегия предписывает выбирать в этом состоянии действие 1. Но на самом деле выбор действия в такой точке не имеет значения. Если бы это было не так, то изменением порога можно было бы добиться улучшения целевой функции[45]. Сравним две стратегии, $\sigma_\xi^{(1)}$ и $\sigma_\xi^{(2)}$, отличающиеся правилом выбора начального действия: первая стратегия выбирает действие 1, а вторая — действие 2. В дальнейшем обе стратегии действуют одинаково и так, как предписывает пороговая стратегия с порогом ξ. Индексируя распределения на шаге n, соответствующие этим стратегиям, единицей и двойкой, рассмотрим величину

$$\Delta_\xi = g^{(1)} - g^{(2)} + \sum_{n=1}^{\infty}\left(\int_S g(u)\pi_n^{(1)}(du) - \int_S g(u)\pi_n^{(2)}(du)\right), \tag{3.17}$$

где $g^{(1)}$ и $g^{(2)}$ — доходы в начальный момент, когда оба распределения, $\pi_0^{(1)}$ и $\pi_0^{(2)}$, сосредоточены в точке $\hat{s} = (\hat{x},\hat{y})$. По определению сравниваемых стратегий

$$g^{(1)} = \hat{x} + 1, \quad g^{(2)} = \frac{\hat{y}+1}{v}. \tag{3.18}$$

поэтому $g^{(1)} - g^{(2)} \neq 0$. Последующие слагаемые также отличны от нуля, поскольку распределения $\pi_n^{(1)}$ и $\pi_n^{(2)}$ не совпадают при любом n. Однако в силу того, что $\lim_{n\to\infty}\pi_n^{(1)} = \lim_{n\to\infty}\pi_n^{(2)} = \pi_\xi$, ряд в (3.17) сходится[46]. Если $\Delta_\xi = 0$, то значение порога ξ следует считать оптимальным. Если же $\Delta_\xi \neq 0$, то значение следует увеличить или уменьшить в зависимости от знака Δ_ξ.

Для реализации этой идеи требуется уметь находить, хотя бы приближенно, распределения $\pi_n^{(1)}$ и $\pi_n^{(2)}$. Очевидный подход заключается в аппроксимации марковской цепи s_n некоторой конечной цепью с переходной матрицей $\mathbf{P}$ и использованием соотношения

$$\pi_n = \pi_{n-1}\mathbf{P},$$

где π_n — вектор–строка вероятностей состояний из конечного множества. Однако прямое применение этого подхода с прямоугольной равномерной сеткой,

[45] Это утверждение следует из интуитивно очевидной непрерывности функции T_ξ.

[46] Причем скорость сходимости — экспоненциальная (см., например, [507], [508, С. 277]).

разбивающей пространство состояний, дает плохие результаты. Дело в том, что, как показывают эксперименты, функция T_ξ имеет очень пологий график вблизи минимума. Поэтому для получения приемлемой точности необходимо образовывать матрицу $\mathbf{P}$ слишком большой размерности — алгоритм становится неконструктивным. Предлагаемый далее метод использует специальное неравномерное разбиение[47] пространства состояний и специальный способ вычисления вероятностей перехода, не требующий использования матрицы $\mathbf{P}$.

Опишем конечную марковская цепь $\hat{s}_n$, которая аппроксимирует марковскую цепь s_n. Вначале проведем дискретизацию множества состояний S. Воспользуемся числовой последовательностью

$$h_i = h_0(1+\alpha)^i,\ i = 0,1,\ldots, \tag{3.19}$$

где $\alpha > 0$ — некоторая константа (причем $L \ll 1$). Зададим множества (прямые линии)

$$\mathcal{B}_i = \{(x,y) : y = v(x - a_i)\},\ i = 0, \pm1, \pm2, \ldots, L,$$

где L — некоторое натуральное число (причем $\alpha \gg 1$) и

$$a_i = \begin{cases} 0, & \text{если } i = 0, \\ a_{i-1} + h_{i-1}, & \text{если } i > 0, \\ a_{i+1} - h_{-i-1}, & \text{если } i < 0. \end{cases}$$

Кроме того, рассмотрим множества $\mathcal{C}_j^+$ и $\mathcal{C}_j^-$, задаваемые следующим образом

$$\mathcal{C}_j^+ = \{(x,y) : x = a_j\},\ j = 1,2,\ldots,L,$$
$$\mathcal{C}_j^- = \{(x,y) : y = va_j\},\ j = 1,2,\ldots,L.$$

Определим совокупность точек $\tilde{S}^{\alpha,L}$:

$$\tilde{S}^{\alpha,L} = \{\tilde{S}_{00}\} \cup \{\tilde{S}_{ij}, i = 0, \pm1, \pm2, \ldots, L; j = 1,2,\ldots,L\},$$

таким образом, что $\tilde{S}_{00} = (0,0)$, $\tilde{S}_{ij} = \mathcal{B}_i \cap \mathcal{C}_j^+$ для $i \geqslant 0$ и $\tilde{S}_{ij} = \mathcal{B}_i \cap \mathcal{C}_j^-$ для $i < 0$.

[47] Как показывают вычислительные эксперименты такая косоугольная сетка достаточна для получения решений за приемлемое вычислительное время. Для полноты картины, однако, необходимо отметить, что к настоящему времени, ввиду важности для решения задач практики численными методами, предложено немало типов разбиений и способов их построения. Работы в этом направлении продолжаются (укажем для примера на [509–513]; см. также [514, § 2], [515]). Вопрос выбора наилучшей сетки из известных может быть предметом отдельных исследований.

Множество $\tilde{S}^{\alpha,L}$ содержит $(L+1)^2$ точек и представляет собой сетку, которая покрывает на плоскости прямоугольник со сторонами H (по оси Ox) и vH (по оси Oy) и с точкой (0,0) в качестве одной из вершин. При этом

$$H = \sum_{i=0}^{L-1} h_i = \frac{h_0(1+\alpha)^L - h_0}{\alpha}. \tag{3.20}$$

Точки из множества $\tilde{S}^{\alpha,L}$ распределены неравномерно (рисунок 12). Большее сгущение имеет место при приближении к началу координат и к прямой B_0 т. е. прямой $y = vx$. Более редко точки располагаются при удалении от начала координат и от прямой B_0.

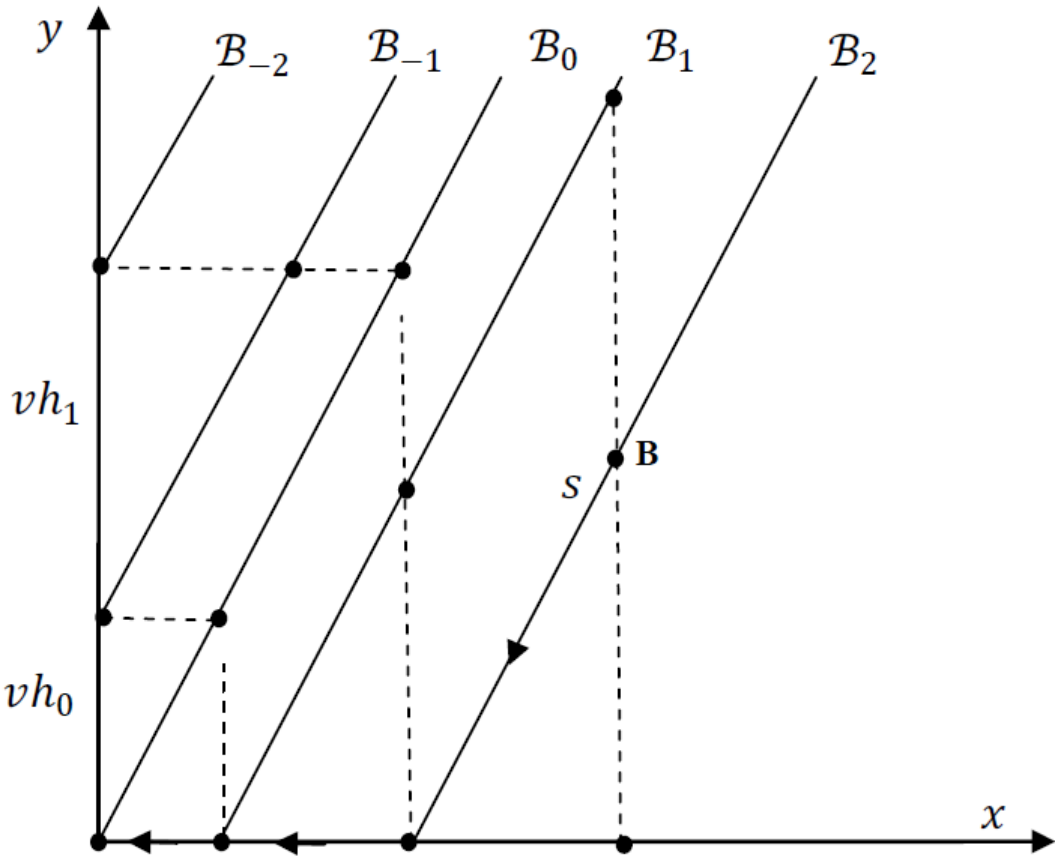


Рисунок 12 — Схема дискретизации множества состояний S

Точки из множества[48] $\tilde{S}^{\alpha,L}$ являются основой для построения множества состояний аппроксимирующей цепи s_n. Следующие рассуждения вытекают из описания множеств $A_1(s)$ и $A_2(s)$, сделанного выше при рассмотрении переходов

[48]Как следует из построения сетки $\tilde{S}^{\alpha,L}$, длина любого звена (горизонтального, вертикального или наклонного) для произвольного маршрута определяется с помощью некоторого значения h_i. Легко понять, что (абсолютное) время, необходимое для того, чтобы система преодолела расстояние вдоль любого звена, равняется h_i.

в цепи s_n. Пусть цепь s_n оказалась после принятия решения на шаге n в некоторой точке $s = (x,y) \in \mathcal{B}_i$ (рисунок 12). Тогда изменение состояния системы до момента прихода следующего задания происходит по "маршруту", отмеченному на рисунок 12 стрелками. Этот маршрут (на рисунке $i = 2$) начинается в точке s и оканчивается в точке (0,0), соответствующей пустой системе. К моменту $(n+1)$-го перехода цепи в новое состояние, система может оказаться в любой точке этого маршрута. Если множество состояний стало дискретным, то, естественно, все точки маршрута должны принадлежать дискретному множеству, но характер "движения", приводящего к новому состоянию цепи, остается прежним: вначале система движется по наклонной прямой, а затем по той оси, которую она достигнет раньше. Определим следующие "множества маршрутов":

$$\begin{aligned}
&\mathcal{A}_0 = \mathcal{B}_0 \cap \tilde{S}^{\alpha,L},\\
&\mathcal{A}_i = \big(\mathcal{B}_i \cap \tilde{S}^{\alpha,L}\big) \cup \{\tilde{S}_{i-1,i-1}, \ldots, \tilde{S}_{1,1}, \tilde{S}_{0,0}\},\ i > 0,\\
&\mathcal{A}_i = \big(\mathcal{B}_i \cap \tilde{S}^{\alpha,L}\big) \cup \{\tilde{S}_{i+1,-i-1}, \ldots, \tilde{S}_{-1,1}, \tilde{S}_{0,0}\},\ i < 0.
\end{aligned}$$

В этих формулах выражения в круглых скобках содержат точки из множества $\tilde{S}^{\alpha,L}$, лежащие на прямой B_i. В фигурных скобках содержатся те точки из $\tilde{S}^{\alpha,L}$, которые "соединяют" точку первого пересечения прямой B_i оси Ox или Oy с началом координат. Более развернутая запись множеств B_i выглядит так:

$$\begin{aligned}
&\mathcal{A}_0 = \{\tilde{S}_{0,L}, \tilde{S}_{0,L-1}, \ldots, \tilde{S}_{0,0}\},\\
&\mathcal{A}_i = \{\tilde{S}_{i,L}, \tilde{S}_{i,L-1}, \ldots, \tilde{S}_{i,i}, \tilde{S}_{i-1,i-1}, \ldots, \tilde{S}_{1,1}, \tilde{S}_{0,0}\},\ i > 0,\\
&\mathcal{A}_i = \{\tilde{S}_{i,L}, \tilde{S}_{i,L-1}, \ldots, \tilde{S}_{i,-i}, \tilde{S}_{i+1,-i-1}, \ldots, \tilde{S}_{-1,1}, \tilde{S}_{0,0}\},\ i < 0.
\end{aligned}$$

Любой дискретный маршрут, подобный изображенному на рисунке 12, является подмножеством A_i. Элементы множества A_i можно естественным образом перенумеровать, например, начиная с точки $\tilde{S}_{0,0}$:

$$\begin{aligned}
&S_{i,0} = \tilde{S}_{0,0}, S_{i,1} = \tilde{S}_{1,1}, \ldots, S_{i,i-1} = \tilde{S}_{i-1,i-1},\\
&S_{i,i} = \tilde{S}_{i,i}, \ldots S_{i,L-1} = \tilde{S}_{i,L-1}, S_{i,L} = \tilde{S}_{i,L},
\end{aligned}$$

если $i \geqslant 0$ и

$$\begin{aligned}
&S_{i,0} = \tilde{S}_{0,0}, S_{i,1} = \tilde{S}_{-1,1}, \ldots, S_{i,-i-1} = \tilde{S}_{i+1,-i-1},\\
&S_{i,-i} = \tilde{S}_{i,-i}, \ldots S_{i,L-1} = \tilde{S}_{i,L-1}, S_{i,L} = \tilde{S}_{i,L}.
\end{aligned}$$

если $i < 0$. Таким образом, для любого i маршрут $\mathcal{A}_i = \{S_{i,0}, S_{i,1}, \ldots, S_{i,L}\}$. В качестве множества состояний конечной марковской цепи $\hat{s}_n$ примем объединение множеств маршрутов

$$S^{\alpha,L} = \bigcup_{i=-L}^{L} \mathcal{A}_i.$$

Множество $S^{\alpha,L}$ содержит $(L+1) \times (2L+1)$ элементов, то есть больше элементов, чем множество $\tilde{S}^{\alpha,L}$. Это объясняется тем, что некоторые точки сетки $\tilde{S}^{\alpha,L}$ входят сразу в несколько множеств A_i. Таковыми являются все, кроме крайних, точки, лежащие на координатных осях. Данное обстоятельство интерпретируется как введение фиктивных состояний — дублеров одного физического состояния. Например, в каждом множестве A_i присутствует точка $S_{i,0} = \tilde{S}_{0,0}$, характеризующая пустую систему. Подобная конструкция множества состояний упрощает описание переходных вероятностей и работу с ними.

Опишем теперь переходы в цепи $\hat{s}_n$. Пусть $\hat{s}_n = S_{ij} \in S^{\alpha,L}$, и пусть $S_{kl} \in S^{\alpha,L}$ — состояние, в которое перешла система под воздействием выбранного в момент n управления. Переход $S_{ij} \to S_{kl}$ происходит мгновенно. Точка S_{kl} определяется механизмом пороговой стратегии, при этом индексы k и l устанавливаются детерминировано по значениям i и j. Далее система будет изменять свое состояние, пока не поступит новое задание. В этот момент цепь переходит в новое состояние $\hat{s}_{n+1}$, которым может быть любая из точек S_{kl}, $S_{k,l-1}$, $\ldots$, S_{k0}. В соответствии с построением множества $S^{\alpha,L}$ (и замечанием в сноске) вероятности попадания в состояния $S_{kl} \to S_{km}$, $m = 0,1,\ldots,l$, зависят только от индекса l, но не от индекса k. Обозначим эти вероятности через q_{lm}, т. е. $q_{lm} = \mathsf{P}\{S_{kl} \to S_{km}\}$. Выпишем формулы для их вычисления. Очевидно, $q_{00} = 1$. Рассмотрим случай $l = 1$. Физически этот случай соответствует тому, что в некоторый момент в системе есть нагрузка, требующая для своего выполнения времени h_0. Вследствие дискретизации множества состояний системы имеются только две возможности: либо в момент прихода очередного задания нагрузка в системе останется прежней, либо система будет пуста. Вероятности соответствующих событий составляют q_{11} и q_{10}. В качестве первой из этих вероятностей естественно принять вероятность того, что время до поступления очередного задания не будет превышать $0.5h_0$ т. е.

$$q_{11} = F(0.5h_0), \quad q_{10} = 1 - q_{11}.$$

Аналогичное рассуждение приводит к следующим формулам для произвольного значения l, $1 \leqslant l \leqslant L$:

$$q_{lm} = F(H_{l-m+1}) - F(H_{l-m}),\ 0 \leqslant m \leqslant l,$$

где

$$H_m = \begin{cases} 0, & \text{если } m = 0, \\ h_{l-1} + h_{l-2} + \cdots + h_{l-m+1} + 0.5h_{m-1}, & \text{если } 1 \leqslant m \leqslant l, \\ 1, & \text{если } m = l+1. \end{cases}$$

При дальнейшей работе с конечной марковской цепью $\hat{s}_n$ необходимо переходить от "косоугольных" координат, выражаемых индексами элементов S_{ij}, к прямоугольным координатам и обратно. Приведем эти преобразования, которые получаются путем несложных выкладок из правил построения множества $S^{\alpha,L}$. Пусть (x,y) — прямоугольные координаты точки $S_{ij} \in S^{\alpha,L}$. Тогда $x = y = 0$ при $i = 0$. Если же $i > 0$, то

$$x = h_0 \frac{(1+\alpha)^j - 1}{\alpha}, \tag{3.21}$$

$$y = \max\left(0, vx - vh_0 \frac{(1+\alpha)^i - 1}{\alpha}\right), \tag{3.22}$$

иначе, т. е. при $i < 0$, имеем

$$x = \max\left(0, \frac{y}{v} - h_0 \frac{(1+\alpha)^{-i} - 1}{\alpha}\right), \tag{3.23}$$

$$y = vh_0 \frac{(1+\alpha)^j - 1}{\alpha}. \tag{3.24}$$

Обратное преобразование, вообще говоря, неоднозначно, во-первых, из-за наличия фиктивных состояний и, во-вторых, из-за того, что можно по-разному выбирать из множества $S_{ij} \in S^{\alpha,L}$, аппроксимирующую произвольную точку $(x,y) \in S$. Можно принять, например, следующий вариант:

$$i = \max\bigl(-L, \min(L, i')\bigr), \tag{3.25}$$

$$j = \max\bigl(-L, \min(L, j')\bigr), \tag{3.26}$$

где

$$i' = \operatorname{sign}\left(x - \frac{y}{v}\right)\left[\frac{\ln\left(1 + \frac{\alpha}{h_0}\left|x - \frac{y}{v}\right|\right)}{\ln(1+\alpha)}\right],$$

$$j' = \begin{cases} \left[\frac{\ln\left(1+\frac{\alpha x}{h_0}\right)}{\ln(1+\alpha)}\right], & \text{если } y < vx, \\ \left[\frac{\ln\left(1+\frac{\alpha y}{h_0 v}\right)}{\ln(1+\alpha)}\right], & \text{если } y \geqslant vx. \end{cases}$$

Здесь, как обычно, $\operatorname{sign}(a)$ означает знак числа a, $[a]$ означает целую часть числа a.

Предположим, что в момент n исходная марковская цепь (т. е. s_n) находится в состоянии (x,y). Тогда после принятия решения относительно поступившего в этот момент задания, цепь перейдет в состояние

$$(\tilde{x}, \tilde{y}) = \begin{cases} (x+1,y), & \text{если } \frac{y}{v} - x > \xi, \\ (x,y+1), & \text{если } \frac{y}{v} - x \leqslant \xi. \end{cases} \tag{3.27}$$

Выберем в качестве начального состояния для аппроксимирующей цепь $\hat{s}_n$ состояние $\hat{s} = (\hat{x}, \hat{y})$ такое, что $\frac{\hat{y}}{v} - \hat{x} = \xi$. Рассмотрим опять упомянутые выше стратегии $\sigma_\xi^{(1)}$ и $\sigma_\xi^{(2)}$. Обозначим через $\hat{\pi}_n^{(1)}$ и $\hat{\pi}_n^{(2)}$ соответствующие этим стратегиям распределения на множестве $S^{\alpha,L}$. Дискретный аналог величины Δ_ξ, определенной формулой (3.17), имеет вид

$$\Delta_\xi^{\alpha,L} = g^{(1)} - g^{(2)} + \sum_{n=1}^{\infty}\sum_i\sum_j g(i,j)\left(\hat{\pi}_n^{(1)}(i,j) - \hat{\pi}_n^{(2)}(i,j)\right),$$

где $g^{(1)}$ и $g^{(2)}$ вычисляются согласно (3.18), $g(i,j) = g(S_{ij})$, и $\hat{\pi}_n^{(i)}(i,j)$ — значения распределений $\hat{\pi}_n^{(1)}$ и $\hat{\pi}_n^{(2)}$ в точке S_{ij}. Будем обозначать через (x_{ij},y_{ij}) прямоугольные координаты точки S_{ij}, вычисленные по формулам (3.21)–(3.24), и через I_{xy}, J_{xy} — индексы обратного преобразования, вычисленные по формулам (3.25) и (3.26). Обозначения $\tilde{x} = \tilde{x}(x,y)$, $\tilde{y} = \tilde{y}(x,y)$ выражают преобразования по формулам (3.27).

Для приближенного нахождения $\Delta_\xi^{\alpha,L}$, т. е. оценки произвольного значения порога ξ с точки зрения его оптимальности, может быть использован следующий алгоритм[49].

[49] К которому дадим следующие пояснения. В строке 8 рассчитываются вероятности состояний после выбора действия 1 на первом шаге; в строке 11 — вероятности состояний после выбора действия 2

Алгоритм III. Псевдокод алгоритма оценки произвольного значения порога ξ

Шаг 1

1: $\Delta_0 = g^{(1)} - g^{(2)}$
2: **if** $i = i_{\hat{x},\hat{y}}$, $j = j_{\hat{x},\hat{y}}$ **then**
3: $\quad \hat{\pi}_0^{(1)}(i,j) = 1$, $\hat{\pi}_0^{(2)}(i,j) = 0$
4: **else**
5: $\quad \hat{\pi}_0^{(1)}(i,j) = 0$, $\hat{\pi}_0^{(2)}(i,j) = 1$

Шаг 2

6: $x = \hat{x} + 1$, $y = \hat{y}$, $k = I_{xy}$, $l = J_{xy}$, $\hat{\pi}_1^{(1)} = 0$
7: **for** $m = 0$ to l **do**
8: $\quad \hat{\pi}_1^{(1)}(k,m) = \hat{\pi}_1^{(1)}(k,m) + q_{lm} \cdot \hat{\pi}_0^{(1)}(k,l)$
9: $x = \hat{x}$, $y = \hat{y} + 1$, $k = I_{xy}$, $l = J_{xy}$, $\hat{\pi}_1^{(2)} = 0$
10: **for** $m = 0$ to l **do**
11: $\quad \hat{\pi}_1^{(2)}(k,m) = \hat{\pi}_1^{(2)}(k,m) + q_{lm} \cdot \hat{\pi}_0^{(2)}(k,l)$
12: $\Delta\hat{\pi}_1 = \hat{\pi}_1^{(1)} - \hat{\pi}_1^{(2)}$
13: $\Delta_1 = \Delta_0 + \sum_{i=-L}^{L} \sum_{j=0}^{L} g(i,j) \cdot \Delta\hat{\pi}_1(i,j)$
14: $n = 1$

Шаг 3

15: $n = n + 1$
16: **for** $i = -L$ to L **do**
17: $\quad$ **for** $j = 0$ to L **do**
18: $\quad\quad x = x_{ij}$, $y = y_{ij}$
19: $\quad\quad k = i_{\tilde{x}\tilde{y}}$, $l = j_{\tilde{x}\tilde{y}}$
20: $\quad\quad \Delta\hat{\pi}_n = 0$
21: $\quad\quad$ **for** $m = 0$ to l **do**
22: $\quad\quad\quad \Delta\hat{\pi}_n = \Delta\hat{\pi}_n + q_{lm} \cdot \hat{\pi}_{n-1}^{(1)}(k,l) - q_{lm} \cdot \hat{\pi}_{n-1}^{(2)}(k,l)$
23: $\Delta_n = \Delta_{n-1} + \sum_{i=-L}^{L} \sum_{j=0}^{L} g(i,j) \cdot \Delta\hat{\pi}_n(i,j)$
24: **if** $|\Delta_n - \Delta_{n-1}| < \varepsilon$ **then**
25: $\quad$ goto *Шаг 3*
26: **else**
27: $\quad$ **return** $\Delta_n = \Delta_{\xi}^{\alpha,L}$.

Предложенный алгоритм представляет собой версию "по-умолчанию", которая допускает модификации в расчете на увеличение эффективности. Например, можно перенести "область сгущения" сетки $\tilde{S}^{\alpha,L}$, которая расположена в прилегающей к нулю окрестности прямой $y = vx$, в район наиболее часто повторяющихся состояний. Центр сгущения можно определять заранее с помощью статистического моделирования. В алгоритме также не предусмотрена

на первом шаге. В строке 12 разность — покомпонентная. В строке 18 задаются прямоугольные координаты точки S_{ij}; в строке 19 — координаты системы на сетке после принятия решения. В строке 24, ε — наперед заданное положительное число ($\varepsilon \ll 1$).

процедура вычисления порога, поскольку она может быть реализована разными способами. Например, задавшись гарантированными границами интервала, внутри которого лежит оптимальное значение порога (скажем, $[0,v^{-1}]$), можно далее очевидным образом действовать методом деления отрезка пополам.

Проиллюстрируем работу *Алгоритма III*. Найдем с его помощью значения порогов в двух системах, отличающихся типом входящего потока. Пусть в каждой из систем быстрый сервер имеет скорость $v = 2$, но в одну поступает пуассоновский поток с параметром λ, а в другую — рекурренетный поток с Парето распределенным интервалом между поступлениями т. е. $F(x) = 1 - b^a x^{-a}$, $x \geqslant b$. Выберем параметры потоков таким образом, чтобы загрузка каждой системы была равна 0.8 и среднее время между поступлениями заданий было одинаковым; т. о.

$$\lambda = 2.4, \quad b = 0.21, \quad a = \frac{1}{1 - \lambda b} \approx 2.016.$$

Отметим, что дисперсия времени между поступлениями для одного потока равна $\lambda^{-2} \approx 0.417$, а для другого — $ab^2/((a-1)^2(a-2)) \approx 5.833$. Параметры сетки $\tilde{S}^{\alpha,L}$ выберем опосредованно, задав размеры наименьшей ($h_0 = 0.005$) и наибольшей ($h_{L-1} = 0.025$) ячейки, а также положив размер области покрытия $H = 10$. При этих значениях размеров параметры сетки равны

$$\alpha = \frac{h_{L-1} - h_0}{H - h_{L-1}} \approx 0.0001, \quad L = \left[\frac{\ln\left(1 + \frac{\alpha H}{h_0}\right)}{\ln(1+\alpha)} \right] \approx 1600.$$

Таким образом, число состояний в аппроксимирующей цепи равно[50] $(L+1) \times (2L+1) \approx 5.2 \times 10^6$. Предложенный алгоритм дает следующие диапазоны для оптимального значения порога:

система с пуассоновским потоком: $\xi \in (0.166, 0.167)$.

система с “Парето потоком”: $\xi \in (0.150, 0.151)$.

[50] Численные эксперименты показывают, что решение задачи может вести себя нерегулярно при уменьшении ячеек сетки и, в итоге, может “сломаться”. По примеру из теории дифференциальных уравнений в частных производных, в которой известен критерий Куранта–Фридрихса–Леви (необходимое условие устойчивости численной схемы), здесь никакого правила (например, условий на длины сторон ячеек сетки по направлениям) установить не удалось. Подходящий размер сетки приходится искать методом проб и ошибок.

Чтобы понять, насколько точны эти результаты, рассмотрим полученные с помощью имитации (метода Монте-Карло) значения целевой функции в окрестности найденных оптимальных значений. Эти значения представлены в таблицах 8 и 9.

Таблица 8 — Значения целевой функции T_ξ в окрестности оптимального значения ξ для системы с пуассоновским входящим потоком при загрузке 0.8

Значение порога, ξ	Среднее время пребывания, T_ξ
0.160	1.25459
0.162	1.25458
0.164	1.25456
0.166	1.25454
0.168	1.25454
0.172	1.25454
0.174	1.25455

Таблица 9 — Значения целевой функции T_ξ в окрестности оптимального значения ξ для системы с входящим Парето–потоком при загрузке 0.8

Значение порога, ξ	Среднее время пребывания, T_ξ
0.144	0.93638
0.146	0.93637
0.148	0.93636
0.150	0.93636
0.152	0.93636
0.154	0.93637
0.156	0.93638

Как видно из полученных значений, итеративный расчет порога дает вполне удовлетворительную точность до третьего знака после запятой. Проверить правильность значений в четвертом знаке путем статистического моделирования весьма затруднительно, поскольку потребовалось бы оценивать целевую функцию с точностью до шестого знака после запятой, в то время как одна оценка этой функции до пятого знака после запятой занимает несколько часов работы современного стандартного персонального компьютера. В то же время одна оценка с помощью предложенного алгоритма занимает примерно 5–10 минут.

Независимую оценку точности можно получить по численными результатам из [272], где для решения рассматриваемой задачи использовался принципиально другой метод. Значения в таблице 8 относятся к кривой $\rho = 0.8$ на первом из трех рисунков в [272, Figure 6], который воспроизведен ниже (см. рисунок 13).

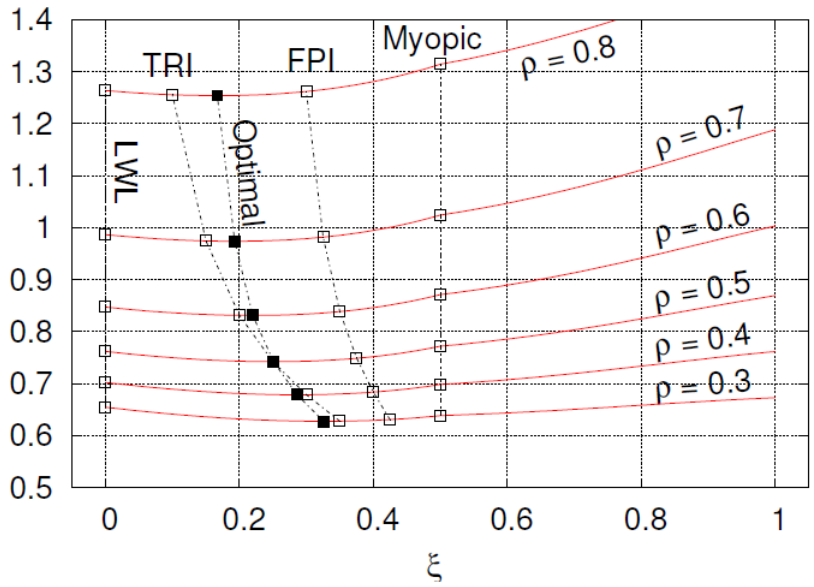


Рисунок 13 — Система из двух серверов (производительности 1 и 2) с пуассоновским потоком заданий единичного размера. Графики стационарного среднего времени пребывания в системе зависимости от значения порога ξ при различных значениях загрузки ρ. Оптимальные значения порога отмечены ■; значения при других стратегиях (TRI, FPI, Myopic) отмечены □. Рисунок взят из [272, Figure 6]

Хотя по рисунку и трудно делать какие-либо количественные выводы, визуально очевидно, что найденное решение является наилучшим[51].

Рассуждение[52], позволившее построить алгоритм для нахождения приближенного значения оптимального порога при $M = 2$, "проходит" и для произвольного[53] $2 \leqslant M < \infty$. Но разработать на его основе какую-либо численную процедуру не удается. Вместе с тем для общего случая представляется

[51] Примечательно (см. [272, Раздел 4.2]), что иногда близкие к оптимальным значения целевой функции получаются, если значение порога ξ вычислять по простому эвристическому правилу $\xi = (v_1^{-1} - v_2^{-1})(1 - \rho)$. На рисунке 13 значения по этому правилу обозначены TRI и отмечены □.

[52] См. стр. 164.

[53] Однако, для систем, состоящих из более, чем двух серверов структура оптимальной стратегии неизвестна: она совершенно необязательно должна быть пороговой (см., например, [191, Figure 4] и [283]).

довольно естественным следующий способ[54] поэтапного нахождения значений порогов $\xi^{(2)},\dots,\xi^{(M)}$, опирающийся на разработанный алгоритм для двух серверов. Предположим, что серверы занумерованы в порядке возрастания производительности т. е. $v^{(1)}<v^{(2)}<\dots<v^{(M)}$ и входной поток заданий является рекуррентным с распределением $F_1(x)$. На первом этапе с помощью предложенного алгоритма находится приближенно значение порога, $\xi^{(2)}$, для системы из двух серверов производительности $v^{(1)}$ и $(v^{(2)}+\dots+v^{(M)})$ соответственно. Затем, уже при известном $\xi^{(2)}$, решается задача нахождения[55] подходящего рекурретного потока $F_2(x)$ для потока заданий, отклоненных сервером 1. На втором этапе для системы из двух серверов производительности $v^{(2)}$ и $(v^{(3)}+\dots+v^{(M)})$ соответственно, с распределением входного потока $F_2(x)$, рассчитывается значение порога, $\xi^{(3)}$. После этого распределения потока заданий, отклоненных сервером 2, приближается некоторым рекуррентным потоком $F_3(x)$ и т. д. Таким образом, на каждом этапе осуществляется дискретизация множества состояний, строится аппроксимирующая цепь Маркова, применяется предложенная выше итерационная процедура нахождения порога и подбирается распределение потока оклоненных заданий. Решение о том, что в какую из M очередей направляется поступающее задание принимается так. Пусть в момент поступления система находится в состоянии $(x_1,x_2,\dots,x_M)$, где x_i — текущая незаконченная работа в сервере i. Тогда выбирается сервер 1, если $x_1\leqslant\sum_{i=2}^{M}x_i+\xi^{(2)}$. Иначе требуется дополнительная проверка: если $x_2\leqslant\sum_{i=3}^{M}x_i+\xi^{(3)}$, то выбирается сервер 2. Иначе требуется дополнительная проверка и т. д.

Закончим параграф двумя замечаниями относительно последней из рассмотренных задач. Для нахождения значений оптимальных порогов $\xi^{(1)},\dots,\xi^{(M)}$ в принципе применимы ныне популярные алгоритмы эволюционной оптимизации[56] [478], не требующие знания градиента целевой функции. Но поскольку в задаче нельзя получить выражения для самой целевой функции, такие алгоритмы приходится применять “внутри” имитационной модели. Поэтому качество получаемых решений является низким. Еще одним подходом для системы с произвольным числом серверов является подход, основанный

[54] Не гарантирующий, вообще говоря, нахождение значений именно оптимальных порогов.

[55] Например, можно воспользоваться методами из [516]; см. также [517].

[56] Например, particle swarm optimization, ant colony optimization, artificial swarm intelligence, gravitational search algorithm и др.

на использовании метода Монте–Карло в сочетании с адаптивными алгоритмами для управления частично наблюдаемыми марковскими цепями. Как показывают эксперименты, такой метод имеет подходящий “угол атаки” как для пороговой стратегии, так и для самого общего варианта постановки задачи[57]. С его помощью можно находить стратегию, которая являются лучшей по сравнению со всеми известными эвристическими стратегиями (см. [178] и [526]).

[57] С принципиальной точки зрения здесь достаточна и методология динамического программирования (см., например, уравнения (8.46) в [193]). Однако, как известно, объемы вычислений, требующиеся для получения приемлемых ответов, могут оказаться чрезмерно большими. Основанные на идеях марковского процесса принятия решений специальные методы вынуждают накладывать на изучаемые системы особые оганичения. Типичным является ограничение на входящие потоки (см., например, стр. 110 в [518]): полностью приходится отказываться от потоков, циркулирующих в реальных системах (см. [519; 520] и репозитории https://www.cs.huji.ac.il/labs/parallel/workload/, http://gwa.ewi.tudelft.nl/, http://ita.ee.lbl.gov/, http://mawi.wide.ad.jp/mawi/), а также от таких реальных ситуаций, когда размеры заданий зависят от длин интервалов между поступлениями [521]. Другим недостатком является необходимость внесения изменений в выкладки всякий раз, когда изменяются предположения о структуре системы (например, когда меняется число процесоров в серверах [518, Раздел 4], когда серверы могут выходить из строя и т.п.). Но, несмотря на сказанное, нельзя не отметить, что в ряде случаев (например, когда поступающие необязательно по пуассоновскому потоку задания “видят” временные средние [522; 523]) альтернативу таким методам найти нелегко: по совокупной эффективности они могут давать практически неулучшаемые результаты [178; 191; 518; 524; 525].

3.3 Аналитико–имитацинный подход. Общая схема построения алгоритмов управления при использовании в серверах консервативных дисциплин

В предыдущих параграфах был предложен аналитический подход к реализации идеи диспетчеризации по полной предыстории. На его основе для частично наблюдаемых систем с параллельным обслуживанием на однопроцессорных серверах с дисциплиной FIFO удалось разработать стратегию (см. *Алгоритм II*), являющуюся, как показывают вычислительные эксперименты, наилучшей[58] из всех ранее известных в научной литературе. Уже из самого названия подхода следует, что избрать его можно только в тех случаях, когда либо для значений $\mathsf{E}V_n$[59], либо для связанных с ними величин, зависяших, вообще говоря, от всей предыстории поведения системы до момента t_n, есть вычислительно реализуемые (точные или хорошие приближенные) формулы расчета. Однако, при отличном от FIFO обслуживании в серверах, получить их чаще всего не удается. В этом параграфе описывается более универсальный подход к реализации диспетчеризации по полной предыстории, не накладывающий ограничений на дисциплину обслуживания. Такое расширения области применения достигается путем замены точных значений величин, необходимых диспетчеру для выбора очередного действия, на их статистические оценки, полученные посредством имитационной модели.

В основе новых алгоритмов диспетчеризации по полной предыстории (в сравнении с теми, что предложены в предыдущих параграфах) лежит прием, используемый в теории адаптивного управления и известный под названием идентификационный подход. Суть последнего в следующем. За основу берется алгоритм, успешно действующий по отношению к конкретному объекту, если правильно выбрать параметры алгоритма. Однако выбор параметров зависит от свойств объекта, которые могут быть априори неизвестны. В этом случае иногда удается осуществить настройку параметров непосредственно в процессе управления, основываясь на идентификации по результатам наблюдений.

[58] По крайней мере, как с точки зрения минимума стационарного среднего времени ожидания, так и с точки зрения минимума стационарного среднего времени пребывания задания в системе.

[59] Напомним, что V_n — время, проведенное в системе заданием, поступившим в момент t_n.

В рассматриваемой ситуации, взяв за основу какой-то алгоритм[60], идентифицируются (с помощью метода статистических испытаний) необходимые для реализации, но недоступные для наблюдения, динамические характеристики серверов.

Одним из главных звеньев аналитико–имитационного подхода является компьютерная модель, имитирующая процесс поступления и обслуживания заданий в системе. Модель трактуется[61] как преобразование $\mathfrak{F}$ исходных данных $\mathbb{U}$, принимающих значения из некоторого пространства $\mathfrak{U}$, в выходные данные $\mathbb{V}$, возможные значения которых принадлежат пространству $\mathfrak{B}$ т. е. $\mathfrak{F} : \mathfrak{U} \to \mathfrak{B}$. Тройка $(\mathfrak{U},\mathfrak{B},\mathfrak{F})$ задает модель вместе с составом входных и выходных данных. В наиболее интересных случаях $\mathfrak{F}$ может быть задано только алгоритмически и, при этом, естественно, особое значение имеет точность, с которой $\mathfrak{F}$ воспроизводит процесс обслуживания в системе. Модель $(\mathfrak{U},\mathfrak{B},\mathfrak{F})$ является промежуточным объектом, на котором осуществляется оценка не поддающихся расчету величин, необходимых диспетчеру для выбора управления. Общая схема применения модели такова. Сначала, исходя из основного алгоритма, выбираются выходные данные $\mathbb{V}$ (например, поток значений незаконченной работы в каждом сервере). Затем фиксируются входные данные $\mathbb{U}$: для принятия решения y_n в качестве входных данных могут быть выбраны распределения $F(x)$ и $B(x)$ интервалов между поступлениями и размеров заданий каждого типа, предыстория совершенных действий до момента t_n и моменты их совершения. Наконец оценки значений $\mathbb{V}$ строятся по значениям $\mathfrak{F}(\mathbb{U})$.

В следующем параграфе на примере двух классов частично наблюдаемых систем с параллельным обслуживанием, в которых однопроцессорные серверы используют приципиально различные дисциплины обслуживания очереди, показано, каким образом с помощью предложенного аналитико–имитационного подхода строятся алгоритмы управления.

[60] Например, тот, что хорошо зарекомендовал себя в аналогичой, но полностью наблюдаемой системе: JSQ, HJSQ(d), LWL, Myopic, пороговый или какой-то другой из (огромного!) множества разработанных на сегодняший день алгоритмов (см., например, обзор в [174]). Однако можно поступить и по-другому: выбрать эвристическую стратегию, сконструированную из разумных соображений. Как будет показано далее на примере системы с дисциплиной PS, такой выбор может оказаться наилучшим.

[61] Здесь изложение следует [60].

3.4 Примеры и дополнения

Описание алгоритма при дисциплине FIFO

Рассмотрим уже встречавшуюся в параграфе 3.2 систему, в которой имеется всего два сервера (занумерованные числами 1 и 2) производительности v_1 и v_2, причем $v_1 < v_2$. Напомним, что распределение размера задания, поступившего в систему n-м по счету, есть $B_n(x)$, $0 \leqslant t_1 < \cdots < t_n < \ldots$ — последовательность моментов поступления заданий в систему, а $\Delta_n = t_{n+1} - t_n$ — промежутки между этими моментами. Решение (действие), принимаемое диспетчером в момент t_n относительно поступившего задания обозначается через y_n, $y_n \in \{1, 2\}$. Пусть $W_n^{(m)}$ — время, необходимое для выполнения всех заданий, имеющихся в сервере m в момент t_n, без учета задания, поступившего в этот момент. Следуя описанной выше схеме, выберем[62] за основу, вместо правила (3.2), алгоритм порогового типа. Пусть ξ — некоторая фиксированная неотрицательная величина. Правило диспетчеризации определим следующим образом: задание, поступившее в момент t_n направляется на сервер 1, если $\mathsf{E}W_n^{(1)} + \xi < \mathsf{E}W_n^{(2)}$, и на сервер 2 иначе, т. е.

$$y_n = \begin{cases} 1, & \text{если } \mathsf{E}W_n^{(1)} + \xi < \mathsf{E}W_n^{(2)}, \\ 2, & \text{если } \mathsf{E}W_n^{(1)} + \xi \geqslant \mathsf{E}W_n^{(2)}. \end{cases} \tag{3.28}$$

Чтобы воспользоваться этим правилом необходимо уметь находить оценки динамического состояния серверов, а именно средней незаконченной работы в каждом сервере. Однако вместо того, чтобы вычислять $\mathsf{E}W_n^{(m)}$ будем использовать оценки, полученные по имитируемым на основе наблюденной предыстории траекториям. Обозначим оценку $\mathsf{E}W_n^{(m)}$ к моменту t_n принятия очередного решения через $\hat{W}_n^{(m)}$ и, для определенности, будем считать, что в начальный момент система полностью свободна от заданий т. е. $\hat{W}_1^{(m)} = 0$. Для $n = 2$ наблюдаемая предыстория — это пара $(y_1, \Delta_1 = t_2 - t_1) = h_1$. Оценку $\hat{W}_2^{(m)}$ определим как $\mathsf{E}_{h_1} W_n^{(m)}$, где E_{h_1} — условное математическое ожидание при условии, что предыстория к моменту t_2 была h_1. Продолжая

[62] Такой выбор оправдывается здесь тем, что в полностью наблюдаемых системах с двумя серверами пороговая стратегия бывает оптимальной.

аналогичным образом, для $n = 2,\ldots,k$, определим оценки $\hat{W}_n^{(m)} = \mathsf{E}_{h_n} W_n^{(m)}$, где $h_n = (y_1,\Delta_1,\ldots,y_{n-1},\Delta_{n-1})$. Фиксированное натуральное число k будем называть глубиной памяти. Начиная с номера $n = k+1$ будем строить оценки, исходя из "усеченной" предыстории $h_{n,k} = (y_{n-k},\Delta_{n-k},\ldots,y_{n-1},\Delta_{n-1})$. Полагаем $\hat{W}_n^{(m)} = \mathsf{E}_{h_{n,k}} W_n^{(m)}$, где $\mathsf{E}_{h_{n,k}}$ — условное математическое ожидание при условии, что наблюдаемая часть предыстории на предшествующих k интервалах была $h_{n,k}$ и остаточные времена к моменту t_{n-k} равнялись $W_{n-k}^{(m)} = \hat{W}_{n-k}^{(m)}$. Фиксируя в качестве выходных данных $\mathbb{V}$ поток значений незаконченной работы в каждом сервере в момент поступления n-го задания, а в качестве входных данных $\mathbb{U} = (B_{n-l},\ldots,B_{n-1},\hat{W}_{n-l}^{(1)},\ldots,\hat{W}_{n-l}^{(m)},h_{n,k})$, $l = \min(k,n-k)$, оценки $\hat{W}_n^{(m)}$ получаем путем усреднения значений $\mathfrak{F}(\mathbb{U})$, т. е. усреднения результатов многократной имитации отрезка траектории процесса. Для данного n длина имитируемого отрезка составляет l, начальное значение остаточного времени на сервере m принимается равным $\hat{W}_{n-l}^{(m)}$, а значения действий и промежутков между ними фиксированы и совпадают с наблюденной предысторией $h_{n,k}$.

Отметим, что, несмотря на конечную глубину предыстории, используемой при расчете оценок $\mathsf{E}W_n^{(m)}$ на каждом шаге, фактически новая диспетчеризация (далее условимся обозначать ее AA, как и в предыдущих двух параграфах) учитывает, хотя и косвенно, всю предысторию[63]. Сохранение этой черты, присущей точным алгоритмам, является основой ее оптимизационных возможностей. Для определения оптимальных значений параметров нового алгоритма — глубины памяти k, числа имитируемых траекторий и порогового значения ξ — не удается предложить какого-либо теоретически обоснованного способа. Значения первых двух приходится искать в каждой задаче методом проб и ошибок. Для нахождения подходящего порогового значения могут применяться методы оптимизации на имитируемых тракеториях[64]. Однако, как показывают вычислительные эксперименты, бывает достаточно уже первого приближения, получаемого с помощью алгоритма расчета наилучшего порога из параграфа 3.2[65].

Перейдем к численным примерам и начнем с того же примера, которым начинается параграф 3.2. Рассматривается система из двух серверов суммар-

[63] И, разумеется, применима к системам не только с двумя, но и с произвольным числом серверов (как только задана пороговая стратегия).

[64] См. сноску на стр. 142.

[65] См. стр. 161 и *Алгоритм III*.

ной производительности 1, причем $v^{(1)} = 2/3$ и $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, распределение размера заданий — экспоненциальное со средним 1. Таким образом, загрузка системы ρ совпадает с λ. В таблице 10 даны значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе при различных значениях загрузки ρ, стратегиях RND и PROG, и новой стратегии AA. Значения параметров первых двух были выбраны оптимальным образом и приведены в таблице 4. Значения параметров k и ξ новой стратегии даны в таблице 11: первое было найдено ручным перебором, а второе — с помощью *Алгоритма III*.

Таблица 10 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Оптимальные значения параметров стратегий RND, PROG приведены в таблице 4, а стратегии AA — в таблице 11

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1.76 (1.76)	2.58 (2.63)	3.77 (3.87)	6.39 (6.56)	19.4 (20)
PROG-opt	1.76 (1.76)	2.41 (2.45)	3.22 (3.87)	5.17 (5.29)	15.0 (15.17)
AA	1.74 (1.76)	2.3 (2.36)	3.18 (3.22)	5.11 (5.2)	14.91 (15.06)

Таблица 11 — Значения параметров стратегии AA из таблице 10

ρ	0.1	0.3	0.5	0.7	0.9
глубина памяти k	2	2	3	4	5
пороговое значение ξ	0.79	0.75	0.66	0.48	0.4

Как видно из таблице 10, для диспетчеризации по полной предыстории AA, реализованной на основе аналитико–имитационного подхода, справедливы все те же выводы, что были сделаны для диспетчеризации, основанной на аналитическом подходе (см. стр. 143). Так, почти во всем диапазоне загрузки она позволяет улучшить значение целевого показателя по сравнению с PROG-opt — лучшей из ранее известных стратегий. Однако с ростом загрузки наблюдается уменьшение выигрыша.

Небезынтересно сравнить последние строки в таблице 3 и таблице 11. Сделав это, убеждаемся, что новая стратегия AA с опорой на метод статистических испытаний, уступает "точной" стратегии AA. Однако в случаях, когда подбор значений параметров "точной" стратегии затруднителен, дело может обстоять противоположным образом.

Обратимся теперь к примерам, иллюстрирующим влияние распределения размера заданий на соотношения между стратегиями[66]. В отличие от предыдущего примера будем иметь в виду лишь один целевой функционал — стационарное среднее время пребывания задания в системе. Кроме того, ожидая, что при наличии наблюдений, диспетчер может действовать лучше, чем при их отсутствии, добавим для сравнения диспетчеризацию JSQ — диспетчеризацию по наикратчайшей очереди. По-прежнему будем предполагать, что в системе имеется два сервера производительности $v^{(1)} = 2$ и $v^{(2)} = 1$, входящий поток заданий — пуассоновский интенсивности λ, а средний размер $\mathsf{E}S$ заданий равен единице. В качестве распределения $B(x) = \mathsf{P}\{S < x\}$ размера заданий рассмотрим

- равномерное распределение на интервале [0.1,1.9] (коэффициент вариации $C_B = 0.52$),
- распределение Парето с параметрами $\alpha = 2.5$, $x_m = 0.6$ (коэффициент вариации $C_B = 0.894$), т. е. $B(x) = 1 - \frac{x_m^\alpha}{x^\alpha}$, $x \geqslant x_m$,
- вырожденное распределение , т. е. $B(x) = \mathbf{1}_{(x>1)}$, $x \geqslant 0$ (коэффициент вариации $C_B = 0$).

В следующих трех таблицах (см. таблицах 12—14) даны значения стационарного среднего времени пребывания задания в системе при стратегиях RND, PROG, AA и JSQ и значениях загрузки $\rho = \lambda/3$, равномерно заполняющих интервал [0.2,0.8]. Значения параметров стратегий RND и PROG (см. таблица 15) были выбраны оптимальным образом: для RND — как решение задачи минимизации (5), для PROG — как результат оптимизации на имитируемых траекториях. Глубина памяти k в алгоритме AA динамически корректировалась в процессе имитации в диапазоне от 2 до 6; пороговые значения ξ (см. таблица 16) были получены по *Алгоритму III* и, следовательно, не зависели от выбранного распределения размера задания.

[66]Можно было бы проиллюстрировать и влияние на соотношения между стратегиями характеристик входящего потока. Однако получающиеся здесь результаты полностью подтверждают выводы, сделанные в параграфе 3.2, и поэтому опущены.

Таблица 12 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda/3$. Производительности серверов: $v^{(1)} = 2$, $v^{(2)} = 1$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет равномерное распределение на отрезке [0.1,1.9]. Оптимальные значения параметров стратегий указаны в таблицах 15 и 16

ρ	0.2	0.3	0.4	0.5	0.6	0.7	0.8
RND-opt	0.64	0.75	0.87	1.03	1.24	1.59	2.28
PROG-opt	0.64	0.73	0.79	0.87	0.99	1.20	1.62
AA	0.60	0.68	0.75	0.84	0.97	1.17	1.59
JSQ	0.62	0.67	0.73	0.81	0.92	1.11	1.47

Таблица 13 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda/3$. Производительности серверов: $v^{(1)} = 2$, $v^{(2)} = 1$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет распределение Парето с параметрами $\alpha = 2.5$, $x_m = 0.6$. Оптимальные значения параметров стратегий указаны в таблицах 15 и 16

ρ	0.2	0.3	0.4	0.5	0.6	0.7	0.8
RND-opt	0.69	0.83	0.98	1.19	1.49	1.97	2.95
PROG-opt	0.69	0.79	0.89	1.04	1.24	1.59	2.31
AA	0.65	0.73	0.84	0.98	1.19	1.51	2.20
JSQ	0.62	0.68	0.76	0.86	1.00	1.25	1.74

Данные в таблицах 12—14 еще раз свидетельствуют о том, что соотношения между стратегиями RND, PROG, AA, установленные в параграфе 3.2, сохраняются и при реализации диспетчеризации по полной предыстории с опорой на метод статистических испытаний. Во всех вычислительных экспериментах новая стратегия AA позволяла уменьшить стационарное среднее время пребывания задания в системе по сравнению лучшей из ранее известных стратегий PROG-opt; в приведенных трех примерах достигаемый выигрыш лежит в диапазоне от 1.5% до 10%.

Таблица 14 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda/3$. Производительности серверов: $v^{(1)} = 2$, $v^{(2)} = 1$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет вырожденное распределение в точке 1. Оптимальные значения параметров стратегий указаны в таблицах 15 и 16

ρ	0.2	0.3	0.4	0.5	0.6	0.7	0.8
RND-opt	0.61	0.70	0.81	0.94	1.11	1.39	1.94
PROG-opt	0.61	0.70	0.75	0.80	0.88	1.02	1.29
AA	0.59	0.63	0.68	0.75	0.83	0.97	1.25
JSQ	0.61	0.66	0.70	0.77	0.86	1.00	1.28
TP-opt	0.58	0.63	0.68	0.74	0.83	0.97	1.25

Таблица 15 — Оптимальные значения параметров стратегий RND и PROG из таблиц 12—14

ρ		0.2	0.3	0.4	0.5	0.6	0.7	0.8
Равном.	RND-opt	0	0.07	0.18	0.24	0.27	0.30	0.31
	PROG-opt	1	0.78	0.72	0.70	0.68	0.68	0.68
Парето	RND-opt	0.01	0.08	0.20	0.25	0.28	0.30	0.32
	PROG-opt	0.99	0.78	0.75	0.73	0.70	0.68	0.67
Вырожд.	RND-opt	0	0.03	0.16	0.23	0.27	0.30	0.31
	PROG-opt	1	0.86	0.75	0.68	0.67	0.67	0.67

Таблица 16 — Значения параметра ξ стратегии AA из таблиц 12—14

ρ	0.2	0.3	0.4	0.5	0.6	0.7	0.8
пороговое значение ξ	0.38	0.33	0.28	0.25	0.22	0.19	0.17

Таблица 17 — Относительный средний выигрыш в эффективности при стратегии AA по сравнению с PROG-opt — наилучшей из известных стратегий — в зависимости от коэффициента вариации C_B размера заданий (по данным из таблиц 12—14)

Распределение	Эксп.	Парето	Равном.	Вырожд.
C_B	1	0.894	0.52	0
выигрыш, %	2.8	4.0	5.5	6.1

Средний выигрыш при различных интенсивностях зависит от типа распределения размера задания и от коэффициента вариации[67] (см. таблицу 17): с уменьшением C_B (т. е. уменьшением дисперсии размера заданий) наблюдается его рост. С влиянием дисперсии можно также связать результат сравнения стратегий AA и JSQ. Диспетчеризация JSQ интересна тем, что использует наблюдения, хотя и неполные, но связанные с состояниями очередей. Интуитивно можно предположить, что такие наблюдения должны давать существенные преимущества по сравнению с полностью ненаблюдаемыми характеристиками нагрузки. Это предположение, однако, подтверждается только для больших значений дисперсии размера заданий. Для малых значений стратегия JSQ дает относительно меньший выигрыш по сравнению с оптимальной детерминированной диспетчеризацией PROG и, более того, проигрывает новой стратегии AA. Можно предположить, что последнее обстоятельство связано с усилением влияния предыстории в ситуации, когда мала дисперсия и соответственно в меньшей степени присутствует эффект перемешивания.

Наконец отметим, что результаты в таблице 14 особенно подчеркивают плодотворность предложенного в диссертации подхода к тому, как частично наблюдаемые системы с параллельным обслуживанием должны управляться. В самом деле, в последней строке таблице 14 добавлена для сравнения еще одна стратегия — пороговая стратегия (TP-opt). Именно она в полностью наблюдаемой системе является оптимальной с точки зрения стационарного среднего времени пребывания задания в системе. Сравнивая по таблице значения целевого функционала при стратегиях PROG-opt, AA и TP-opt видим, что существуют условия[68] в которых при полном отсутствии информации о текущем состоянии системы можно осуществлять диспетчеризацию[69] почти так же эффективно, как и при наличии исчерпывающей информации о нем[70].

Ниже на численных примерах показано, что все сделанные выше выводы о соотношениях между стратегиями RND, PROG и AA справедливы не только

[67] Совпадающего в рассматриваемых случаях с корнем из дисперсии, т. к. всюду $\mathsf{E}S = 1$.

[68] Как, например, в тех, что указаны в подписи к таблице 14.

[69] И этой диспетчеризацией является диспетчеризация по предыстории, но не программная стратегия.

[70] Информация о прошлых состояниях системы, т. е. более глубокая предыстория не дает дополнительных возможностей для управления: общая теория говорит о том, что оптимальная стратегия является марковской (см., например, [193, С. 253]).

при использовании в серверах дисциплины FIFO. Смена дисциплины обслуживания требует, конечно, и замены правила диспетчеризации (3.28).

Описание алгоритма при дисциплине PS

Пусть в каждом из $M \geqslant 2$ серверов, имеющих производительность соответственно $v^{(1)}, \ldots$ и $v^{(M)}$, реализована дисциплина справедливого разделения процессора. Напомним[71], что в этом случае поступившее в сервер задание сразу же начинает обслуживаться; очередь в традиционном понимании отсутствует. Обслуживание задания происходит с переменной скоростью до тех пор, пока его размер не станет равным нулю. В моменты поступлений новых заданий в сервер или ухода обслуженных происходят скачки скорости обслуживания.

Для того, чтобы иметь возможность сформулировать более или менее сложное правило диспетчеризации, необходимо иметь количественные оценки динамического состояния серверов. Рассмотрим следующие три способа для их получения:

а) состояние сервера m в момент t оценивается по числу заданий $N^{(m)}$, которые в этот момент находятся в сервере;

б) состояние сервера m в момент t оценивается по времени $W^{(m)}$, необходимому для завершения выполнения всех заданий, которые в этот момент находятся в сервере, при условии, что новые задания в систему больше не поступают, т. е., обозначая через $R_1, \ldots, R_{N^{(m)}}$ незаконченную к моменту t работу по каждому заданию, находящемуся в сервере m, имеем

$$W^{(m)} = \frac{R_1 + \ldots R_{N^{(m)}}}{v^{(m)}};$$

в) пусть T_1 — время до завершения выполнения самого маленького из имеющихся в сервере m в момент t заданий, $T_1 + T_2$ — время до окончания выполнения следующего по размеру задания, $\ldots$, $T_1 + \cdots + T_{N^{(m)}}$ — время до окончания выполнения всех заданий т. е., например, если

[71] За строгим математическим описанием можно обратиться к [75, С. 55], откуда и заимствовано приведенное далее описание дисциплины PS.

$R_1 < \cdots < R_{N^{(m)}}$, то

$$T_1 = \frac{R_1}{\frac{v^{(m)}}{N^{(m)}}},$$
$$T_2 = \frac{R_2 - T_1\frac{v^{(m)}}{N^{(m)}}}{\frac{v^{(m)}}{N^{(m)}-1}},$$
$$\cdots$$
$$T_{N^{(m)}} = \frac{R_{N^{(m)}} - T_1\frac{v^{(m)}}{N^{(m)}} - \cdots - T_{N^{(m)}-1}\frac{v^{(m)}}{2}}{v^{(m)}}.$$

Зафиксируем положительное число ζ и положим

$$\omega_1 = T_1, \quad \omega_2 = \zeta\omega_1 + T_2, \ldots, \omega_{N^{(m)}} = \zeta\omega_{N^{(m)}-1} + T_{N^{(m)}}. \tag{3.29}$$

В качестве количественной оценки динамического состояния сервера m в момент t будем брать число $\omega_{N^{(m)}}$.

Легко усмотреть, что (а) — это оценка состояния сервера по длине очереди, (б) — оценка по незаконченной работе. Последнюю оценку будем называть эвристической и заметим, что при $\zeta = 1$ она совпадает с оценкой (б).

Выбранные выше три способа количественной оценки динамического состояния серверов требуют таких наблюдений, которые исключены в рассматриваемых в диссертации системах с параллельным обслуживанием. Однако предположим, что необходимые наблюдения все-таки доступны. Рассмотрим некоторый момент поступления в систему очередного задания. Пусть динамическое состояние сервера m оценивается[72] по одному из способов (а)—(в); обозначим оценку через $\kappa^{(m)}$. Поскольку речь идет о моменте поступления задания, то количественная оценка состояния сервера возможна, вообще говоря, как без учета нового задания, так и в предположении, что оно отправлено именно на данный сервер. В последнем варианте будем снабжать обозначения дополнительным символом "+"[73]. Сформулируем теперь два правила диспетчеризации.

Первое правило основано на непосредственном сравнении количественных оценок состояний серверов: поступившее в момент t_n задание направляется на сервер с номером y_n, выбранным равновероятно из множества

$$\left\{ m : v^{(m)} = \max_{j\in\mathcal{J}} v^{(j)} \right\}, \tag{3.30}$$

[72]Предполагается, что для всех серверов применяется один и тот же способ оценки.

[73]Например, запись $\kappa_+^{(m)} = W_+^{(m)}$ означает, что в момент поступления задания в систему оценка состояния сервера m трактуется как время до окончания выполнения всех заданий, которые были в этом сервере, плюс новое задание.

где $\mathcal{J} = \{j : \kappa^{(j)} = \min_{m\in\{1,\dots,M\}} \kappa^{(m)}\}$. Оценивая $\kappa^{(m)}$ по способу (а), получим диспетчеризацию по минимальной длине очереди т. е. JSQ, а по способу (б) — алгоритм, известный в литературе[74], как Myopic. При оценке динамического состояния серверов по способу (в) получим диспетчеризацию по минимальной эвристической оценке (далее — Heuristic). Отметим, что в (3.30) вместо оценок $\kappa^{(m)}$ можно подставить и оценки $\kappa_+^{(m)}$. При этом легко понять, что, вне зависимости от используемых оценок, алгоритм JSQ будет приводить к одним и тем же результатам. Однако с двумя другими алгоритмами это не так.

Второе правило диспетчеризации основано на сравнении прогнозируемого увеличения количественных оценок состояний серверов: поступившее в момент t_n задание направляется на сервер с номером y_n, выбранным равновероятно из множества (3.30), но теперь $\mathcal{J} = \{j : \kappa^{(j)} = \min_{m\in\{1,\dots,M\}}(\kappa_+^{(m)} - \kappa^{(m)})\}$.

Чтобы воспользоваться сформулированными правилами необходимо заменить отсутствующие данные о $\kappa^{(1)},\dots,\kappa^{(M)}$ и/или $\kappa_+^{(1)},\dots,\kappa_+^{(M)}$ статистическими оценками, которые получаются на основе доступных наблюдений. Сделать это можно следующим образом. Пусть в момент t_n в систему поступило задание. Пусть $h_{n,k} = (y_{n-k},\Delta_{n-k},\dots,y_{n-1},\Delta_{n-1})$ — предыстория к моменту t_n глубины k, составленная из имеющихся наблюдений, то есть из управлений y_i и промежутков Δ_i между моментами совершения действий y_i и y_{i-1}. Пусть также $s_{n,k} = (s_{n-k},\dots,s_{n-1})$ — вектор независимых реализаций сл.в., имеющих функции распределения размера $(n-k)$-го, …, $(n-1)$-го по счету задания. С помощью вектора наблюдений $h_{n,k}$, а также с помощью вектора $s_{n,k}$ сымитируем независимый от основного процесса отрезок траектории следующим образом. В начальный момент в пустую (вспомогательную) систему поступает задание объемом s_{n-k}, которое направляется на сервер y_{n-k}. Спустя время Δ_{n-k} поступает задание объемом s_{n-k+1}, которое направляется на сервер y_{n-k+1}, и так далее, вплоть до поступления задания объемом s_{n-1}. Задания обрабатываются точно так же как в основной системе. Для n-го задания решение выбирается на основе одного из двух описанных выше правил диспетчеризации. Обозначим это управление $\hat{y}_1$. Повторим процедуру точно таким же образом еще $r \geqslant 0$ раз. В результате получим набор управлений $(\hat{y}_1,\hat{y}_2,\dots,\hat{y}_{r+1})$. В качестве искомого

[74]Если хотя бы два сервера имеют различную производительность. В случае одинаковых серверов алгоритм Myopic идентичен алгоритму LWL.

управления y_n в основной системе примем то управление, которое встречается в наборе наиболее часто[75].

Таким образом, каждому описанному выше алгоритму (JSQ, Myopic и Heuristic) соответствует "приближенный" алгоритм[76] по предыстории (обозначаемый соответственно $\mathsf{AA}_{\mathsf{JSQ}}$, $\mathsf{AA}_{\mathsf{Myopic}}$ и $\mathsf{AA}_{\mathsf{Heuristic}}$), действующий в условиях отсутствия наблюдений за состоянием системы. Каждый из новых алгоритмов является параметрическим: неизвестными являются значения параметра k (глубина предыстории), r (число имитируемых траекторий) и ζ (константа из (3.29)). Отыскание их оптимальных значений, по существу, представляет собой отдельную, пока нерешенную задачу.

Рассмотрим несколько примеров, иллюстрирующих соотношения между описанными стратегиями, а также стратегиями RND и PROG, при различных конфигурациях системы. Ограничимся системой с двумя серверами[77]. Пусть производительность первого равна $v^{(1)} = 2$, второго — $v^{(2)} = 1$, входящий поток — пуассоновский с интенсивностью λ, средний размер $\mathsf{E}S$ заданий равен единице. В качестве распределения $B(x) = \mathsf{P}\{S < x\}$ размера заданий рассмотрим

- экспоненциальное распределение (коэффициент вариации $C_B = 1$),
- равномерное распределение на интервале [0.5,1.5] (коэффициент вариации $C_B = 0.289$),
- распределение Вейбулла с параметрами $a = 1.(41)$, $b = 0.5$, т. е. $B(x) = 1 - e^{-ax^b}$, $x \geqslant 0$ (коэффициент вариации $C_B = 2.45$),
- вырожденное распределение , т. е. $B(x) = \mathbf{1}_{(x>1)}$, $x \geqslant 0$ (коэффициент вариации $C_B = 0$).

Таким образом, загрузка системы ρ совпадает с $\lambda/3$. В таблице 18 приводятся значения стационарного среднего времени пребывания задания в системе при восьми стратегиях и $\rho = 0.33$. Значения параметров стратегий даны в таблице 19. Для RND они были выбраны оптимальным образом (см. (6)); эти же

[75] Возможные конфликты разрешаются в пользу более быстрых серверов и равновероятным выбором среди них.

[76] Чтобы не загромождать изложение здесь говорится об одном "приближенном" алгоритме, и не уточняется о каком именно. Но важно помнить общую картину: каждому из способов оценки динамического состояния серверов (а)—(в) соответствует два "приближенных" алгоритма, из которых в каждом конкретном случае следует выбрать лучший. В приводимых ниже примерах приводятся результаты именно для лучшего варианта.

[77] Выводы, которые делаются далее, как показывают вычислительные эксперименты (см. примеры в [304, таблицы 2–5]), остаются в силе и для систем с большим числом серверов. Необходимо, однако, иметь в виду и сноску на стр. 145.

значения использовались для параметров стратегии PROG. Что касается новых стратегий $\mathsf{AA}_{\mathsf{JSQ}}$, $\mathsf{AA}_{\mathsf{Myopic}}$ и $\mathsf{AA}_{\mathsf{Heuristic}}$, то в таблицах приведены наилучшие значения параметров, которых удалось добиться[78]. Поэтому может быть неслучайно, что иногда один и тот же тип алгоритмов оказывается лучшим.

Таблица 18 — Значения стационарного среднего времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различных распределениях размера заданий. Загрузка системы $\rho = 0.33$. Производительности серверов: $v^{(1)} = 2$, $v^{(2)} = 1$. Входящий поток — пуассоновский с интенсивностью $\lambda = 1$. Значения параметров стратегий даны в таблице 19

Распределение размера заданий	Вырожд.	Равном.	Эксп.	Вейб.
RND-opt	0.915	0.913	0.914	0.914
PROG-RND-opt	0.846	0.847	0.858	0.891
JSQ	0.721	0.724	0.732	0.737
$\mathsf{AA}_{\mathsf{JSQ}}$	0.711	0.754	0.894	0.980
Myopic (первое правило)	0.711	0.714	0.729	0.747
$\mathsf{AA}_{\mathsf{Myopic}}$	0.711	0.750	0.872	0.973
Heuristic (второе правило)	0.698	0.699	0.699	0.696
$\mathsf{AA}_{\mathsf{Heuristic}}$	0.698	0.733	0.808	0.875

Таблица 19 — Значения параметров стратегий из таблице 18

Распределение размера заданий	Вырожд.	Равном.	Эксп.	Вейб.
RND-opt	0.828	0.828	0.828	0.828
PROG-RND-opt	0.828	0.828	0.828	0.828
$\mathsf{AA}_{\mathsf{JSQ}}$	$k = 2$	$k = 2$	$k = 5$	$k = 12$
$\mathsf{AA}_{\mathsf{Myopic}}$	$k = 3$	$k = 2$	$k = 5$	$k = 10$
Heuristic	$\zeta = 2.25$	$\zeta = 3$	$\zeta = 2.25$	$\zeta = 2.25$
$\mathsf{AA}_{\mathsf{Heuristic}}$	$k=2, \zeta=2$	$k=2, \zeta=2.75$	$k=5, \zeta=2.5$	$k=5, \zeta=2.75$

Напомним, что роль точки отсчета играют стратегии RND-opt и PROG-RND-opt: они является основными объектами сравнения для новых диспетчеризаций $\mathsf{AA}_{\mathsf{JSQ}}$, $\mathsf{AA}_{\mathsf{Myopic}}$ и $\mathsf{AA}_{\mathsf{Heuristic}}$. Другие стратегии (JSQ, Myopic и Heuristic), которые, вообще говоря, неприменимы в изучаемых системах, служат методологическим целям. Как видно из таблицы 18 во всех случаях хотя бы одна новая диспетчеризация позволяет улучшить значение целевого функционала. Вычислительные эксперименты показывают, что разброс достигаемого выигрыша очень большой: от 1% до 70%. И, как уже неоднократно упоминалось

[78] При этом число r имитируемых траекторий не варьировалось вовсе и принималось равным 10.

выше, с увеличением "степени случайности" распределения размера задания имеет место уменьшение выигрыша. Кроме того, численные результаты в таблице 18 свидетельствуют о том, что новые, "приближенные" алгоритмы могут не уступать или вовсе превосходить в эффективности точные, основанные на наблюдениях. Интуитивно оправданным кажется предположение, что наблюдения должны давать существенные преимущества точным алгоритмам. Однако оно подтверждается только для больших значений дисперсии размера заданий.

Сменим входящий поток на "более случайный". Предположим, что времена между поступлениями имеют гиперэкспоненциальное распределение с ф. р. $F(x) = 1 - 0.5e^{-\frac{2}{3}x} - 0.5e^{-2x}$, $x \geqslant 0$. Таким образом, среднее время между поступлениями заданий равно единице, коэффициент вариации $C_F \approx 1.225$ (дисперсия ≈ 1.5), а загрузка системы ρ, как и в предыдущем примере, равна 0.33. В таблице 20 приводятся значения стационарного среднего времени пребывания задания в системе при тех же стратегиях и распределениях размера заданий, что были рассмотрены в предыдущем примере. Значения всех параметров стратегий, которые подбирались на имитируемых траекториях, даны в таблице 21.

Таблица 20 — Значения стационарного среднего времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различных распределениях размера заданий. Загрузка системы $\rho = 0.33$. Производительности серверов: $v^{(1)} = 2$, $v^{(2)} = 1$. Входящий поток — гиперэкспоненциальный с интенсивностью 1. Значения параметров стратегий указаны в таблице 21

Распределение размера заданий	Вырожд.	Равном.	Эксп.	Вейб.
RND-opt	1.04	1.034	1	0.958
PROG-RND-opt	0.913	0.911	0.911	0.919
JSQ	0.794	0.796	0.789	0.773
$\mathsf{AA}_{\mathsf{JSQ}}$	0.777	0.821	0.833	1.047
Myopic (первое правило)	0.777	0.78	0.784	0.783
$\mathsf{AA}_{\mathsf{Myopic}}$	0.777	0.817	0.924	1.012
Heuristic (второе правило)	0.77	0.772	0.751	0.728
$\mathsf{AA}_{\mathsf{Heuristic}}$	0.77	0.791	0.874	0.904

Как видно из таблицы 20, с изменением характера входного потока качественная картина поведения целевого функционала в сравнении с предыдущим примером не поменялась. Во всех рассмотренных случаях нашлась (по

Таблица 21 — Значения параметров стратегий из таблицы 20

Распределение размера заданий	Вырожд.	Равном.	Эксп.	Вейб.
RND-opt	0.79	0.789	0.789	0.795
PROG-RND-opt	0.79	0.789	0.789	0.795
$\mathsf{AA}_{\mathsf{JSQ}}$	$k=2$	$k=2$	$k=2$	$k=5$
$\mathsf{AA}_{\mathsf{Myopic}}$	$k=5$	$k=2$	$k=10$	$k=10$
Heuristic	$\zeta=2.25$	$\zeta=2$	$\zeta=2.25$	$\zeta=2.25$
$\mathsf{AA}_{\mathsf{Heuristic}}$	$k=2, \zeta=2.25$	$k=2, \zeta=2.75$	$k=10, \zeta=2.75$	$k=5, \zeta=2.75$

крайней мере одна) новая диспетчеризация, позволившая улучшить значение стационарного среднего времени пребывания задания в системе по сравнению с наилучшей из ранее известных стратегий. Отмеченная выше тенденция уменьшения выигрыша с увеличением дисперсии размера задания сохраняется. Как показывают вычислительные эксперименты для новых диспетчеризаций имеет место своеобразная нечувствительность (с качественной точки зрения) к степени "случайности" входящего потока. В самом деле, обратимся к рисункам 14—17, на которых приведены значения стационарного среднего времени пребывания для новых диспетчеризаций $\mathsf{AA}_{\mathsf{JSQ}}$, $\mathsf{AA}_{\mathsf{Myopic}}$ и $\mathsf{AA}_{\mathsf{Heuristic}}$ в зависимости от глубины предыстории k и коэффициента вариации C_F входящего потока. Рассматриваемая система состоит из двух серверов производительности $v^{(1)}=2$ и $v^{(2)}=1$; входящий поток — гиперэкспоненциальный со средним 1, размеры заданий имеют распределение Вейбулла со средним 1.

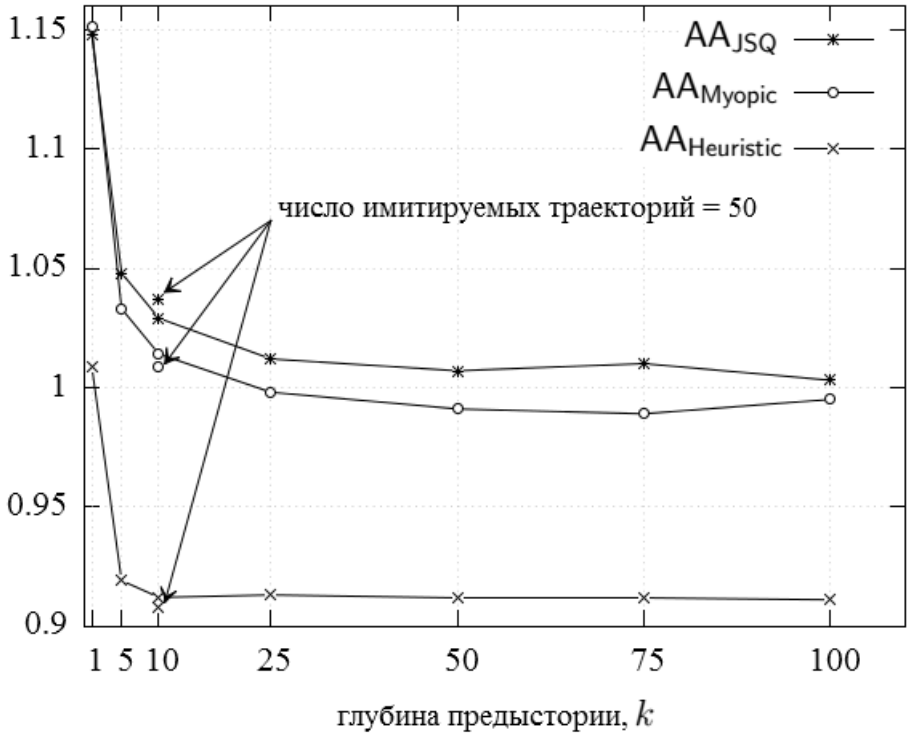


Рисунок 14 — Зависимость от глубины предыстории стационарного среднего время пребывания задания в системе из двух серверов. Производительности серверов: $v^{(1)} = 2$ и $v^{(2)} = 1$. Входящий поток — гиперэкспоненциальный со средним 1 и коэффициентом вариации $C_F \approx 1.225$; $F(x) = 1 - 0.75e^{-1.5x} - 0.25e^{-0.5x}$, $x \geqslant 0$. Размер заданий имеет распределение Вейбулла с параметрами $a = 1.(41)$ и $b = 0.5$, средним 1 и $C_B \approx 2.45$. Загрузка системы $\rho = 0.33$. Число имитируемых траекторий $r = 10$

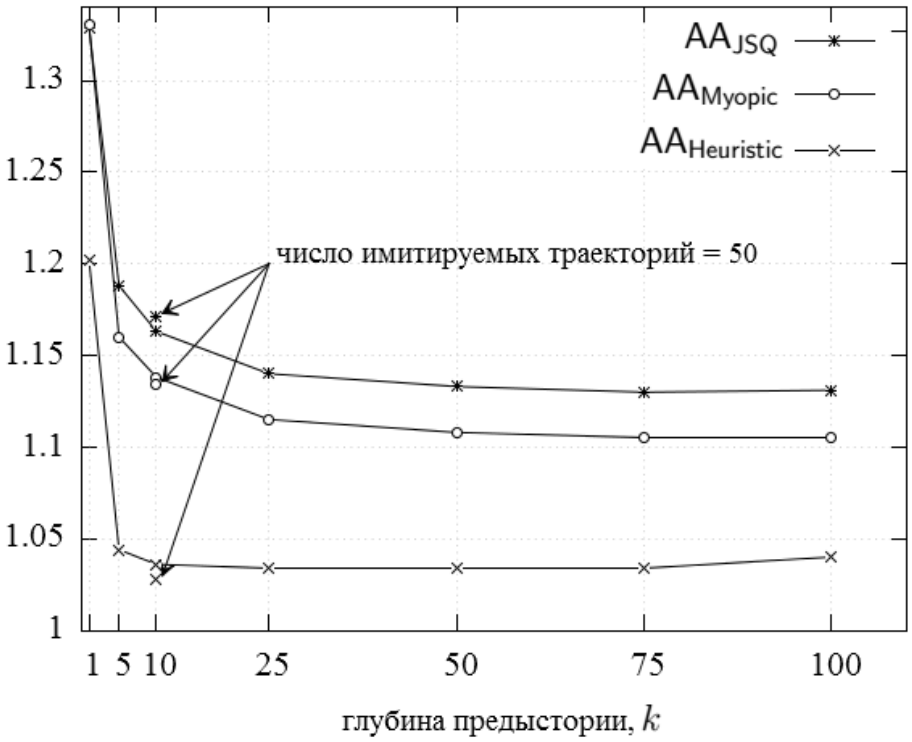


Рисунок 15 — Зависимость от глубины предыстории стационарного среднего время пребывания задания в системе из двух серверов. Производительности серверов: $v^{(1)} = 2$ и $v^{(2)} = 1$. Входящий поток — гиперэкспоненциальный со средним 1 и коэффициентом вариации $C_F = 2$; $F(x) = 1 - 0.9571e^{-2x} - 0.1429e^{-0.25x}$, $x \geqslant 0$. Размер заданий имеет распределение Вейбулла с параметрами $a = 1.(41)$ и $b = 0.5$, средним 1 и $C_B \approx 2.45$. Загрузка системы $\rho = 0.33$. Число имитируемых траекторий $r = 10$

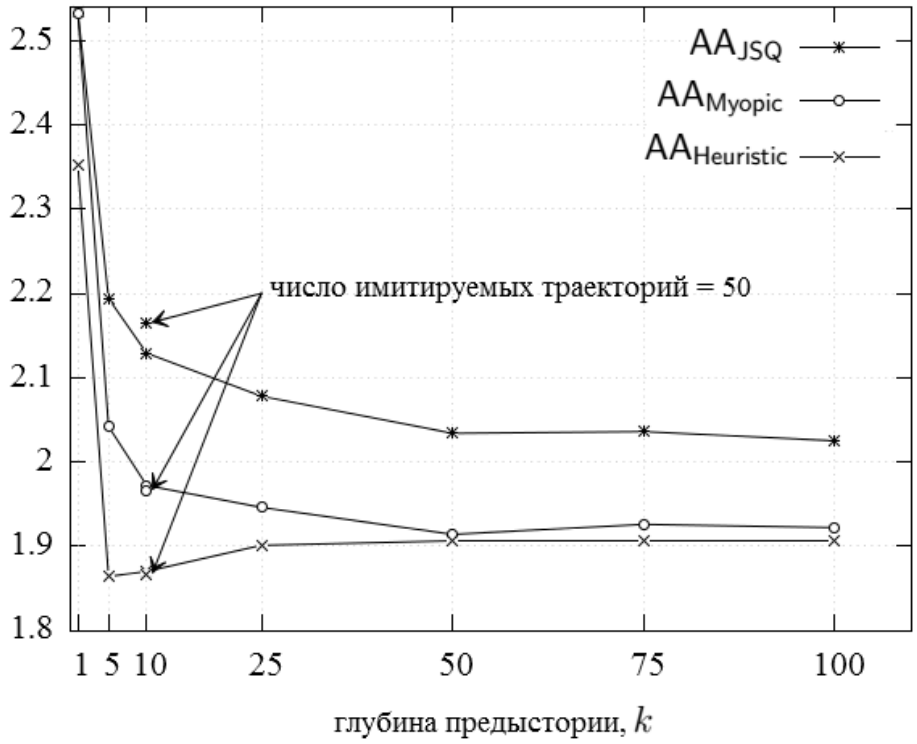


Рисунок 16 — Зависимость от глубины предыстории стационарного среднего время пребывания задания в системе из двух серверов. Производительности серверов: $v^{(1)} = 2$ и $v^{(2)} = 1$. Входящий поток — гиперэкспоненциальный со средним 1 и коэффициентом вариации $C_F \approx 3.924$; $F(x) = 1 - 0.918e^{-5x} - 0.082e^{-0.1x}$, $x \geqslant 0$. Размер заданий имеет распределение Вейбулла с параметрами $a = 1.(41)$ и $b = 0.5$, средним 1 и $C_B \approx 2.45$. Загрузка системы $\rho = 0.33$. Число имитируемых траекторий $r = 10$

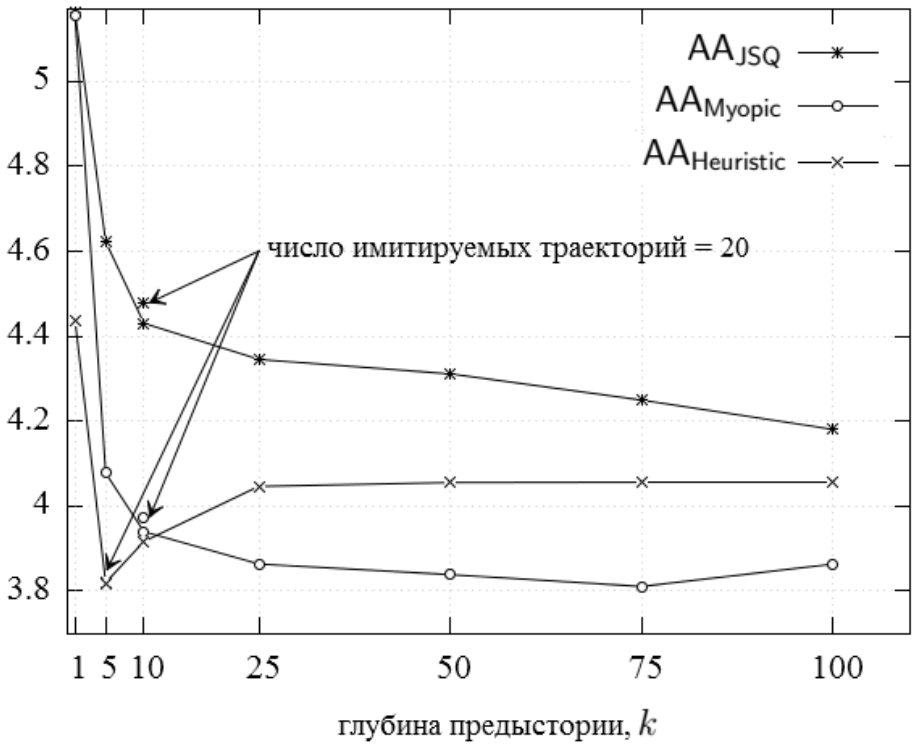


Рисунок 17 — Зависимость от глубины предыстории стационарного среднего время пребывания задания в системе из двух серверов. Производительности серверов: $v^{(1)} = 2$ и $v^{(2)} = 1$. Входящий поток — гиперэкспоненциальный со средним 1 и коэффициентом вариации $C_F \approx 5.925$; $F(x) = 1 - 0.95748e^{-10x} - 0.0452e^{-0.05x}$, $x \geqslant 0$. Размер заданий имеет распределение Вейбулла с параметрами $a = 1.(41)$ и $b = 0.5$, средним 1 и $C_B \approx 2.45$. Загрузка системы $\rho = 0.33$. Число имитируемых траекторий $r = 10$

Как видно из рискунков, после определенного значения k (в примерах — это $k \geqslant 25$) значение целевого функционала (по крайней мере при лучшей из диспетчеризаций) практически перестает зависеть[79] как от глубины предыстории, так и от того, насколько случаен входящий поток. Этот вывод подтверждают и другие вычислительные эксперименты. Однако, с количественной точки зрения влияние дисперсии входящего потока на значение целевого функионала существенно: чем больше дисперсия, тем больше и среднее время пребывания задания в системе. Кроме того, с ростом дисперсии изменяется и эффективность диспетчеризаций. Так, наилучшая при малых значениях дисперсии диспетчеризация $\mathsf{AA}_{\mathsf{Heuristic}}$ уступает лидирующее место диспетчеризации $\mathsf{AA}_{\mathsf{Myopic}}$ при больших значениях дисперсии.

Подводя итог параграфам 3.3 и 3.4, отметим следующее. Результаты вычислительных экспериментов указывают на принципиальную возможность улучшать значения целевых функционалов в частично наблюдаемых системах с параллельным обслуживанием на основе предложенного аналитико–имитационного подхода и диспетчеризаций, учитывающих предысторию решений и моментов их принятия. Сам подход к порождению диспетчеризаций является универсальным: взяв за основу любой из алгоритмов, применимых в полностью наблюдаемых системах, можно заменить отсутствующие данные статистическими оценками, которые получаются на основе доступных наблюдений, и получить выигрыш целевой функции по сравнению со всеми[80] ранее известными стратегиями.

[79] По-видимому, слаба и зависимость от числа имитируемых траекторий. Впрочем здесь, разумеется, сложно сделать окончательный вывод, имея в виду распределения времен обслуживания со сверхвысоким коэффициентом вариации.

[80] См. сноску на стр. 142.

Глава 4. Дальнейшие исследования алгоритмов управления в отсутствие динамической информации

Рассмотрим частично наблюдаемую систему с параллельным обслуживанием, состоящую из $M \geqslant 2$ серверов, в которую поступает рекуррентный поток заданий. Интервалы между поступлениями заданий образуют последовательность независимых случайных величин с распределением $F(x)$, средним $\int_0^\infty x dF(x) = \lambda^{-1}$ и коэффициентом вариации $C_F < \infty$. Задания имеют случайный объем (размер), который определяется одним и тем же распределением $B(x) = \mathsf{P}\{S < x\}$; его среднее значение и коэффициент вариации далее обозначаются соответственно через $\mathsf{E}S$ и C_B. Каждое поступившее задание должно быть немедленно направлено на один из серверов. Цель диспетчера, осуществляющего этот выбор (в автоматическом или ручном режиме), — минимизировать стационарное среднее время пребывания задания в системе.

Напомним, что ограничение частичной наблюдаемости связано с тем, что при принятии решений диспетчеру недоступна информация о текущем (прошлом или будущем) состоянии системы (например, длины очередей, незаконченная работа в каждом сервере и т. п.). В то же время известны распределения $F(x)$ и $B(x)$, производительности[1] серверов $v^{(1)}, \ldots, v^{(M)}$ и полная предыстория принятых решений (включая моменты времени, в которые эти решения принимались). В каждом сервере имеется очередь неограниченной емкости для хранения заданий и один процессор для обработки, причем выбор на обслуживание происходит в соответствии с одной из консервативных дисциплин[2] (например, FIFO, LIFO, RANDOM, SJF, PS, PLIFO, FB, PSJF или SPRT). Серверы работают независимо, без обмена заданиями и являются абсолютно надежными.

Как уже было отмечено во введении и в главе 3 недоступность информации о текущем состоянии системы при выборе управления очень сужает множество допустимых стратегий. Из всех ранее известных в научной лите-

[1] Причем не все $v^{(m)}$ равны между собой.

[2] Выбор конкретной дисциплины обслуживания (и целевого функционала) накладывает ограничения на моменты распределений $F(x)$ и $B(x)$: не оговаривая это особо, соответствующие ограничения предполагаются выполненными. Например, при пуассоновском потоке и дисциплине FIFO считается, что для размеров заданий допустимы только распределения с конечным вторым моментом. См. также сноску на стр. 134.

ратуре для рассматриваемых систем применимы лишь две — случайный выбор (далее — RND) и детерминированный выбор (т. е. программная стратегия, далее — PROG). Обе из них требуют предварительной оценки $M-1$ параметров[3].

В предыдущей главе предложены принципиально новые конструктивные подходы для порождения диспетчеризаций, которые (по крайней мере в случае наиболее популярных критериев оптимальности) превосходят как RND, так и PROG во всем диапазоне изменений значений исходных параметров системы. Однако и они не свободны от недостатков, главным из которых, пожалуй, является вычислительная сложность.

В этой главе описывается новый простой и эффективный алгоритм, который (хотя и уступает алгоритмам главы 3, но) успешно конкурирует как со стратегий RND, так и с PROG даже при оптимальных значения параметров последних. Эти свойства, вкупе с тем обстоятельством, что он требует для своей настройки оценки меньшего числа параметров[4], дают основания назвать его лучшим для частично наблюдаемых стохастических систем с параллельным обслуживанием.

4.1 Алгоритмы управления на основе виртуальных вспомогательных процессов при использовании в однопроцессорных серверах консервативных дисциплин

Пусть $0 \leqslant t_1 < t_2 < \ldots$ — последовательность моментов поступления заданий в систему. Решение (действие), принимаемое в момент t_n относительно вновь поступившего задания, обозначим через y_n. Полагаем, что $y_n = m$, если n-е по счету задание направлено на сервер m.

Предположим, что рассматриваемая система полностью наблюдаема. Тогда общеупотребительная схема построения правила, по которому выбирается

[3]Напомим, что, строго говоря, так обстоит дело только с диспетчеризацией RND. В случае стратегии PROG требуется решать более сложную задачу: находить оптимальную бесконечную детерминированную последовательность (действий). Подходящие для этого методы пока не разработаны (см. обсуждение, начиная со стр. 20). Но известны алгоритмы, реализующие основной замысел детерминированных стратегий — сделать входящий поток на каждый сервер более регулярным (см. [161]), чем тот, что порождается рандомизированным выбором; наиболее эффективные из них зависят от $M-1$ параметров.

[4]В том варианте алгоритма, который обсуждается далее, — всего одного.

сервер для вновь поступившего задания, заключается в следующем. Задана некоторая функция, которая позволяет количественно оценить состояние каждого сервера на основе текущей длины очереди и остаточного объема работы для всех заданий в очереди. После того как такие оценки выполнены для всех серверов, задание направляется на сервер с минимальной оценкой. В случае, когда несколько серверов имеют одинаковую минимальную оценку, выбирается сервер с наибольшей производительностью. Если по-прежнему имеется неоднозначность, то она разрешается равновероятным выбором. Рассмотрим следующие варианты выбора оценочной функции:

- текущее число заданий в сервере, т. е. алгоритм JSQ;
- суммарный остаточный размер всех заданий в очереди, включая обрабатываемое задание, т. е. алгоритм LWL;
- суммарное остаточное время для окончания выполнения всех заданий в очереди, включая обрабатываемое задание и вновь поступившее и (условно) присоединенное к очереди, в предположении, что больше задания в систему не поступают, т. е. алгоритм Myopic.

В случае алгоритма JSQ оценка состояния сервера строится по "наличному составу" заданий; в двух других случаях учитывается и вновь поступившее задание. Однако отметим, что ни в одной из этих стратегий не учитывается важное обстоятельство: задания, которые поступят позже, могут изменить время выполнения имеющихся заданий.

Теперь вернемся к исходной постановке, т. е. предположим, что диспетчеру недоступна информация о текущем (прошлом или будущем) состоянии системы. Пусть наряду с основным процессом поступления и обслуживания заданий в заданной системе обслуживания, имеется еще $k \geqslant 1$ вспомогательных процессов, моделирующих точно такую же систему, т. е. с такими же серверами и пр. Функционирование вспомогательных процессов в точности копирует работу основного процесса в части моментов поступления и распределения заданий, но различается размерами заданий. Кроме этого, различие между основным и вспомогательными процессами заключается в характере наблюдений. Если для основного процесса, наблюдения по условию ограничены моментами поступления заданий и совершенными управлениями, то вспомогательные процессы полностью наблюдаемы.

Более подробно и точно диспетчеризацию при неполном наблюдении можно описать индуктивно. Начальные состояния всех процессов одинаковые

(например, соответствуют пустой системе). Пусть все процессы (то есть основной и k вспомогательных) проработали до некоторого момента t_n поступления очередного задания в основном процессе. Этот момент принудительно считается моментом поступления задания и для каждого из вспомогательных процессов. Однако размер нового задания определяется для каждого из вспомогательных процессов индивидуально и независимо с помощью заданной (и известной по предположению) функции распределения размера заданий. Каждый из вспомогательных процессов, исходя из заданного в нем правила диспетчеризации, определяет номер сервера, на который следовало бы отправить его собственное новое задание. Номера серверов, выбранные для различных вспомогательных процессов, вообще говоря, разные, однако среди них есть, по крайней мере, один наиболее часто встречающийся номер. Сервер с этим номером и выбирается для задания, поступившего в момент t_n в основном процессе. Более того, серверы с указанным номером являются фактическими приемниками новых заданий во всех вспомогательных процессах, независимо от того, что было выбрано в результате работы индивидуального правила диспетчеризации. Далее работа основного и вспомогательных процессов (то есть имитация выполнения заданий) протекает независимо друг от друга вплоть до следующего момента t_{n+1} поступления задания в основном процессе. Подчеркнем, что поступление заданий во вспомогательных процессах происходит только в моменты t_n, t_{n+1}, ... поступления заданий в основном процессе. Отметим также, что основной процесс отображает "реальный" процесс (хотя и может быть заменен имитационной моделью), в то время как вспомогательные процессы являются чисто виртуальными компьютерными моделями.

Пусть теперь имеется всего один вспомогательный процесс т. е. $k = 1$. Выберем в качестве правила диспетчеризации для вспомогательного процесса правило Myopic. Предположим, что все задания, поступающие на серверы вспомогательного процесса (идентичные с серверами основного процесса), имеют одинаковый размер $\zeta > 0$. Пусть t_n и t_{n+1} — два последовательных момента поступления заданий в основном процессе. Обозначим через $z_n^{(m)}$ время, необходимое для выполнения всех заданий, имеющихся в сервере m вспомогательного процесса в момент t_n, с учетом задания, распределенного в этот момент каким-то образом на один из серверов того же вспомогательного процесса. Через $\tilde{z}_{n+1}^{(m)}$ обозначим время, необходимое для выполнения всех заданий, имеющихся в сервере m вспомогательного процесса в момент t_{n+1}, без учета

задания, поступившего в этот момент. Очевидно,

$$\tilde{z}_{n+1}^{(m)} = \max\left(0, z_n^{(m)} - (t_{n+1} - t_n)\right).$$

Положим

$$y_{n+1} = \mathrm{argmin}_{1 \leqslant m \leqslant M}\left(\tilde{z}_{n+1}^{(m)} + \frac{\zeta}{v^{(m)}}\right).$$

Неоднозначность при нахождении минимума разрешается прежним способом. Число y_{n+1} служит номером сервера, на который направляется задание, поступившее в момент t_{n+1}. Причем, для основного процесса размер этого задания определяется согласно заданному распределению, а для вспомогательного процесса размер задания равняется ζ. Таким образом[5],

$$z_{n+1}^{(m)} = \begin{cases} \tilde{z}_{n+1}^{(y_{n+1})} + \frac{\zeta}{v^{(y_{n+1})}}, & \text{если } m = y_{n+1}, \\ \tilde{z}_{n+1}^{(m)}, & \text{иначе.} \end{cases}$$

Формальное описание алгоритма выбора управления для задания, поступившего в момент t_{n+1}, представлено ниже. Помимо исходных значений M, $v^{(1)}, \ldots, v^{(M)}$ входными данными являются: значения[6] незаконченная работа $z_n^{(1)}, \ldots, z_n^{(M)}$ в каждом из серверов вспомогательного процесса в момент t_n, моменты поступлений t_{n+1} и t_n текущего и предыдущего заданий, и значение параметра алгоритма ζ. Выходные данные — это номер сервера y_{n+1}, на который следует отправить поступившее в момент t_{n+1} задание, и значения $z_{n+1}^{(1)}, \ldots, z_{n+1}^{(M)}$.

Алгоритм IV. Псевдокод алгоритма выбора управления для задания, поступившего в момент t_{n+1}, $n \geqslant 0$

1: **for** $m = 1 \to M$ **do**
2: $\quad z_{n+1}^{(m)} = \max\left(0, z_n^{(m)} - (t_{n+1} - t_n)\right)$
3: $y_{n+1} = \mathrm{argmin}_{1 \leqslant m \leqslant M}\left(z_{n+1}^{(m)} + \frac{\zeta}{v^{(m)}}\right)$
4: $z_{n+1}^{(y_{n+1})} = z_{n+1}^{(y_{n+1})} + \frac{\zeta}{v^{(y_{n+1})}}$
5: **return** $y_{n+1}, z_{n+1}^{(1)}, \ldots, z_{n+1}^{(M)}$

---

[5] На первый взгляд может показаться, что оптимальное ζ является также и оптимальным (порогом) для системы из параллельных СМО $\cdot \,|\, D \,|\, 1 \,|\, \infty |$ FIFO (см., например, [272]). Но это не так. Значение ζ во вспомогательном процессе зависит от многих факторов: распределения $B(x)$, дисциплины обслуживания, типа входящего потока и пр.

[6] Значение $z_0^{(m)}$ есть незаконченная работа в сервере n вспомогательного процесса в момент t_0 начала функционирования системы.

Ниже представлен набор численых примеров, цель которого — показать, что предложенный *Алгоритм IV* (далее — AA) успешно конкурирует с диспетчеризациями RND и PROG, а в сбалансированных[7] системах часто и превосходит их по критерию стационарного среднего времени пребывания задания в системе.

4.2 Примеры и дополнения

Предположим, что выбор на обслуживание в каждом сервере происходит в соответствии с дисциплиной FIFO. Рассмотрим сначала полностью марковский случай т. е. предположим, что входящий поток — пуассоновский с интенсивностью λ, а распределение размера заданий — экспоненциальное со средним 1. Пусть система состоит из двух серверов суммарной производительности 1, причем $v^{(1)} = 2/3$ и $v^{(2)} = 1/3$. Таким образом, загрузка системы ρ совпадает с λ. Из всех примеров, которые будут рассмотрены далее, этот пример является наиболее показательным: в нем известна оптимальная стратегия распределения заданий[8] — это программная стратегия (последовательность Битти; см. стр. стр. 22).

В таблице 22 даны значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе при различных значениях загрузки ρ и трех стратегиях: RND, PROG и AA. Значения их параметров (см. таблица 23) при каждом значении загрузки были выбраны оптимальным образом: для RND — как решение задачи минимизации (5), для PROG и AA — как результат оптимизации на имитируемых траекториях.

По таблице 22 видно, что, по сравнению с рандомизированной стратегией, новый алгоритм позволяет уменьшить значение стационарного среднего времени пребывания в системе при любом значении загрузки ρ. Как и с описанными в предыдущей главе разновидностями стратегии AA, имеет место тенденция:

[7] Под этим подразумевается, что производительности серверов не слишком сильно отличаются друг от друга, т. е. среди серверов нет настолько быстрых, что выгодно отправлять все задания именно на них. Такое положение дел не является удивительным: влияние соотношения между числом серверов и их производительностью (а также и правилом обработки очереди) на эффективность диспетчеризации отмечалась в литературе и ранее (см., например, [527]).

[8] В классе стратегий, не использующих для принятия решений информацию о текущем (прошлом или будущем) состоянии системы, и моментах времени, в которые решения принимались.

Таблица 22 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметров стратегий приведены в таблице 23

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1.76 (1.76)	2.58 (2.63)	3.77 (3.87)	6.39 (6.56)	19.4 (20)
PROG-opt	1.76 (1.76)	2.41 (2.45)	3.22 (3.87)	5.17 (5.29)	15.0 (15.17)
AA	1.74 (1.76)	2.31 (2.35)	3.19 (3.87)	5.18 (5.29)	15.1 (15.81)

Таблица 23 — Оптимальные значения параметров стратегий из таблицы 22

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1	0.855	0.784	0.701	0.676
PROG-opt	1	0.7684	0.7076	0.6825	0.6734
AA	1.7	1.75	1.5	1.251	1.1

чем выше загрузка, тем больше выигрыш (в этом примере он достигает 20%). Вычислительные эксперименты показывают, что этот вывод остается справедливым и в самых общих предположениях об исходных параметрах системы. По-другому обстоит дело при сравнении нового алгоритма с оптимальной[9] диспетчеризацией PROG. Здесь новый алгоритм приводит к меньшим (в этом примере до 5%) значениям целевой функции при загрузке ниже средней, но с ростом загрузки начинает ей уступать. Оказывается, что какая система бы ни рассматривалась, такова типичная картина: при сравнении нового алгоритма с наилучшим из известных[10], выигрыш уменьшается с увеличением загрузки системы и, в итоге, становится отрицательным[11]. Это обстоятельство является ожидаемым следствием преимуществ нового алгоритма (его универсальности и простоты[12]), которые в полной мере раскрываются в следующих примерах.

[9] В указанном в сноске на предыдущей странице смысле.

[10] Но лишь при близко к оптимальному выбору значений его параметров!

[11] Этого, однако, не происходит при использовании алгоритмов из предыдущей главы: там выигрыш стремится к нулю, оставаясь все время положительным.

[12] И поэтому алгоритмы главы 3, которые реализуют идею диспетчеризации по предыстории действий и моментам поступления полноценным образом, доставляют наилучшие значения целевой функции во всем диапазоне загрузки.

Предположим теперь, что полностью марковская система состоит из $M > 2$ серверов различной производительности, суммарно равной единице. Для определенности положим $v^{(m)} = 2m/(M(M+1))$, $1 \leqslant m \leqslant M$. Таким образом, загрузка системы ρ, как и в предыдущем примере, совпадает с λ. При $M > 2$ оптимальная стратегия распределения заданий уже неизвестна. Поэтому в качестве программной стратегии будем использовать (8). Заметим, что стратегии RND и PROG зависят от $M-1$ параметров, а новая стратегия AA — от одного. В таблице 24 и 25 приведены значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе с $M = 64$ и $M = 128$ серверами при различных значениях загрузки. Значения параметров стратегий (см. таблица 26) были выбраны следующим образом. Для нахождения оптимальных значений параметров $(p_1, \ldots, p_M)$ стратегии RND при каждом значении ρ решалась задача минимизации (5). Для стратегии PROG было рассмотрено два случая: когда ее параметры $(d_1, \ldots, d_M)$ равны $(p_1, \ldots, p_M)$ (далее PROG-RND-opt) и когда $d_m = v^{(m)}$ (далее PROG-LB[13]). Значения параметра стратегии AA находились на имитируемых траекториях.

Таблица 24 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из 64 серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительность сервера m равна $v^{(m)} = m/(32 \times 65)$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметра стратегии AA приведены во второй строке таблице 26

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	46.2 (46)	68.3 (69)	103 (105)	182 (192)	565 (616)
PROG-RND-opt	38.5 (39)	48.6 (51)	65.7 (71)	105 (114)	300 (332)
PROG-LB	64 (127)	67.1 (130)	81.2 (158)	122 (247)	334 (608)
AA	37.2 (37)	48.2 (49)	67.2 (73)	111 (127)	330 (520)

Сопоставляя данные данные в таблицах 24 и 25, видим, что с увеличением числа серверов соотношения между стратегиями, установленные на примере двухсерверной системы, сохраняются. Новый алгоритм всегда лучше рандомизированной стратегии (в представленных примерах выигрыш достигает без

[13] LB от англ. load balancing, т. е. нагрузка балансируется диспетчером между серверами пропорционально их производительности.

Таблица 25 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из 128 серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительность сервера m равна $v^{(m)} = m/(64 \times 129)$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметра стратегии AA приведены в третьей строке таблицы 26

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	92.0 (92)	136.2 (137)	206 (212)	362 (387)	1126 (1225)
PROG-RND-opt	76.5 (78)	96.2 (100)	130 (138)	207 (226)	592 (656)
PROG-LB	128 (272)	134 (274)	162 (332)	242 (489)	665 (1304)
AA	73.7 (23)	95.6 (98)	133 (141)	219 (261)	652 (911)

Таблица 26 — Значения параметра стратегии AA из таблицы 24 и 25

ρ	0.1	0.3	0.5	0.7	0.9
$N = 64$	3.15	2.24	1.77	1.36	1.11
$N = 128$	3.4	2.3	1.72	1.38	1.05

малого 100%). В сравнении с программной стратегией он обнаруживает меньшие значения стационарного среднего (и стандартного отклонения) времени пребывания при загрузке ниже средней.

С точки зрения практики к обрисованной выше картине необходим следующий важный штрих. Премущество лучшей из ранее известных стратегий (PROG) над новым алгоритмом проявляется обычно только в области высокой загрузки и только когда значения ее параметров выбраны наилучшим образом. В отсутсвие возможности осуществить такой выбор, ее преимущество сходит на нет. Обратимся, например, к стратегии PROG-LB т. е. стратегии PROG, в которой доля заданий d_m, направляемых на сервер m, пропорциональна производительности сервера. Как видно из таблиц 24 и 25, при таком, заведомо неоптимальном, но порой единственно возможном, выборе значений параметров стратегии PROG, новый алгоритм оказываются наилучшим во всем диапазоне загрузки.

Перейдем к немарковскому случаю и ограничимся[14] системой с двумя серверами производительности $v^{(1)} = 4/5$ и $v^{(2)} = 1/5$, в которую поступают задания, средний размер которых равен единице. Тогда загрузка системы ρ совпадает с λ. В таблицах 27 и 29, приводятся значения стационарного среднего (и стандартного отклонения) времени пребывания, позволяющие оценить влияние характеристик входящего потока и распределения размера заданий на эффективность нового алгоритма. В качестве распределения $B(x)$ размера заданий выбраны

- вырожденное распределение, т. е. размер всех заданий равен единице,
- бимодальное распределение, сосредоточенное в точках 0.5 и 5 с вероятностями 8/9 и 1/9 соответственно.

В качестве распределения $F(x)$ входящего потока взяты

- равномерное распределение на интервале $[1{,}2\lambda^{-1}-1]$, т. е. среднее время между поступлениями заданий равно λ^{-1},
- экспоненциальное распределение с параметром λ,
- гиперэкспоненциальное распределение с параметрами $(a_1;\lambda_1{,}\lambda_2)$, т. е. среднее время между поступлениями заданий равно $a_1\lambda_1^{-1}+(1-a_1)\lambda_2^{-1} = \lambda^{-1}$,
- эрланговское распределение с параметрами $(k;\Lambda)$, т. е. среднее время между поступлениями заданий равно $k\Lambda^{-1} = \lambda^{-1}$.

Каждая из трех доступных диспетчеру стратегий зависит от одного параметра. Однако в сделанных предположениях уже даже для стратегии RND нет возможности точно вычислить его оптимальное значение. Указанные в таблицах значения — наилучшие из тех, что удалось найти на имитируемых траекториях. Как видно из таблицы 27 и 29, для немарковских систем с дисциплиной FIFO в однопроцессорных серверах установленные ранее соотношения между стратегиями не меняется. При условии, что есть возможность находить близкие к оптимальным значения параметров $p_1,\ldots,p_M$, новый алгоритм обнаруживает меньшие значения стационарного среднего (и стандартного отклонения) времени пребывания при загрузках ниже средней; в противном случае, он является наилучшим во всем диапазоне загрузки.

[14] Поскольку основной вывод, сделанный выше на примере марковской модели, остается справедливым и здесь.

Таблица 27 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в двухсерверной системе при различных входящих потоках и распределениях размера заданий. Производительности серверов: $v^{(1)} = 4/5$, $v^{(2)} = 1/5$. Загрузка системы $\rho = \lambda = 0.3$. Значения параметров стратегий приведены в таблице 28

		Распред. размера заданий	
		Вырожд. ($C_B = 0$)	Бимод. ($C_B = 2$)
Распределение входящего потока	Равном. ($C_F \approx 0.4$)	RND 1.295 (0.39) PROG 1.295 (0.39) AA 1.257 (0.04)	RND 0.675 (0.89) PROG 0.673 (0.87) AA 0.625 (≈ 0)
	Эксп. ($C_F = 1$)	RND 1.63 (0.67) PROG 1.63 (0.67) AA 1.62 (0.65)	RND 2.38 (3) PROG 2.38 (3) AA 2.33 (3)
	Гиперэксп. $(1/6; 0.1, 0.5)$ ($C_F \approx 1.61$)	RND 1.92 (1) PROG 1.92 (1) AA 1.87 (0.9)	RND 3.04 (4.2) PROG 2.9 (3.9) AA 2.83 (3.7)
	Эрлангл. $(50; 15)$ ($C_F \approx 0.14$)	RND 1.25 (≈ 0) PROG 1.25 (≈ 0) AA 1.25 (≈ 0)	RND 1.72 (2.1) PROG 1.72 (2.1) AA 1.72 (2.1)

Таблица 28 — Значения параметров стратегий из таблицы 27

		Распределение размера заданий	
		Вырожд.	Бимод.
Распределение вход. потока	Равном.	$p_1 = d_1 = 0.9899$, $\zeta = 1$	$p_1 = d_1 = 0.9889$, $\zeta = 1$
	Эксп.	$p_1 = d_1 = 1$, $\zeta = 1.2$	$p_1 = d_1 = 1$, $\zeta = 1.72$
	Гиперэксп.	$p_1 = d_1 = 1$, $\zeta = 1.24$	$p_1 = d_1 = 0.87$, $\zeta = 1.72$
	Эрлангл.	$p_1 = d_1 = \zeta = 1$	$p_1 = d_1 = \zeta = 1$

Таблица 29 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в двухсерверной системе при различных входящих потоках и распределениях размера заданий. Производительности серверов: $v^{(1)} = 4/5$, $v^{(2)} = 1/5$. Загрузка системы $\rho = \lambda = 0.5$. Значения параметров стратегий приведены в таблице 30

		Распределение размера заданий	
		Вырожд. ($C_B = 0$)	Бимод. ($C_B = 2$)
Распределение входящего потока	Равном. ($C_F \approx 0.29$)	RND 1.307 (0.38) PROG 1.307 (0.38) AA 1.269 (0.06)	RND 0.676 (0.91) PROG 0.673 (0.88) AA 0.625 (≈ 0)
	Эксп. ($C_F = 1$)	RND 2.30 (1.4) PROG 2.29 (1.4) AA 2.11 (1.14)	RND 4.1 (5.6) PROG 3.79 (4.8) AA 3.77 (4.8)
	Гиперэксп. $(0.9804; 0.9804, 0.0196)$ ($C_F \approx 2$)	RND 7.97 (8.2) PROG 6.07 (4.7) AA 6.06 (4.7)	RND 12.7 (15) PROG 11.6 (13.1) AA 11.6 (13.8)
	Эрлангл. $(50; 25)$ ($C_F = 0.14$)	RND 1.25 (≈ 0) PROG 1.25 (≈ 0) AA 1.25 (≈ 0)	RND 3.05 (4.1) PROG 2.89 (3.7) AA 2.9 (3.7)

Таблица 30 — Значения параметров стратегий из таблицы 29

		Распределение размера заданий	
		Вырожд.	Бимод.
Распределение вход. потока	Равном.	$p_1 = d_1 = 0.9899$, $\zeta = 1$	$p_1 = d_1 = 0.9889$, $\zeta = 1$
	Эксп.	$p_1 = d_1 = 0.972$, $\zeta = 1.3$	$p_1 = d_1 = 0.88$, $\zeta = 1.48$
	Гиперэксп.	$p_1 = d_1 = 0.8$, $\zeta = 2.5$	$p_1 = d_1 = 0.815$, $\zeta = 1.2$
	Эрлангл.	$p_1 = d_1 = \zeta = 1$	$p_1 = d_1 = 0.92$, $\zeta = 1.7375$

Аналогичным образом обстоит дело, если сменить дисциплину обслуживания во всех серверах с FIFO на любую из указанных в начале параграфа[15]. Для примера возьмем дисциплину SRPT, которая в однолинейных СМО, как известно, минимизирует стационарное средне время пребывания задания в системе (см., например, [528]). Ограничимся системой из двух серверов суммарной производительности 1, в которую поступает пуассоновский поток заданий, средний размер которых равен единице. Положим, как и в первом примере, $v^{(1)} = 2/3$ и $v^{(2)} = 1/3$. Тогда загрузка системы ρ совпадает с λ. В указанных условиях каждая из доступных диспетчеру стратегий (RND, PROG и AA), зависит от одного параметра. Таблица 31 содержит значения стационарного среднего (и стандартного отклонения) времени пребывания в системе при различных значениях загрузки, когда размер заданий имеет равномерное распределение на [0.5,1.5]. Значения тех же характеристики в случае экспоненциального распределенного размера заданий даны в таблице 33. При пуассоновском потоке оптимальные значения параметров $(p_1, \ldots, p_M)$ стратегии RND можно находить численно (см. таблицы 32 и 34), путем минимизации среднего времени пребывания задания в системе (см. [529; 530]) —

$$\sum_{m=1}^{M} p_m \int_0^{\infty} \frac{\int_0^x u\left(1 - B\left(uv^{(m)}\right)\right) du}{x^2\left(1 - \lambda p_m \int_0^x u dB\left(uv^{(m)}\right)\right)} dx,$$

при ограничениях $0 \leqslant p_m \lambda \mathsf{E}(S/v^{(m)}) < 1$ для каждого m. Для получения хороших оценок параметров стратегий PROG и AA, а также RND при непуассоновском входящем потоке, приходится привлекать имитационное моделирование.

Как видно из таблицы 31 и 33, варьируя значения единственного параметра новый алгоритм можно успешно адаптировать к принципиально[16] новым условиям: он всегда “выигрывет” у рандомизированной стратегии и может уступать наилучшей из ранее известных при оптимальном выборе значений параметров последней.

[15] Судя по вычислительным экспериментам, — на любую консервативную дисциплину.

[16] Если при дисциплине FIFO еще можно усмотреть связь между структурой алгоритма и динамикой времен пребывания заданий в серверах, то при любой другой дисципилне она теряется.

Таблица 31 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет равномерное распределение на [0.5,1.5]. Оптимальные значения параметров стратегий приведены в таблице 32

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1.63 (0.65)	2.07 (1.45)	2.73 (2.46)	3.89 (5.29)	8.94 (29)
PROG-opt	1.63 (0.65)	2.07 (1.46)	2.46 (1.8)	3.08 (3.29)	5.95 (17)
AA	1.623 (0.61)	1.92 (0.93)	2.3 (1.4)	2.98 (2.85)	5.84 (15)

Таблица 32 — Оптимальные значения параметров стратегий из таблицы 31

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1	0.9901	0.7838	0.7088	0.6761
PROG-opt	1	0.9901	0.7838	0.7088	0.6667
AA	1.15	1.15	1.24	1.18	1.12

Таблица 33 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из двух серверов при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительности серверов: $v^{(1)} = 2/3$, $v^{(2)} = 1/3$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Оптимальные значения параметров стратегий приведены в таблице 34

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1.627 (1.77)	2.03 (2.83)	2.64 (4.16)	3.6 (7.5)	6.92 (32)
PROG-opt	1.627 (1.77)	2.03 (2.83)	2.49 (3.74)	3.19 (6.24)	5.75 (24)
AA	1.627 (1.77)	1.97 (2.45)	2.41 (3.5)	3.17 (6.12)	5.77 (24)

Таблица 34 — Оптимальные значения параметров стратегий из таблицы 33

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	1	1	0.802	0.717	0.678
PROG-opt	1	1	0.802	0.717	0.673
AA	0.705	1.24	1.24	1.2	1.09

Дополнения

Модификации и обобщения. Предложенный алгоритм допускает различные модификации. Например, если рассматривать $z_{n+1}^{(m)}$ как незаконченную работу в сервере m во вспомогательном процессе в момент t_{n+1}, и вспомнить, что все задания имеют одинаковый размер ζ, то ничто не мешает заменить $z_{n+1}^{(m)}$ на число заданий в сервере m в момент t_{n+1}. Такой вариант алгоритма представляется более предпочтительным, чем исходный, например, в тех случаях, когда в серверах реализована дисциплина справедливого разделения процессора: здесь для поступающего задания важнее число заданий, с которым он "делит" процессорное время (хотя бы в момент своего поступления!), чем суммарная незаконченная работа в системе.

Поскольку новый алгоритм является параметрическим, его можно без изменений применять при более общих предположениях о входящем потоке[17] и структуре[18] системы, чем те, что сделаны в начале главы. Так, например, ординарный входящий поток можно заменить на групповой без внутренней структуры. Как показано на численных примерах в [300], вне зависимости от способа диспетчеризации[19], такое обощение не меняет установленные выше соотношения между стратегиями. При этом в качестве целевого функционала можно взять стационарное среднее время пребывания в системе как отдельного задания, так и группы целиком.

Новый алгоритм остается результативным, если серверы являются многопроцессорными. В отличие от ранее известных стратегий, в новой такое изменение структуры системы можно учесть. Для этого, при расчете $\tilde{z}_n^{(m)}$, необ-

[17] Можно отказаться от предположения о рекуррентности входящего потока и рассматривать коррелированные потоки [133]. Однако вместе с этим, для получения содержательных выводов, требуется и новые приемы анализа. В частности это связано с тем, что, как известно, наличие корреляции во входном потоке существенно ухудшает характеристики СМО (например, в таблице 22 при $\rho = 0.5$ и входном MAP–потоке, задаваемом матрицами $D_0 = \left(\begin{smallmatrix} -1.6991125 & 0 \\ 0.0005075 & -0.05512 \end{smallmatrix}\right)$ и $D_1 = \left(\begin{smallmatrix} 1.681415 & 0.0176975 \\ 0.0060675 & 0.048545 \end{smallmatrix}\right)$, стандартное отклонение времени пребывания задания в системе увеличивается более, чем в 10 раз).

[18] Открытым остается вопрос об эффективности новой методики в случае, когда (все или некоторые) очереди в серверах имеют конечную емкость.

[19] Т. е. либо группой целиком, либо каждое задание по отдельности.

ходимо заменить[20] рекурсию Линдли рекурсией Кифера–Вольфовица. За счет варьирования значений параметров, как ранее известные, так и новую стратегии можно приспособить к случаю, когда серверы не являются абсолютно надежными. Например, если каждый сервер в случайные моменты времени независимо от остальных отключается и остается недоступным случайное время, причем распределения $U(x)$ и $D(x)$ периодов доступности и недоступности одинаковы для всех серверов, то соотношения между стратегиями остается прежним (см. таблице 7 в [300], когда серверы работоспособны 90% времени). И опять же, в новом алгоритме, в отличие от ранее известных, можно реализовать эту особенность системы: последовательно и независимо порождая значения случайных величин с распределениями $U(x)$ и $D(x)$, находятся периоды доступности и недоступности каждого вспомогательного процесса, а пересчет значений $\tilde{z}_n^{(m)}$ приостанавливается в течение периода недоступности процесса m. Из сказанного следует, что, при наличии точной информации о доступности серверов, ее можно учесть в новом алгоритме. Однако для рандомизированной и программной статегий информация такого рода оказывается бесполезной. Как следствие, судя по вычислительным экспериментам, в таких условиях (по-прежнему частичной наблюдаемости!) наилучая из ранее известных стратегий даже при оптимальных значениях парамеров уступает новому алгоритму во всем диапазоне загрузки. Наконец заметим, что предложенный подход к диспетчеризации через вспомогательные процессы позволяет ослабить предположение о том, что распределения $U(x)$ и $D(x)$ (и, вообще говоря, $B(x)$) известны точно. Предполагая, что определяющие параметры этих распределений являются случайными величинами (с известными распределениями), для получения нового алгоритма достаточно добавить порождение их значений в описанную выше схему работы вспомогательных процессов.

Наконец, примечательным является поведение нового алгоритма при ослаблении ограничений на доступность наблюдений за системой: его эффективность возрастает на десятки процентов. Вместе с тем эффективность ранее известных стратегий для частично наблюдаемых систем не меняется, т. к. использовать в них новую информацию невозможно. Вернемся к рассмотренной выше марковской системе из $M = 128$ однопроцессорных серверов

[20]При этом вычислительная сложность возрастает, что связано, главным образом, с необходимостью упорядочения в момент каждого поступления M наборов чисел. Размер набора m, $1 \leqslant m \leqslant M$, равен числу процессоров в сервере m.

производительностей $v^{(m)} = 2m/(M(M+1))$, $1 \leqslant m \leqslant M$. Сменим дисциплину обслуживания с FIFO на PS и предположим, что диспетчеру в моменты принятия решений доступна информация о текущих числах заданий в серверах. В этих новых условиях у него на выбор есть (по крайней мере[21]) четыре стратегии: RND, PROG, AA[22] и JSQ[23]. В таблице 35 приведены значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе при различных значениях загрузки. Значения параметров стратегий (см. таблица 36) были выбраны тем же способом, что и в случае, описанном таблице 25.

Таблица 35 — Значения стационарного среднего (и стандартного отклонения) времени пребывания задания в системе из 128 серверов с дисциплиной PS при различных стратегиях диспетчеризации и различной загрузке $\rho = \lambda$. Производительность сервера m равна $v^{(m)} = m/(64 \times 129)$. Входящий поток — пуассоновский с интенсивностью λ, размер заданий имеет экспоненциальное распределение со средним 1. Значения параметра стратегии AA приведены в таблице 36

ρ	0.1	0.3	0.5	0.7	0.9
RND-opt	92.0 (100)	136.2 (164)	206 (274)	362 (548)	1130 (1949)
PROG-RND-opt	76.4 (79)	96.3 (110)	130 (170)	207 (305)	595 (1000)
AA	66.7 (67.1)	71.2 (71.4)	77.4 (78.8)	90.4 (94.3)	164 (170)
JSQ	66.7 (67.1)	71.2 (71.4)	77.2 (78.8)	86.8 (91.7)	117 (238)

Таблица 36 — Значения параметра стратегии AA из таблицы 35

ρ	0.1	0.3	0.5	0.7	0.9
AA	0.25	0.2	0.15	0.1	0.05

Как видно по таблице 35, с помощью некоторой информации о текущем состоянии системы (в моменты поступлений) и нового алгоритма (но не ранее известных) фактически удается приблизиться к результатам стратегий для наблюдаемых систем.

Оценка параметров. Новый алгоритм зависит от одного единственного параметра (ζ). Вычислительные эксперименты показывают, что в каждой конкретной системе существует лишь единственное оптимальное значение ζ,

[21] Т. к. известны и экзотические стратегии, как, например, HJSQ(d); см. [174].

[22] В начале параграфа указано, каким образом следует модифицировать *Алгоритм IV*.

[23] Напомним, что так в диссертации кодируется диспетчеризация по наикратчайшей очереди.

т. е. такое значение при котором достигается глобальный минимум стационарного среднего времени пребывания задания в системе. Однако формулу для вычисления ζ получить не удается. Поэтому оптимизационную задачу приходится каждый раз решать, во-первых, приближенно (т.к. множество значений ζ несчетно), и, во-вторых, с помощью имитационного моделирования, сложность которого зависит от сложности расчета времени пребывания в системе поступающего задания. Когда в серверах используется дисциплина FIFO, ситуация облегчается тем обстоятельством, что рекурсия Линдли позволяет при моделировании легко рассчитывать время пребывания в системе n-го задания, распределенного на сервер m, при любой из рассматриваемых стратегий. Если обозначить время пребывания в системе n-го задания при стратегии RND (или PROG) и новой стратегии соответственно V_n и $\tilde{V}_n$, то для выбора "направления движения" при поиске наилучшего значения ζ может служить знак суммы $N^{-1}\sum_{n=1}^{N}\left(\tilde{V}_n - V_n\right)$ при достаточно большом N. Также можно поступить, если в системе имеются многопроцессорные сервера: для них рекурсия Линдли заменяется рекурсией Кифера–Вольфовица. Однако никаких выражений уже нельзя предложить в случае использования в серверах таких дисциплин обслуживания, как RANDOM, SJF, PS, PLIFO, FB, PSJF или SPRT. Здесь использование имитационного моделирования для нахождения ζ выглядит неизбежным.

В случае произвольного входящего потока расчет оптимальных значений целевой функции как при рандомизированной, так и при программной стратегии даже для однопроцессорных серверов связан с серьезными трудностями. Как уже отмечалось во Введении, для RND задача нахождения оптимального набора $(p_1, \ldots, p_M)$ из M чисел — вероятностей p_m выбора для очередного задания сервера m, в редких случаях может быть решена в явном виде. В общем же случае приходится либо использовать какой-то из инженерных подходов (например, балансировать нагрузку[24]), либо аппроксимировать входящий поток с помощью рекуррентного и использовать при решении оптимизационной задачи приближенные формулы[25] для стационарного среднего времени пребывания (например, известную формулу Крамера–Лангенбах-Бельца (см. [537] или [416,

[24] Т. е. положить $p_m = v^{(m)} / \sum_{m=1}^{M} v^{(m)}$.

[25] По проведенным (немногочисленным) вычислительным экспериментам с некоторыми известными в литературе результатами [531–533], для многопроцессорных серверов наблюдается низкое качество решений. Однако было бы преждевременным заявить о полной непригодности такого подхода: по вопросам аппроксимаций для систем и сетей массового обслуживания имеется обширная

С. 118])), либо осуществлять оптимизацию на имитируемых траекториях. Для рассматриваемых в диссертации задач последний подход позволяет рассчитывать на приемлемое качество решений.

В случае программной стратегии вообще не известен метод нахождения оптимальной последовательности действий. Наилучшие из встречающихся в научной литературе алгоритмов порождения ее элементов зависят от $M-1$ параметров — вероятностей $(d_1,\ldots,d_M)$. Обычным и достаточно хорошим образом действия является замена набора $(d_1,\ldots,d_M)$ оптимальным для стратегии RND набором $(p_1,\ldots,p_M)$. При этом, конечно, оптимальность программной стратегии не гарантируется.

Теоретическое обоснование. Остается практически непонятной теоретическая основа продуктивности столь простого в реализации, но интуитивно совершенно не очевидного алгоритма. Некоторые эффекты удалось обнаружить в процессе моделирования. Например, анализ численных экспериментов показал, что, по-видимому, преимущество нового алгоритма получается за счет более активного использования медленных серверов. Так, для случая двух серверов (см. таблица 22) и загрузки 0.1 новый алгоритм предписывает посылать на медленный сервер почти 10% заданий, тогда как (оптимальная!) программная стратегия вообще не использует медленный сервер. Другой, более тонкий эффект связан с принципиальной возможностью получения выигрыша за счет применения нового алгоритма. Например, при низкой загрузке в случае двух серверов с дисциплиной FIFO и пуассоновского входящего потока, выигрыш возможен тогда и только тогда, когда производительность быстрого сервера не превосходит $2(v^{(1)}+v^{(2)})/3$. Для общего случая подобного соотношения получить не удалось. В итоге, на основании вычислительных экспериментов можно сделать лишь довольно расплывчатый и требующий дальнейших уточнений вывод о том, что получение выигрыша возможно лишь когда производительности серверов несильно отличаются друг от друга.

Покажем, что малая загрузка системы является одним из благоприятных условий для получения теоретических обоснований нового алгоритма. Пусть в систему из двух серверов с дисциплиной FIFO поступает ординарный пуассоновский поток заданий интенсивности λ, причем второй момент $\mathsf{E}S^2$ размера заданий конечен. Предположим, что $\lambda\to 0$ т. е. обычно поступающему заданию

литература (например, [116;517;534;535] и [536, Section 6.3]) и внесение ясности в этот вопрос представляется предметом отдельных исследований.

не приходится ожидать в очереди. Покажем, что при выполнении этих условий существует такое $\zeta > 0$, что стационарное среднее $\mathsf{E}V$ время пребывания задания в системе при новом алгоритме меньше, чем при стратегиях RND и PROG с оптимально выбранными значениями параметров.

Занумеруем серверы в порядке убывания производительности. При рандомизированной стратегии пуассоновский входящий поток просеивается независимо в соответствии с набором вероятностей $(p_1,p_2 = 1-p_1)$ и, таким образом, на m-й сервер поступает пуассоновский поток интенсивности λp_m. Известно[26], что существует $\lambda^* > 0$ такое, что при $\lambda < \lambda^*$, решение задачи минимизации (5) есть набор (1,0), предписывающий отправлять каждое задание на сервер с максимальной производительностью. Кроме того, при $\lambda < \lambda^*$ стратегия PROG тождественна RND[27].

В силу дисциплины FIFO в каждом сервере время пребывания любого задания складывается из двух независимых компонент: время ожидания в очереди и время обслуживания. В теории массового обслуживания известен[28] следующий результат: для СМО $M\,|\,GI\,|\,1\,|\,\infty\,|\,\mathsf{FIFO}$ в условиях малой загрузки стационарное время ожидания W примерно[29] совпадает с $\max(0,S-T)$, где T — интервал между двумя последовательными поступлениями, причем

$$\lim_{\lambda\to 0}\frac{\mathsf{E}W}{\mathsf{E}(\max(0,S-T))} = 1. \tag{4.1}$$

Таким образом, время ожидания в очереди (в указанной выше системе) в стационарном режиме при малой загрузке есть (без малого) время ожидания в очереди задания, поступающего следом за заданием, поступившим в пустую систему.

[26] См., например, раздел 2.2. в [149].

[27] В указанных условиях при стратегии RND легко выписывается явная формула для $\mathsf{E}V$. Однако при стратегии AA получить явную формулу для $\mathsf{E}V$ не удается. Хотя каждый сервер и остается однолинейной СМО, но входящий в нее поток получается путем просеивания пуассоновского потока с вероятностью, зависящей от времени. Несмотря на наличие в этом направлении некоторых аналитических результатов (см., например, [538;539] и ссылки в них), пригодные для расчета и/или анализа формулы для $\mathsf{E}V$ (даже в условиях малой загрузки!) найти не удается. Отметим еще одно обстоятельство. Поскольку каждое задание во вспомогательном процессе имеет одинаковый размер ζ, на исходную задачу можно посмотреть и по-другому — как на задачу о наиболее плотном размещении (разнотипных) интервалов фиксированной длины, "поступающих" в случайные моменты времени (см. [540–544] и [545, С. 42–46]).

[28] См. доказательство в [546, Corollary 3.2] и [547].

[29] В метрике полной вариации при $\lambda \to 0$, а более точно — в смысле определения Definition 2.1 в [546].

Рассмотрим в указанных выше предположениях две параллельно работающие системы: одна со стратегий RND, другая — с новой стратегией AA. Пусть обе системы находятся в стационарном режиме и в некоторый момент (момент 0), когда они обе свободны от заданий, приходит первое задание. В системе с новой стратегией это задание всегда поступает в сервер 1. Также происходит и в системе со стратегией RND. В обеих системах первое задание будет обслуживаться среднее время $\mathsf{E}S/v^{(1)}$.

Пусть второе задание поступило через T_2 единиц времени. В системе со стратегией RND оно (как и все последующие) будет отправлено в сервер 1. В системе с новой стратегией другое решение (т.е. сервер 2) принимается, если

$$\max\left(0,\frac{\zeta}{v^{(1)}}-T_2\right)+\frac{\zeta}{v^{(1)}}>\frac{\zeta}{v^{(2)}},$$

что эквивалентно $T_2<\max(0,\frac{2\zeta}{v^{(1)}}-\frac{\zeta}{v^{(2)}})$. Иначе второе задание будет отправлено в сервер 1. Таким образом, если $\frac{2}{v^{(1)}}\leqslant\frac{1}{v^{(2)}}$ система с новой стратегией идентична системе со стратегией RND. Поэтому далее предположим, что $\frac{2}{v^{(1)}}>\frac{1}{v^{(2)}}$ и для сокращения записи введем обозначение $v^*=\frac{2}{v^{(1)}}-\frac{1}{v^{(2)}}$.

Разберемся в том, как различаются средние времена пребывания каждого задания в системе со стратегией RND и в системе с новой стратегией. Для первого задания средние времена совпадают. Среднее время пребывания второго задания в системе с новой стратегией равно

$$\underbrace{\int_{\zeta v^*}^{\infty}\lambda e^{-\lambda x}\mathsf{E}\left(\max\left(0,\frac{S_1}{v^{(1)}}-x\right)+\frac{S_2}{v^{(1)}}\right)dx}_{\text{сервер 1}}+\underbrace{\int_0^{\zeta v^*}\lambda e^{-\lambda x}\mathsf{E}\left(\frac{S_2}{v^{(2)}}\right)dx}_{\text{сервер 2}},\qquad(4.2)$$

а в системе со стратегией RND:

$$\int_0^{\infty}\lambda e^{-\lambda x}\mathsf{E}\left(\max\left(0,\frac{S_1}{v^{(1)}}-x\right)+\frac{S_2}{v^{(1)}}\right)dx.\qquad(4.3)$$

Вычитая (4.3) из (4.2), получаем:

$$\int_0^{\zeta v^*}\lambda e^{-\lambda x}\left(\underbrace{\frac{\mathsf{E}S}{v^{(2)}}-\frac{\mathsf{E}S}{v^{(1)}}-\mathsf{E}\left(\max\left(0,\frac{S}{v^{(1)}}-x\right)\right)}_{=f(x)}\right)dx.\qquad(4.4)$$

Функция f является непрерывной, причем

$$f(0) = \left(\frac{1}{v^{(2)}} - \frac{2}{v^{(1)}}\right) \mathsf{E}S < 0,$$
$$\lim_{x\to\infty} f(x) = \left(\frac{1}{v^{(2)}} - \frac{1}{v^{(1)}}\right) \mathsf{E}S > 0.$$

Кроме того, $f'(x) > 0$, $x \geqslant 0$, т.е. f — возрастающая функция. Поэтому существует такое $\zeta > 0$, что интеграл будет отрицательным и, таким образом, среднее время пребывания второго задания в системе с новой стратегией будет меньше, чем в системе со стратегией RND.

Аналогичные рассуждения для последующих заданий показывают, что среднее время ожидания задания в системе с новой стратегией либо совпадает с RND, либо меньше него. Проведем рассуждения для третьего задания. Пусть время между приходом третьего и второго задания равно T_3. В системе со стратегией RND оно будет отправлено в сервер 1. В системе с новой стратегией ситуация сложнее. Третье задание будет отправлено в другую очередь (т.е. сервер 2) в двух случаях: если при $T_2 \geqslant \zeta v^*$ выполняется неравенство

$$\max\left(0, \max\left(0, \frac{\zeta}{v^{(1)}} - T_2\right) + \frac{\zeta}{v^{(1)}} - T_3\right) + \frac{\zeta}{v^{(1)}} > \frac{\zeta}{v^{(2)}}, \tag{4.5}$$

или если при $T_2 < \zeta v^*$ выполняется неравенство

$$\max\left(0, \max\left(0, \frac{\zeta}{v^{(1)}} - T_2\right) - T_3\right) + \frac{\zeta}{v^{(1)}} > \max\left(0, \frac{\zeta}{v^{(2)}} - T_3\right) + \frac{\zeta}{v^{(2)}}. \tag{4.6}$$

Однако при $T_2 < \zeta v^*$ неравенство (4.6) никогда не выполняется. Поэтому остается только первый случай, согласно которому первое и второе задания направляются в сервер 1, а третье — в сервер 2. Разность[30] между средним временем пребывания третьего задания в системе с новой стратегий и со стратегией RND (при условии, что второе и третье задания поступили соответственно в очерез x и y единиц времени) равна

$$f_2(x,y) = \frac{\mathsf{E}S}{v^{(2)}} - \frac{\mathsf{E}S}{v^{(1)}} - \mathsf{E}\max\left(0, \max\left(0, \frac{S_1}{v^{(1)}} - x\right) + \frac{S_2}{v^{(1)}} - y\right),$$

[30] Здесь не учтен сучай, когда второе задание было отправлено в сервер 2, а третье — в сервер 1. Но в этом случае среднее время пребывания третьего задания в системе с новой стратегией будет меньше, чем в системе со стратегией RND .

а безусловная вероятность, с учетом (4.5), есть

$$\int_{\zeta v^*}^{\frac{\zeta}{v^{(1)}}} \lambda e^{-\lambda x} dx \int_0^{\zeta v^* + \frac{\zeta}{v^{(1)}} - x} \lambda e^{-\lambda y} f_2(x,y) dy + \\ + \int_{\frac{\zeta}{v^{(1)}}}^{\infty} \lambda e^{-\lambda x} dx \int_0^{\zeta v^*} \lambda e^{-\lambda y} f_2(x,y) dy. \quad (4.7)$$

Осталось показать это, если ζ выбрано таким образом, что интеграл в (4.4) отрицателен, то и сумма (4.7) будет отрицательной (или равна нулю). Для этого заметим, что

$$f_2(x,y) = \underbrace{\frac{\mathsf{E}S}{v^{(2)}} - \frac{\mathsf{E}S}{v^{(1)}} - \mathsf{E}\left(\max\left(0, \frac{S_2}{v^{(1)}} - y\right)\right)}_{=f(y)} + \\ + \mathsf{E}\left(\max\left(0, \frac{S_2}{v^{(1)}} - y\right)\right) - \mathsf{E}\max\left(0, \frac{S_2}{v^{(1)}} - y, \frac{S_1}{v^{(1)}} + \frac{S_2}{v^{(1)}} - y - x\right). \quad (4.8)$$

Поскольку ζ в (4.4) выбрано таким образом, что $f(y) < 0$ при $0 < y < \zeta v^*$, а разность во второй строке (4.8) по крайней мере неположительна, то второй интеграл в (4.7) отрицателен и имеет первый порядок малости (относительно λ). Первый интеграл в (4.7) может оказаться положительным, но имеет второй порядок малости[31].

В заключение отметим, что предположение о пуассоновости входящего потока является существенным в проведенных рассуждениях. Также оно является существенным и для выполнения соотношения (4.1). Например, как показано в [547, Раздел 3] для детерминированного входного потока и $B(x) = 1 - (x+1)^{-3}$, $x \geqslant 0$, (4.1) не выполняется. Другие контрпримеры можно найти в [546, Remark 3.2] и [547, Раздел 3]. Более широкий класс потоков, для которого выполняется (4.1), описан в [546, Corollary 3.4]. В этом случае для обоснования существования ζ, даже в условиях малой загрузки, необходимо искать другой путь.

---

[31] Действительно,

$$\int_{\zeta v^*}^{\frac{\zeta}{v^{(1)}}} \lambda e^{-\lambda x} dx \int_0^{v^* + \frac{\zeta}{v^{(1)}} - x} \lambda e^{-\lambda y} dy = e^{-\lambda \zeta v^*} - e^{-\lambda \frac{\zeta}{v^{(1)}}} \left(1 - \lambda e^{-\lambda v^*} \left(\frac{\zeta}{v^{(2)}} - \frac{\zeta}{v^{(1)}}\right)\right) \approx \\ \approx \lambda^2 \left(\frac{\zeta}{v^{(1)}} + v^*\right)\left(\frac{\zeta}{v^{(2)}} - \frac{\zeta}{v^{(1)}}\right) + o(\lambda^2).$$

Заключение

Тематика этой диссертационной работы относится к развивающемуся научному направлению — исследованиям стохастическим систем обслуживания с частичной наблюдаемостью (или, что то же, — с неполным наблюдением, с неполным информационным описанием и т. п.). Словосочетание “частичная наблюдаемость” может трактоваться весьма широко. Это, например, отсутствие априорной информации (даже на уровне представления) о структуре объекта, и/или ограниченная возможность наблюдения объекта и его идентификации и т. д. Таким образом, в это направление “укладываются” многие задачи, возникающие в современных информационных, телекоммуникационны, вычислительных и других технических системах (см., например, [12–19;42;70;72;548]).

Если от системы в процессе функционирования поступает какая-либо информация, то при решении проблем, связанных по крайней мере с системным анализом, преимущество целесообразно отдать адаптивным стратегиям обработки информации. В отсутствие же обратной связи первостепенное значение приобретает умение воспользоваться доступной априрной информацией о системе; прямое применение адаптивных приемов здесь невозможно.

Решенная в диссертационной работе проблема — разработка комплекса вероятностных моделей и создание на их основе методов анализа и алгоритмов управления для стохастических систем обслуживания с частичной наблюдаемостью — относится к проблемам последнего рода. Если на основе результатов, изложенных диссертации, подвести итог всему исследованию, то можно сказать, что его основные результаты заключаются в следующем.

1. Развит аналитический аппарат анализа стационарных характеристик ранее не изучавшихся классов СМО инверсионного типа, допускающих не сохраняющее работу обслуживание. Расширена область применения известной методики [98], в соответствии с которой с помощью (определенным образом вводимой) совокупности вспомогательных СМО получаются рекуррентные процедуры вычисления стационарных характеристик исходных систем.
2. Введена и систематически изучена новая дисциплина — инверсионный порядок обслуживания с обобщенным вероятностным приоритетом.

При надлежащем выборе параметров введенная новая дисциплина превращается в распространенные на практике и наиболее изученные.

3. Область применения СМО инверсионного типа расширена на класс задач, связанных с получением оценок фактических значений стационарных вероятностно–временных характеристик изолированно функционирующих стохастических систем обслуживания с частичной наблюдаемостью.
4. Разработан метод получения оценок (фактических) значений стационарных вероятностно–временных характеристик изолированно функционирующих систем, частичная наблюдаемость которых связана с тем, что используемые для управления очередями времена обслуживания могут не совпадать с (ненаблюдаемыми) фактическими. Сформулированы условия и получено доказательство эффективности предложенного метода для ряда систем, моделируемых немарковскими системами массового обслуживания с пуассоновскими входящими потоками.
5. Изучен класс стохастических систем с параллельным обслуживанием, диспетчеризация в которых осуществляется в таких условиях частичного наблюдения, что исключают возможность прямого применения методов теории адаптации. Проведена систематизация известных в научной литературе результатов этой области.
6. Разработаны новые методы диспетчеризации для широкого класса частично наблюдаемых стохастических систем с параллельным обслуживанием, основанные на общей идее — использовании при управлении входящими потоками полной предыстории наблюдаемых компонент. Основанные на них алгоритмы превосходят (по крайней мере по наиболее популярным критериям) ранее известные в научной литературе.
7. Предложена новая простая и эффективная конструкция стратегии управления входящими потоками в системах с параллельным обслуживанием при отсутствии информации об их динамическом состоянии, основанная на использовании виртуальных вспомогательных процессов.

В связи с рассмотренными в диссертационной работе задачами остаются вопросы, поиск ответов на которые представляется интересным направлением дальнейших исследований. Не останавливаясь повторно на тех, что уже были

освещены в тексте, скажем несколько слов о других, носящих принципиальный характер. Метод получения оценкок (фактических) значений характеристик частично наблюдаемых систем, предложенный в главе 2, приводит к содержательным результатам в стационарном случае. Разработка аналитических основ, расширяющих его область применения на системы с периодическими потоками и на переходный режим функционирования, является новой и на данный момент нерешенной задачей. Кроме того, интересным является вопрос о возможности обобщения полученных результатов на частично наблюдаемые СеМО, системы конечной емкости, а также на системы, двойственные[32] тем, что рассмотрены в главе 2. В связи с предложенными в главах 3 и 4 решениями задачи диспетчеризации, наиболее интересные вопросы связаны с новыми "неточными" стратегиями. В частности, стратегии главы 4, даже в самом простом виде, остаются слишком сложными для анализа. Здесь еще только предстоит найти подходящий "угол атаки" и сформированить теоретическую основу продуктивности предложенной конструкции.

[32] Т. е. в которых прогнозные интервалы между поступлениями заявок могут не совпадать с фактическими, а времена обслуживания известны в точности.

Список сокращений и условных обозначений

LWL	least work left
JSQ	join the shortest queue
HJSQ(d)	hybrid JSQ with d choices
AA	arrival aware
RND	random
PAP	probabilistic allocation policy
BS	bernoulli splitting
TP	threshold policy
FPI	first policy iteration
FIFO	first–in–first–out
LIFO	last–in–first–out
PS	processor–sharing
PLIFO	preemptive last-in-first-out
SJF	(non-preemptive) shortest job first,
PSJF	preemptive shortest job first
FB	foregroung-backgroud processor–sharing
SRPT	shortest remaining processing time
LIFO GPP	last–in–first–out with generalized probabilistic priority
LIFO Re	last–in–first–out with resampling
LIFO PRD	last–in–first–out preemptive different
SMART	small response time
FSP	fair scheduling policy
УФИ	убывающая функция интенсивности
ГНСХИ	гармоничное новое в среднем хуже использованного
СМО	система массового обслуживания
СеМО	сеть массового обслуживания
ТМО	теория массового обслуживания
ПЛС	преобразование Лапласа–Стилтьеса
ПЛ	преобразование Лапласа
ПФ	производящая функция
сл. в.	случайная величина
ф. р.	функция распределения

Список литературы

1. *Коновалов М.Г.* Модели и технологии адаптивной обработки информации для частично наблюдаемых систем // *Докт. диссертация.* — 2007. — 345 с.

2. *Sragovich V.G.* Mathematical theory of adaptive control. — Singapore: World Scientific, 2006. — 492 pp.

3. *Назин А.В., Позняк А.С.* Адаптивный выбор вариантов: Рекуррентные алгоритмы. — Москва: Наука. Гл. ред. физ.-мат. лит., 1986. — 288 с.

4. *Коновалов М.Г.* Методы адаптивной обработки информации и их приложения. — Москва: ИПИ РАН, 2007. — 212 с.

5. *Cao X.R.* Stochastic learning and optimization: A sensitivity–based approach. — Springer, 2007. — 566 pp.

6. *Bertsekas D.P.* Dynamic programming and optimal control. — 4 edition. — Belmont, MA, USA: Athena Scientific, 2012. — 1270 pp.

7. *Sutton R., Barto A.* Reinforcement learning. — 2 edition. — Cambridge, Massachusetts; London, England: MIT Press, 2018. — 552 pp.

8. *Elliot R., Aggoun L., Moore J.* Hidden Markov models. — Springer, 2008. — 374 pp.

9. *Poznyak A.S., Najim K., Gomez-Ramirez E.* Self–learning control of finite Markov chains. — CRC Press, 2000. — 316 pp.

10. *Колногоров А.В.* Управление в случайной среде. — Великий Новгород: Новгородский государственный университет имени Ярослава Мудрого, 2013. — 87 с.

11. *Пугачев В.С., Синицин И.Н.* Теория стохастических систем. — М.: Логос, 2004. — 1000 с.

12. *Chen Y., Hasenbein J.* Staffing large–scale service systems with distributional uncertainty // *Queueing Systems.* — 2017. — Vol. 87, no. 1. — Pp. 55–79.

13. *Cohen A., Saha S.* Asymptotic optimality of the generalized $c\mu$ rule under model uncertainty // *Stochastic Processes and Their Applications.* — 2021. — Vol. 136. — Pp. 206–236.

14. *Economou A.* The impact of information structure on strategic behaviour in queueing systems // *Queueing theory 2: Advanced trends.* — 2020. — Pp. 137–169.

15. Statistical inference for $M_t/G/\infty$ queueing systems under incomplete observations / D. Li, Q. Hu, L. Wang, D. Yu // *European Journal of Operational Research.* — 2019. — Vol. 279. — Pp. 882–901.

16. *Cao P., Zhong Z., Huang J.* Dynamic routing in a distributed parallel many–server service system: The effect of ξ–choice // *European Journal of Operational Research.* — 2021. — Vol. 294, no. 1. — Pp. 219–235.

17. Задача оптимального стохастического управления потоком данных по неполной информации / Б.М. Миллер, К.Е. Авраченков, К.В. Степанян, Г.Б. Миллер // *Проблемы передачи информ.* — 2005. — Т. 41, № 2. — С. 89–110.

18. *Robbins T., Medeiros D., Dum P.* Evaluating arrival rate uncertainty in call centers // *Proceedings of the 38th Winter Simulation Conference.* — 2006. — Pp. 2180–2187.

19. *Ben-Tal A., Nemirovsky A.* Robust solutions of linear programming problems contaminated with incertain data // *Mathematical Programming.* — 2000. — Vol. 88, no. 3. — Pp. 411–424.

20. *Ушаков В. Г., Ушаков Н.Г.* Декомпозиция при частично известном распределении ошибки // *Информ. и её примен.* — 2011. — Т. 5, № 4. — С. 36–39.

21. *Miller G.K., Bhat U.N.* Estimation for renewal processes with unobservable gamma or Erlang interarrival times // *Journal of Statistical Planning and Inference.* — 1997. — Vol. 61. — Pp. 355–372.

22. Инженерные методы расчета сетей при проектировании распределенных автоматизированных систем массового обслуживания / Р.В. Билик, З.П. Мясоедова, Н.В. Петухова, М.П. Фархадов. — Москва: Макс Пресс, 2010. — 256 с.

23. *Mitzenmacher M.* Scheduling with predictions and the price of misprediction // *Proceedings of the 11th Innovations in Theoretical Computer Science Conference.* — 2020. — Pp. 1–18.

24. Non–clairvoyant scheduling with predictions / S. Im, R. Kumar, M. Qaem, M. Purohit // *Proceedings of the 33rd ACM Symposium on Parallelism in Algorithms and Architectures.* — 2021. — Pp. 285–294.

25. *Соколов И.А.* Теория и практика применения методов искусственного интеллекта // *Вестник РАН.* — 2019. — Т. 89, № 4. — С. 365–370.

26. *Haenlein M., Kaplan A. A.* Brief history of artificial intelligence: On the past, present, and future of artificial intelligence // *Calif. Manage. Rev.* — 2019. — Т. 61, № 4. — С. 5–14.

27. *Борисов А. В., Босов А. В., Жуков Д. В.* Стратегия исследований и разработок в области искусственного интеллекта I: Основные понятия и краткая хронология // *Системы и средства информ.* — 2021. — Т. 31, № 1. — С. 57–68.

28. *Попков Ю.С.* Процедуры рандомизированного машинного обучения // *Автомат. и телемех.* — 2019. — Т. 9. — С. 122–142.

29. Are user runtime estimates inherently inaccurate? / C. Lee, S. Schwartzman, J. Hardy, A. Snavely // *Lecture Notes in Computer Science.* — 2004. — Vol. 3277. — Pp. 253–263.

30. *Tsafrir D.* Using inaccurate estimates accurately // *Lecture Notes in Computer Science.* — 2010. — no. 6253. — Pp. 208–221.

31. *Lu D., Sheng H., Dinda P.* Size–based scheduling policies with inaccurate scheduling information // *Proceedings of the 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems.* — 2004. — Pp. 31–38.

32. *Rawat M., Kshemkalyani A.* SWIFT: Scheduling in web servers for fast response time // *Second IEEE International Symposium on Network Computing and Applications.* — 2003. — Pp. 51–58.

33. Size–based scheduling to improve web performance / M. Harchol-Balter, B. Schroeder, N. Bansal, M. Agrawal // *ACM Trans. Comput. Syst.* — 2003. — Vol. 21, no. 2. — Pp. 207–233.

34. Effects and implications of file size/service time correlation on web server scheduling policies / D. Dong Lu, P. Dinda, Y. Qiao, H. Sheng // *Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems.* — 2005. — Vol. 1. — Pp. 258–267.

35. Looking at the server side of peer–to–peer systems / Y. Qiao, D. Lu, R. Bustamante, P. Dinda // *Proceedings of the 7th workshop on Workshop on languages, compilers, and run–time support for scalable systems.* — 2004. — Pp. 1–8.

36. Scheduling in MapReduce–like systems for fast completion time / H. Chang, M. Kodialam, R.R. Kompella et al. // *Proceedings of the 30th IEEE International Conference on Computer Communications.* — 2011. — Pp. 3074–3082.

37. Re–optimizing data–parallel computing / S. Agarwal, S. Kandula, N. Bruno et al. // *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation.* — 2012. — Pp. 1–21.

38. Same queries, different data: Can we predict runtime performance? / A. D. Popescu, V. Ercegovac, A. Balmin et al. // *Proceedings of the 28th International Conference on Data Engineering Workshops.* — 2012. — Pp. 275–280.

39. *Verma A., Cherkasova L., Campbell R. H.* ARIA: Automatic resource inference and allocation for MapReduce environments // *Proceedings of the 8th ACM International Conference on Autonomic Computing.* — 2011. — Pp. 235–244.

40. *Lipton R. J., Naughton J. F.* Query size estimation by adaptive sampling // *J. Comput. Syst. Sci.* — 1995. — Vol. 51, no. 1. — Pp. 18–25.

41. *Mengistu T.M., Che D.* Survey and taxonomy of volunteer computing // *ACM Comput. Surv.* — 2019. — Vol. 52, no. 3. — P. 59.

42. *Lingenbrink D., Iyer K.* Optimal signaling mechanisms in unobservable queues with strategic customers // *Proceedings of the ACM Conference on Economics and Computation.* — 2017. — Pp. 347–347.

43. Discovering statistical models of availability in large distributed systems: An empirical study of seti@home / B. Javadi, D. Kondo, J.M. Vincent, D.P. Anderson // *IEEE Transactions on Parallel and Distributed Systems.* — 2011. — Vol. 22, no. 11. — Pp. 1896–1903.

44. *Javadi B., Thulasiraman P., Buyya R.* Cloud resource provisioning to extend the capacity of local resources in the presence of failures // *Proceedings of the 14th International Conference on High Performance Computing and Communication.* — 2012. — Pp. 311–319.

45. *Monsalve S.A., Carballeira F.G., Mateos A.C.* Analyzing the performance of volunteer computing for data intensive applications // *Proceedings of the International Conference on High Performance Computing and Simulation.* — 2016. — Pp. 597–604.

46. *Kianpisheh S., Kargahi M., Charkari N.M.* Resource availability prediction in distributed systems: An approach for modeling non–stationary transition probabilities // *IEEE Transactions on Parallel and Distributed Systems.* — 2017. — Vol. 28, no. 8. — Pp. 2357–2372.

47. *Yao G., Ding Y., Hao K.* Using imbalance characteristic for fault–tolerant workflow scheduling in cloud systems // *IEEE Transactions on Parallel and Distributed Systems.* — 2017. — Vol. 28, no. 12. — Pp. 3671–3683.

48. *Parkhomenko S.S., Ledeneva T.M.* Scheduling in volunteer computing networks, based on neural network prediction of the job execution time // *International Journal of Parallel, Emergent and Distributed Systems.* — 2018. — Vol. 34, no. 4. — Pp. 430–447.

49. *Lavoie E., Hendren L.* Personal volunteer computing // *Proceedings of the 16th ACM International Conference on Computing Frontiers.* — 2019. — Pp. 240–246.

50. *Егоров В.Ю.* Особенности диспетчеризации процессов при функционировании виртуальных машин // *Системы и средства информ.: дополнительный выпуск.* — 2009. — С. 58–67.

51. *Коваленко И.Н.* Теория массового обслуживания // *Итоги науки. Теория вероятностей. Математическая статистика. Теоретическая кибернетика.* — 1970. — С. 5–100.

52. *Зорин А.В.* Теория конфликтных систем обслуживания при их функционально–статистическом задании // *Докт. диссертация.* — 2016. — 351 с.

53. *Федоткин М.А., Зорин А.В.* Стохастические модели процессов адаптивного управления конфликтными потоками неоднородных требований // *Теория вероятн. и ее примен.* — 2020. — Т. 65, № 1. — С. 163–164.

54. *Кудрявцев Е.В., Федоткин М.А.* Анализ дискретной модели системы адаптивного управления конфликтными неоднородными потоками // *Вестник Московского университетя. Сер. 15: Вычисл. матем. и киберн.* — 2019. — Т. 1. — С. 19–26.

55. *Федоткин М.А.* Оптимальное управление конфликтными потоками и маркированные точечные процессы с выделенной дискретной компонентой. I // *Литовский математический сборник.* — 1988. — Т. 28, № 4. — С. 783–794.

56. *Федоткин М.А.* Оптимальное управление конфликтными потоками и маркированные точечные процессы с выделенной дискретной компонентой. II // *Литовский математический сборник.* — 1989. — Т. 29, № 1. — С. 148–159.

57. *Whitt W.* A broad view of queueing theory through one issue // *Queueing Systems.* — 2018. — Vol. 89. — Pp. 3–14.

58. *Boon M., Boxma O., Foss S.* Editorial introduction to “100 views on queues” // *Queueing Systems.* — 2022. — Vol. 100. — Pp. 167—-168.

59. *Печинкин А.В.* Анализ однолинейных систем массового обслуживания с различными дисциплинами обслуживания // *Докт. диссертация.* — 1985. — 311 с.

60. *Калашников В.В., Рачев С.Т.* Математические методы построения стохастических моделей обслуживания. — М.: Наука. Гл. ред. физ.-мат.лит., 1988. — 312 с.

61. *Золотарев В.М.* Метрические расстояния в пространствах случайных величин и их распределений // *Мат. сб.* — 1976. — Т. 143, № 101. — С. 416–454.

62. *Соколов А.Е., Ушаков И.А.* Математические методы моделирования при создании систем связи // *Техника средств связи. Сер. АСУ.* — 1978. — № 2. — С. 3–11.

63. *Соколов А.Е.* Системы связи и системный анализ // *Техника средств связи. Сер. АСУ.* — 1979. — № 2. — С. 49–58.

64. *Crovella M.E.* Performance evaluation with heavy tailed distributions // *Lecture Notes in Computer Science.* — 2001. — Vol. 2221. — Pp. 1–10.

65. *Rai I., Urvoy-Keller G., Biersack E.* Analysis of LAS scheduling for job size distributions with high variance // *ACM SIGMETRICS Perform. Eval. Rev.* — 2003. — Vol. 31, no. 1. — Pp. 218–228.

66. *Feng H., Misra V., Rubenstein D.* PBS: A unified priority–based CPU scheduler // *ACM SIGMETRICS Perform. Eval. Rev.* — 2007. — Vol. 35, no. 1. — Pp. 203–214.

67. *Nuyens M., Wierman A., Zwart B.* Preventing large sojourn times using SMART scheduling // *Oper. Res.* — 2008. — Vol. 56, no. 1. — Pp. 88–101.

68. A new heavy–tailed discrete distribution for LRD $M/G/\infty$ sample generation / A. Suarez-Gonzalez, J. Lopez-Ardao, C. Lopez-Garccia et al. // *Perform. Eval.* — 2002. — Vol. 47. — Pp. 197–219.

69. *Leland W., Ott T.J.* Load–balancing heuristics and process behavior // *SIGMETRICS Perform. Eval. Rev.* — 1986. — Vol. 14, no. 1. — Pp. 54–69.

70. *Wierman A., Nuyens M.* Scheduling despite inexact job–size information // *SIGMETRICS Perform. Eval. Rev.* — 2008. — Vol. 35, no. 1. — Pp. 25–36.

71. *Wierman A., Harchol-Balter M., Osogami T.* Nearly insensitive bounds on SMART scheduling // *SIGMETRICS Perform. Eval. Rev.* — 2005. — Vol. 33, no. 1. — Pp. 205–216.

72. *Mailach R., Down D.* Scheduling jobs with estimation errors for multi–server systems // *Proceedings of the 29th International Teletraffic Congress.* — 2017. — Pp. 10–18.

73. *Wierman A.* Fairness and classifications // *SIGMETRICS Perform. Eval. Rev.* — 2007. — Vol. 34, no. 4. — Pp. 4–12.

74. *Ward A., Whitt W.* Predicting response times in processor–sharing queues // *Proceedings of the Fields Institute Conference on Communication Networks.* — 2000. — Pp. 1–29.

75. *Яшков С.Ф.* Анализ очередей в ЭВМ. — М.: Радио и связь, 1989. — 216 с.

76. *Беляев Ю.К., Чистякова Н.В.* Непараметрическая оценка распределения длины требования в системе с дисциплиной разделения процессора // *Вероятностные процессы и их приложения / МИЭМ.* — 1987. — С. 12–17.

77. *Dell'Amico M., Carra D., Michiardi P.* PSBS: Practical size–based scheduling // *IEEE Transactions on Computers.* — 2015. — Vol. 99. — Pp. 1–15.

78. *Friedman E. J., Henderson S. G.* Fairness and efficiency in web server protocols // *SIGMETRICS Perform. Eval. Rev.* — 2003. — Vol. 31. — Pp. 229–237.

79. *Nagle J.* On packet switches with infinite storage // *IEEE Transactions on Communications.* — 1987. — Vol. 35, no. 4. — Pp. 435–438.

80. *Gorinsky S., Jechlitschek C.* Fair efficiency, or low average delay without starvation // *Proceedings of the 16th International Conference on Computer Communications and Networks.* — 2007. — Pp. 424–429.

81. *Wierman A.* Fairness and scheduling in single server queues // *Surveys in Operations Research and Management Science.* — 2011. — Vol. 16, no. 1. — Pp. 39–48.

82. *Raz D., Levy H., Avi-Itzhak B.* A resource–allocation queueing fairness measure // *SIGMETRICS Perform. Eval. Rev.* — 2004. — Vol. 32, no. 1. — Pp. 130–141.

83. *Sandmann W.* A discrimination frequency based queueing fairness measure with regard to job seniority and service requirement // *Proceedings of the 1st Euro NGI Conference on Next Generation Internet Networks.* — 2005. — Pp. 106–113.

84. *Wierman A., Harchol-Balter M.* Classifying scheduling policies with respect to higher moments of conditional response time // *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems.* — 2005. — Pp. 229–240.

85. *Chen Y., Whitt W.* Set–valued performance approximations for the $GI/GI/K$ queue given partial information // *Probability in the Engineering and Informational Sciences.* — 2020. — Pp. 1–23.

86. *Pender J.* The impact of dependence on unobservable queues // *https://cpb-us-e1.wpmucdn.com/blogs.cornell.edu/dist/3/7882/files/2018/06/correlation-23ngfyo.pdf*. — 2017.

87. *Haviv M., Oz B.* Self–regulation of an unobservable queue // *Management Science.* — 2017. — Vol. 64, no. 5. — Pp. 2380–2389.

88. *Kittsteiner T., Moldovanu B.* Priority auctions and queue disciplines that depend on processing time // *Management Science.* — 2005. — Vol. 51, no. 2. — Pp. 236–248.

89. *Shalmon M.* Analysis of the $GI/GI/1$ queue and its variations via the LCFS preemptive resume discipline and its random walk interpretation // *Probability in the Engineering and Informational Sciences.* — 1988. — Vol. 1. — Pp. 215–230.

90. *Baron O.* Regulated random walks and the LCFS backlog probability: Analysis and application // *Oper. Res.* — 2008. — Vol. 56, no. 2. — Pp. 471–486.

91. *Abate J., Whitt W.* Limits and approximations for the $M/G/1$ LIFO waiting–time distribution // *Oper. Res. Lett.* — 1997. — no. 20. — Pp. 199–206.

92. *Sigman K.* Queues under preemptive LIFO and ladder height distributions for risk processes: A duality // *Communications in Statistics. Stochastic Models.* — 1996. — Vol. 12, no. 4. — Pp. 725–735.

93. *Asmussen S., Glynn P.* On preemptive–repeat LIFO queues // *Queueing Systems.* — 2017. — Vol. 87, no. 1–2. — Pp. 1–22.

94. *Ritter G., Wacker U.* The mean waiting time in a $G/G/m/\infty$ queue with the LCFS–P service discipline // *Advances in Applied Probability.* — 1991. — Vol. 23, no. 2. — Pp. 406–428.

95. *Fakinos D.* The $G/G/1$ (LCFS/P) queue with service depending on queue size // *European Journal of Operational Research.* — 1992. — Vol. 59. — Pp. 303–307.

96. *Shanthikumar J. G., Sumita U.* On $G/G/1$ queues with LIFO–P service discipline // *Journal of the Operations Research Society of Japan.* — 1986. — Vol. 29, no. 3. — Pp. 220–230.

97. *Fralix B., Riano G.* A new look at transient versions of Little's law, and $M/G/1$ preemptive last–come–first–served queues // *J. Appl. Probab.* — 2010. — no. 47. — Pp. 459–473.

98. *Печинкин А.В.* Об одной инвариантной системе массового обслуживания // *Math. Operationsforsch. und Statist. Ser. Optimization.* — 1983. — Т. 14, № 3. — С. 433–444.

99. *Бочаров П.П., Печинкин А.В., Фонг Н.Х.* Стационарные вероятности состояний системы $MAP/G/1/r$ с повторными заявками и приоритетным обслуживанием первичных заявок // *Автомат. и телемех.* — 2000. — № 8. — С. 68–78.

100. *Печинкин А.В.* Характеристики обслуживания в системе $GI/GI/1/\infty$ при дисциплине LCFS с прерыванием // *Автомат. и телемех.* — 1993. — № 6. — С. 130–141.

101. *Печинкин А.В., Соколов И.А.* Система массового обслуживания с ненадежным прибором в дискретном времени // *Информ. и её примен.* — 2011. — Т. 5, № 4. — С. 6–17.

102. *Печинкин А.В.* Двухприоритетная система с резервированием каналов и марковским входящим потоком // *Информ. и её примен.* — 2011. — Т. 5, № 1. — С. 2–11.

103. *Печинкин А.В., Соколов И.А., Шоргин С.Я.* Ограничение на суммарный объем заявок в дискретной системе $Geo/G/1/\infty$ // *Информ. и её примен.* — 2012. — Т. 6, № 3. — С. 107–113.

104. *Печинкина О.А.* Асимптотическое распределение длины очереди в системе $M/G/1$ с инверсионной вероятностной дисциплиной обслуживания // *Вестник РУДН. Сер.: Прикладн. матем. и информ.* — 1995. — № 1. — С. 87–100.

105. *Таташев А.Г.* Система обслуживания с инверсионной дисциплиной, двумя типами заявок и марковским входящим потоком // *Автомат. и телемех.* — 2003. — № 11. — С. 122–127.

106. *Таташев А.Г.* Одна инверсионная дисциплина обслуживания в одноканальной системе с разнотипными заявками // *Автомат. и телемех.* — 1999. — № 7. — С. 177–181.

107. *Таташев А.Г.* Система $MAP/GI/1/n$ с инверсионной дисциплиной и обслуживанием прерванной заявки заново с прежней длительностью // *Автомат. и телемех.* — 2002. — № 11. — С. 103–107.

108. *Милованова Т.А.* Система $BMAP/G/1$ с инверсионным порядком обслуживания и вероятностным приоритетом // *Автомат. и телемех.* — 2009. — № 5. — С. 155–168.

109. *Pechinkin A.V., Shorgin S.Ya.* The discrete–time queueing system with inversive service order and probabilistic priority // *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools.* — 2008. — Pp. 1–6.

110. *Abate J., Whitt W.* An operational calculus for probability distributions via Laplace transforms // *Advances in Applied Probability.* — 1996. — Vol. 28, no. 1. — Pp. 75–113.

111. *Krishnamoorthy A., Pramod P., Chakravarthy S.* Queues with interruptions: A survey // *TOP.* — 2014. — Vol. 22, no. 1. — Pp. 290–320.

112. *Gaver D.P.* A waiting line with interrupted service, including priorities // *J. Roy. Stat. Soc. B.* — 1962. — Vol. 24, no. 1. — Pp. 73–90.

113. *Finch P.D.* The output process of the queueing system $M/G/1$ // *J. Roy. Stat. Soc. B.* — 1959. — Vol. 21, no. 2. — Pp. 375–380.

114. *Burman D.Y.* Insensitivity in queueing systems // *Advances in Applied Probability.* — 1981. — Vol. 13. — Pp. 846–859.

115. *Рыков В.В.* Управляемые системы массового обслуживания // *Итоги науки и техн. Теория вероятностей. Математическая статистика. Теоретическая кибернетика.* — 1975. — № 12. — С. 43–153.

116. *Башарин Г.П., Бочаров П.П., Коган Я.А.* Анализ очередей в вычислительных сетях: Теория и методы расчета. — М.: Наука, 1989. — 336 с.

117. *Джейсуол Н.К.* Очереди с приоритетами. — М.: Мир, 1973. — 280 с.

118. *Матвеев В.Ф., Ушаков В.Г.* Системы массового обслуживания. — М.: изд–во Моск. ун–та, 1984. — 240 с.

119. Приоритетные системы обслуживания / Б.В. Гнеденко, Э.А. Даниелян, Б.Н. Димитров и др. — М.: изд–во Моск. ун–та, 1973. — 447 с.

120. *Бронштейн О.И., Духовный И.М.* Модели приоритетного обслуживания в информационно–вычислительных системах. — М.: Наука, 1976. — 220 с.

121. *Климов Г.П., Мишкой Г.К.* Приоритетные системы обслуживания с ориентацией. — М.: изд–во МГУ, 1979. — 222 с.

122. *Ушаков В.Г.* Однолинейная система обслуживания с относительным приоритетом // *Известия АН СССР. Техн. кибер.* — 1978. — Т. 1. — С. 76–80.

123. A recursive analysis technique for multidimensionally infinite Markov chains / T. Osogami, A. Wierman, M. Harchol-Balter, A. Scheller-Wolf // *ACM SIGMETRICS Perform. Eval. Rev.* — 2004. — Т. 32, № 2. — С. 3–5.

124. *Neuts M.* Queues solvable without Rouche's theorem // *Oper. Res.* — 1979. — Vol. 27. — Pp. 767–781.

125. *Rumyantsev A.* Stability of multiclass multiserver models with automata–type phase transitions // *CEUR Workshop Procee.* — 2021. — Vol. 2792. — Pp. 213–225.

126. *Hordijk A., van Dijk N.M.* Adjoint processes, job local balance and insensitivity for stochastic networks // *Bull. Internat. Statist. Inst.* — 1983. — Vol. 50. — Pp. 776–788.

127. *van Dijk N.M.* Simple bounds for queueing systems with breakdowns // *Perform. Eval.* — 1988. — Vol. 8. — Pp. 117–128.

128. An integral equation approach to the $M/G/2$ queue / C. Knessl, B.J. Matkowsky, Z. Schuss, C. Tier // *Oper. Res.* — 1990. — Vol. 38, no. 3. — Pp. 506–518.

129. *Hokstad P.* On the steady–state solution of the $M/G/2$ queue // *Advances in Applied Probability.* — 1979. — Vol. 11, no. 1. — Pp. 240–255.

130. *Wiens D. P.* On the busy period distribution of the $M/G/2$ queueing system // *J. Appl. Probab.* — 1989. — Vol. 26, no. 4. — Pp. 858–865.

131. *Boxma O.J., Deng Q., Zwart B.* Waiting–time asymptotics for the $M/G/2$ queue with heterogeneous servers // *Queueing Systems.* — 2002. — Vol. 40. — Pp. 5–31.

132. *Печинкин А.В.* Двухприориетная система массового обслуживания с инверсионным порядком обслуживания // *Техника средств связи. Сер. СС.* — 1985. — № 1. — С. 72–81.

133. *Вишневский В.М., Дудин А.Н.* Системы массового обслуживания с коррелированными входными потоками и их применение для моделирования телекоммуникационных сетей // *Автомат. и телемех.* — 2017. — № 8. — С. 3–59.

134. *Dudin A.N., Klimenok V.I., Vishnevsky V.M.* The theory of queuing systems with correlated flows. — Cham, Switzerland: Springer, 2020. — 410 pp.

135. *Bent N.* On a queuing model where potential customers are discouraged by queue length // *Scandinavian Journal of Statistics.* — 1975. — Vol. 2, no. 1. — Pp. 34–42.

136. *Abouee-Mehrizi H., Baron O.* State–dependent $M/G/1$ queueing systems // *Queueing Systems.* — 2016. — Vol. 82, no. 1–2. — Pp. 121–148.

137. *Kerner Y.* The conditional distribution of the residual service time in the $M_n/G/1$ queue // *Stochastic Models.* — 2008. — Vol. 24. — Pp. 364–375.

138. *Gupta U.C., Srinivasa Rao T.S.S.* On the analysis of single server finite queue with state dependent arrival and service processes: $M_n/G_n/1/K$ // *OR Spectrum.* — 1998. — Vol. 20. — Pp. 83–89.

139. *Бочаров П.П., Шлумпер Л.О.* Однолинейная система массового обслуживания с фоновыми заявками // *Автомат. и телемех.* — 2005. — № 6. — С. 74–88.

140. *Lee T.T.* $M/G/1/N$ queue with vacation time and exhaustive service discipline // *Oper. Res.* — 1984. — Vol. 32. — Pp. 774–785.

141. Analysis of queueing system with non–preemptive time limited service and impatient customers / C. Kim, A. Dudin, O. Dudina, V. Klimenok // *Methodol. Comput. Appl. Probab.* — 2020. — Vol. 22. — Pp. 401–432.

142. *Kempa W.M., Marjasz R.* Distribution of the time to buffer overflow in the $M/G/1/N$-type queueing model with batch arrivals and multiple vacation policy // *J. Oper. Res. Soc.* — 2020. — Vol. 71, no. 3. — Pp. 447–455.

143. *Cooper R.B., Niu S.* Benes's formula for $M/G/1$-FIFO explained by preemptive–resume LIFO // *J. Appl. Probab.* — 1986. — Vol. 23, no. 2. — Pp. 550–554.

144. *Niu S.* Representing workloads in $GI/G/1$ queues through the preemptive–resume LIFO queue discipline // *Queueing Systems.* — 1988. — Vol. 3, no. 2. — Pp. 157–178.

145. *Ephremides A., Varaiya P., Walrand J.* A simple dynamic routing problem // *IEEE Transactions on Automatic Control.* — 1980. — Vol. 25, no. 4. — Pp. 690–693.

146. *Liu Z., Towsley D.* Optimality of the round–robin routing policy // *J. Appl. Probab.* — 1994. — Vol. 31, no. 2. — Pp. 466–475.

147. *Liu Z.R., Righter R.* Optimal load balancing on distributed homogeneous unreliable processors // *Oper. Res.* — 1998. — Vol. 46. — Pp. 563–573.

148. *Altman E., Gaujal B., Hordijk A.* Balanced sequences and optimal routing // *J. ACM.* — 2000. — Vol. 47, no. 4. — Pp. 752–775.

149. *Combe M.B., Boxma O.J.* Optimization of static traffic allocation policies // *Theoretical Computer Science.* — 1994. — Vol. 125. — Pp. 17–43.

150. *Bell C. H., Stidham S.* Individual versus social optimization in the allocation of customers to alternative servers // *Management Science.* — 1983. — Vol. 29, no. 7. — Pp. 831–839.

151. *Tang C.S., van Vliet M.* Traffic allocation for manufacturing systems // *European Journal of Operational Research.* — 1994. — Vol. 75, no. 1. — Pp. 171–185.

152. *Ibaraki T.I., Katoh N.* Resource allocation problems: Algorithmic approaches. — 2nd edition. — Cambridge: MIT Press, 1988. — 246 pp.

153. *Tang C., van Vliet M.* Traffic allocation for manufacturing systems // *European Journal of Operational Research.* — 1994. — Vol. 75. — Pp. 171–185.

154. *Громов А.И.* О некоторых алгоритмах динамического распределения потока // *Сб. научн. трудов "Системы массового обслуживания и информатика".* — 1978. — С. 79–83.

155. *Башарин Г.П., Громов А.И.* Матричный метод нахождения стационарного распределения для некоторых нестандартных СМО // *Автомат. и телемех.* — 1978. — Т. 1. — С. 29–38.

156. *Вишневский В.М., Семенова О.В.* Системы поллинга: Теория и применение в широкополосных беспроводных сетях. — М.: Техносфера, 2007. — 312 с.

157. *Саксонов Е.А.* Многоканальная СМО со стохастическим управлением диспетчеризацией и групповым обслуживанием // *Тезисы докладов Республиканской научно–технической школы–семинара "Анализ и синтез систем массового обслуживания и сетей ЭВМ". Часть I.* — 1990. — С. 212–217.

158. *Hoekstra G.J., van der Mei R.D., Bhulai S.* Optimal job splitting in parallel processor sharing queues // *Stochastic Models.* — 2012. — Vol. 28. — Pp. 144–166.

159. *Сигнаевский В.А., Коган Я.А.* Методы оценки быстродействия вычислительных систем. — М.: Наука. Гл. ред. физ.-мат.лит., 1991. — 256 с.

160. *Hajek B.* The proof of a folk theorem on queuing delay with applications to routing in networks // *J. ACM.* — 1983. — Vol. 30, no. 4. — Pp. 834–851.

161. *Humblet P.A.* Determinism minimizes waiting times in queues // *Technical Report, LIDS — Department of Electrical Engineering and Computer Science.* — 1982.

162. *Рогозин Б.А.* Некоторые экстремальные задачи теории массового обслуживания // *Теория вероятностей и ее прим.* — 1966. — Vol. 11, no. 1. — Pp. 161–169.

163. *Климов Г.П.* Экстремальные задачи в теории массового обслуживания // *Сб. Кибернетику – на службу коммунизму.* — 1964. — Vol. 2. — Pp. 310–325.

164. *Hajek B.* Extremal splittings of point processes // *Math. Oper. Res.* — 1985. — Vol. 10, no. 4. — Pp. 1–42.

165. *Morse M., Hedlund G.* Symbolic dynamics II: Sturmian trajectories // *American Journal of Mathematics.* — 1940. — Vol. 62. — Pp. 1–42.

166. *Hordijk A., van der Laan D.A.* Periodic routing to parallel queues and billiard sequences // *Math. Method Oper. Res.* — 2004. — Vol. 59, no. 2. — Pp. 173–192.

167. *van der Laan D.A.* The structure and performance of optimal routing sequences // *PhD Thesis.* — 2003. — 208 pp.

168. *van der Laan D.A.* Routing jobs to servers with deterministic service times // *Math. Oper. Res.* — 2005. — Vol. 30, no. 1. — Pp. 195–224.

169. *Gaujal B., Hyon E.* Optimal routing policy in two deterministic queues // *Calculateurs Paralleles.* — 2001. — Vol. 13. — Pp. 601–634.

170. *Anselmi J., Gaujal B., Nesti T.* Control of parallel non–observable queues: Asymptotic equivalence and optimality of periodic policies // *Stochastic Systems.* — 2015. — Vol. 5. — Pp. 120–145.

171. *Arian Y., Levy Y.* Algorithms for generalized round robin routing // *Oper. Res. Lett.* — 1992. — Vol. 12. — Pp. 313–319.

172. *Sano S., Miyoshi N.* Applications of m–balanced sequences to some network scheduling problems // *Proceedings of the 5th Workshop on Discrete Event Systems.* — 2000. — Pp. 317–325.

173. *Hordijk A., van der Laan D.A.* On the average waiting time for regular routing to deterministic queues // *Math. Oper. Res.* — 2005. — Vol. 30. — Pp. 521–544.

174. *Коновалов М.Г., Разумчик Р.В.* Обзор моделей и алгоритмов размещения заданий в системах с параллельным обслуживанием // *Информ. и её примен.* — 2015. — Т. 9, № 4. — С. 56–67.

175. *Semchedine F., Bouallouche-Medjkoune L., Aissani D.* Task assignment policies in distributed server systems: A survey // *J. Netw. Comput. Appl.* — 2011. — Vol. 34, no. 4. — Pp. 1123–1130.

176. *Harchol-Balter M., Crovella M., Murta C.* On choosing a task assignment policy for a distributed server system // *Lecture Notes in Computer Science.* — 1998. — Vol. 1469. — Pp. 231–242.

177. *Hordijk A., Koole G.* On the assignment of customers to parallel queues // *Probability in the Engineering and Informational Sciences.* — 1992. — Vol. 6, no. 4. — Pp. 495–511.

178. *Коновалов М.Г., Разумчик Р.В.* Диспетчеризация в системе с параллельным обслуживанием с помощью распределенного градиентного управления марковской цепью // *Информ. и её примен.* — 2021. — Т. 15, № 3. — С. 41–50.

179. On value functions for FCFS queues with batch arrivals and general cost structures / E. Hyytiä, R. Righter, J. Virtamo, L. Viitasaari // *Perform. Eval.* — 2020. — Vol. 138. — P. 102083.

180. *Stidham S., Weber R.* A survey of Markov decision models for control of networks of queues // *Queueing Systems.* — 1993. — Vol. 13. — Pp. 291–314.

181. *Hyytiä E., Righter R.* Simulation and performance evaluation of mission critical dispatching systems // *Perform. Eval.* — 2019. — Vol. 135. — P. 102038.

182. *Hyytiä E., Righter R.* Routing jobs with deadlines to heterogeneous parallel servers // *Oper. Res. Lett.* — 2016. — Vol. 44. — Pp. 507–513.

183. *Wang A., Ziedins I.* Probabilistic selfish routing in parallel batch and single–server queues // *Queueing Systems.* — 2018. — Vol. 88. — Pp. 389–407.

184. $M/M/1 - PS$ queue and size–aware task assignment / E. Hyytiä, J. Virtamo, S. Aalto, A. Penttinen // *Perform. Eval.* — 2011. — Vol. 68, no. 11. — Pp. 1136–1148.

185. *Hellemans T., Bodas T., Van Houdt B.* Performance analysis of workload dependent load balancing policies // *Proc. ACM Meas. Anal. Comput. Syst.* — 2019. — Vol. 3, no. 2. — Pp. 1–35.

186. Dispatching fixed–sized jobs with multiple deadlines to parallel heterogeneous servers / E. Hyytiä, R. Righter, O. Bilenne, X. Wuc // *Perform. Eval.* — 2017. — Vol. 114. — Pp. 32–44.

187. *Hyytiä E., Righter R., Samuelsson S.* Beyond shortest queue routing with heterogeneous servers and general cost functions // *Proceedings of the 11th EAI International Conference on Performance Evaluation Methodologies and Tools.* — 2017. — Pp. 206–213.

188. *Foss S., Stolyar A.L.* Large–scale Join-Idle-Queue system with general service times // *J. Appl. Probab.* — 2017. — Vol. 54, no. 4. — Pp. 995–1007.

189. *Terekhov D., Down D., Beck C.* Queueing–theoretic approaches for dynamic scheduling: A survey // *Surveys in Operations Research and Management Science.* — 2014. — Vol. 19. — Pp. 105–129.

190. *Wierman A.* Scheduling for today's computer systems: Bridging theory and practice // *PhD Thesis.* — 2007. — 359 pp.

191. *Hyytiä E.* Lookahead actions in dispatching to parallel queues // *Perform. Eval.* — 2013. — Vol. 70, no. 10. — Pp. 859–872.

192. *Круглов В.М.* Пуассоновские процессы. — М.: изд–й отдел факультета ВМиК МГУ им. М.В. Ломоносова, 2008. — 128 с.

193. *Уолрэнд Дж.* Введение в теорию сетей массового обслуживания. — Москва: Мир, 1993. — 335 с.

194. *Aalto S., Ayesta U., Righter R.* On the Gittins index in the $M/G/1$ queue // *Queueing Systems.* — 2009. — Vol. 63. — P. 437.

195. *Edmonds J.* Scheduling in the dark // *Theoretical Computer Science.* — 2000. — Vol. 235. — Pp. 109–141.

196. *Avrahami N., Azar Y.* Minimizing total flow time and total completion time with immediate dispatching // *Algorithmica.* — 2007. — Vol. 47. — Pp. 253–268.

197. *Altman E., Jimenez T. ad Nunez-Queija R., Yechiali U.* Optimal routing among $\cdot/M/1$ queues with partial information // *Stochastic Models.* — 2004. — Vol. 20, no. 2. — P. 149–171.

198. *Gaujal B., Hyon E., Jean-Marie A.* Optimal routing in two parallel queues with exponential service times // *Discrete Event Dynamic Systems.* — 2006. — Vol. 16, no. 2. — Pp. 71–107.

199. *Brun O.* Performance of non–cooperative routing over parallel non-observable queues // *Probability in the Engineering and Informational Sciences.* — 2016. — Vol. 30. — Pp. 455–469.

200. *Anselmi J.* Asymptotically optimal open–loop load balancing // *Queueing Systems.* — 2017. — Vol. 87. — Pp. 245–267.

201. *Herrmann J.W.* Using aggregation to construct periodic policies for routing jobs to parallel servers with deterministic service times // *J. Sched.* — 2012. — Vol. 15. — Pp. 181–192.

202. *Milito R., Ferndndez-Gaucherand E.* Open–loop routing of N arrivals to M parallel queues // *Proceedings of the 33rd Conference on Decision and Control.* — 1994. — Pp. 184–189.

203. *Sethuraman J., Squillante M.* Optimal stochastic scheduling in multiclass parallel queues // *ACM SIGMETRICS Perform. Eval. Rev.* — 1999. — Vol. 27, no. 1. — Pp. 93–102.

204. *Neely M. J., Modiano E.* Convexity in queues with general inputs // *IEEE Transactions on Information Theory.* — 2005. — Vol. 51, no. 2. — Pp. 706–714.

205. *Клейнрок Л.* Вычислительные системы с очередями. — М.: Мир, 1979. — 600 с.

206. *Бертсекас Д., Галлагер Р.* Сети передачи данных. — М.: Мир, 1989. — 544 с.

207. *Мизин И.А., Кулешов А.П.* Сети ЭВМ // *Итоги науки и техн. Сер. Тех. кибернет.* — 1986. — Т. 20. — С. 3–135.

208. *Жожикашвили В.А., Вишневский В.М.* Сети массового обслуживания. Теория и применение к сетям ЭВМ. — М.: Радио и связь, 1988. — 192 с.

209. *Haviv M., Roughgarden T.* The price of anarchy in an exponential multi–server // *Oper. Res. Lett.* — 2007. — Vol. 35, no. 4. — Pp. 421–426.

210. *Altman E., Ayesta U., Prabhu B.* Load balancing in processor sharing systems // *Telecommunication Systems.* — 2011. — Vol. 47, no. 1. — Pp. 35–48.

211. *Yum T.* The design and analysis of a semidynamic deterministic routing tables // *IEEE Trans. commun.* — 1981. — Vol. 29, no. 4. — Pp. 498–504.

212. *Yum T., Schwartz M.* The join–biased–queue rule and its application to routing in computer communication networks // *IEEE Trans. commun.* — 1981. — Vol. 29, no. 4. — Pp. 505–511.

213. *Shinya S., Naoto M., Ryohei K.* m–Balanced words: A generalization of balanced words // *Theor. Comput. Sci.* — 2004. — Vol. 314, no. 1. — Pp. 97–120.

214. *Altman E., Gaujal B., Hordijk A.* Regular ordering and applications in control policies // *Discrete Event Dynamic Systems.* — 2002. — Vol. 12, no. 2. — Pp. 187–210.

215. *Altman E., Gaujal B., Hordijk A.* Discrete–event control of stochastic networks: Multimodularity and regularity. — NJ, USA: Springer-Verlag New York, 2003. — 313 с.

216. *van der Laan D.* Routing jobs to servers with deterministic service times. — Leiden University, 2000.

217. Optimal balanced control for call centers / S. Bhulai, T. Farenhorst-Yuan, B. Heidergott, D. van der Laan // *Annals of Operations Research.* — 2012. — Vol. 201. — Pp. 39–62.

218. *Hordijk A., van der Laan D.A.* The unbalance and bounds on the average waiting time for periodic routing to one queue. The unbalance of routing sequences // *Math. Method Oper. Res.* — 2004. — Vol. 59. — Pp. 1–23.

219. *Altman E., Gaujal B., Hordijk A.* Admission control in stochastic event graphs // *IEEE Trans. Automat. Control.* — 2000. — Vol. 45. — Pp. 854–867.

220. *Altman E., Gaujal B., Hordijk A.* Multimodularity, convexity and optimization properties // *Math. Oper. Res.* — 2000. — Vol. 25. — Pp. 324–347.

221. Synchronization and linearity / F. Baccelli, G. Gohen, G.J. Olsder, J.P. Quadrat. — 2nd edition. — Cambridge: MIT Press, 1992. — 246 pp.

222. *Itai A., Rosberg Z.* A golden ratio control policy for a multi–access channel // *IEEE Trans. Automat. Control.* — 1984. — Vol. 29. — Pp. 712–718.

223. *Rosberg Z.* Deterministic routing to buffered channels // *IEEE Trans. Commun.* — 1986. — Vol. 34. — Pp. 504–507.

224. *Anselmi J., Gaujal B.* The price of forgetting in parallel and nonobservable queues // *Perform. Eval.* — 2011. — Vol. 68, no. 12. — Pp. 1291–1311.

225. *Hordijk W., Hordijk A., Heidergott B.* A genetic algorithm for finding good balanced sequences in a customer assignment problem with no state information // *Asia Pacific Journal of Operational Research.* — 2015. — Vol. 32, no. 2. — P. 1550015.

226. *Sabria F., Daganzo C.* Approximate expressions for queueing systems with scheduled arrivals and established service // *Transportation Science.* — 1989. — Vol. 23, no. 3. — Pp. 159–165.

227. *Magistad J.G., Anderson R.L.* Some steady state and transient solutions for sampled queues. — North Carolina: Institute of Statistics, 1964. — 81 pp.

228. *Kitaev M. Yu.* The $M/G/1$ processorsharing model: Transient behavior // *Queueing Systems.* — 1993. — Vol. 14. — Pp. 239–273.

229. *Reiman M., Simon B.* An interpolation approximation for queueing systems with Poisson input // *Oper. Res.* — 1988. — Vol. 36, no. 3. — Pp. 454–469.

230. *Garcia J., Brun O., Gauchard D.* Transient analytical solution of $M/D/1/N$ queues // *J. Appl. Probab.* — 2002. — Vol. 39, no. 4. — Pp. 853–864.

231. *Wang C.* On the transient delays of $M/G/1$ queues // *J. Appl. Probab.* — 1999. — Vol. 36, no. 3. — Pp. 882–893.

232. *Abate J., Whitt W.* Transient behavior of the $M/G/1$ workload process // *Oper. Res.* — 1994. — Vol. 42, no. 4. — Pp. 750–764.

233. *Stadje W.* A new approach to the Lindley recursion // *Statistics and Probability Letters.* — 1997. — Vol. 31, no. 3. — Pp. 169–175.

234. *Ackroyd M.* Approximate characterisation of nonstationary discrete time $GI/G/1$ systems // *Perform. Eval.* — 1986. — Vol. 6, no. 2. — Pp. 117–123.

235. *Fredericks A.A.* A class of approximations for the waiting time distribution in a $GI/G/1$ queueing system // *The Bell System Technical Journal.* — 1982. — Vol. 61, no. 3. — Pp. 295–325.

236. *Jagerman D.L.* Approximate mean waiting times in transient $GI/G/1$ queues // *The Bell System Technical Journal.* — 1982. — Vol. 61, no. 8. — Pp. 2003–2022.

237. *Konheim A.G.* An elementary solution of the queuing system $GI/G/1$ // *SIAM J. Comput.* — 1975. — Vol. 4, no. 4. — Pp. 540–545.

238. *Grassmann W.K., Jain J.L.* Numerical solutions of the waiting time distribution and idle time distribution of the arithmetic $GI/G/1$ queue // *Oper. Res.* — 1989. — Vol. 37, no. 1. — Pp. 141–150.

239. *Ackroyd M.* Computing the waiting time distribution for the $G/G/1$ queue by signal processing method // *IEEE Transactions on Communications.* — 1980. — Vol. 28, no. 1. — Pp. 52–58.

240. *Ackroyd M.* Stationary and cyclostationary finite buffer behaviour computation via Levinson's method // *AT& T Bell Lab. Tech. J.* — 1984. — Vol. 63. — Pp. 2159–2170.

241. *Hokstad P.* Relations for the workload of the $GI/G/s$ queue // *Advances in Applied Probability.* — 1985. — Vol. 17, no. 4. — Pp. 887–904.

242. *Wolf D.* Approximating the stationary waiting time distribution function of $GI/GI/1$–queues with arithmetic interarrival time and service time distribution function // *OR Spektrum.* — 1982. — Vol. 4. — Pp. 135–148.

243. *Alsmeyer G.* Random recursive equations and their distributional fixed points. — Springer, 2012. — 297 pp.

244. *Akar N., Sohraby K.* System–theoretical algorithmic solution to waiting times in semi–Markov queues // *Perform. Eval.* — 2019. — Т. 66, № 11. — С. 587–606.

245. *Minh D.* The discrete–time single-server queue with time–inhomogeneous compound Poisson input and general service time distribution // *J. Appl. Probab.* — 1978. — Т. 15, № 3. — С. 590–601.

246. *Bambos N., Walrand J.* An invariant distribution for the $G/G/1$ queueing operator // *Advances in Applied Probability.* — 1990. — Vol. 22, no. 1. — Pp. 254–256.

247. *Shanthikumar J., Sumita U.* Modified Lindley process with replacement: Dynamic behavior, asymptotic decomposition and applications // *J. Appl. Probab.* — 1989. — Vol. 26, no. 3. — Pp. 552–565.

248. *Peigne M., Woess W.* Recurrence of two–dimensional queueing processes, and random walk exit times from the quadrant // *Ann. Appl. Probab.* — 2021. — Vol. 31, no. 31. — Pp. 2519–2537.

249. *Fryer M.J., Winsten C.B.* An algorithm to compute the equilibrium distribution of a one–dimensional bounded random walk // *Oper. Res.* — 1986. — Vol. 34, no. 3. — Pp. 449–454.

250. *Малышев В.А.* Уравнения Винера–Хопфа и их применения в теории вероятностей // *Итоги науки и техн. Сер. Теор. вероятн. Мат. стат. Теор. кибернет.* — 1976. — Т. 13, № 3. — С. 5–35.

251. *Aven T.* Upper (lower) bounds on the mean of the maximum (minimum) of a number of random variables // *J. Appl. Probab.* — 1985. — Vol. 22, no. 3. — Pp. 723–728.

252. *Gravey A.* A simple construction of an upper bound for the mean of the maximum of n identically distributed random variables // *J. Appl. Probab.* — 1985. — Vol. 22, no. 4. — Pp. 844–851.

253. *Gallot S.* A bound for the maximum of a number of random variables // *J. Appl. Probab.* — 1966. — Vol. 2, no. 3. — Pp. 556–558.

254. *Merlevede F., Peligrad M.* Rosenthal–type inequalities for the maximum of partial sums of stationary processes and examples // *The Annals of Probability.* — 2013. — Vol. 41, no. 2. — Pp. 914–960.

255. *Marshall K.T.* Some inequalities in queueing // *Oper. Res.* — 1968. — Vol. 16. — Pp. 651–665.

256. *Bergman R., Stoyan D.* On exponential bounds for the waiting time distribution function in $GI/G/1$ // *J. Appl. Probab.* — 1976. — Vol. 13, no. 2. — Pp. 411–417.

257. *Bertsimas D., Natarajan K., Teo C.* Tight bounds on expected order statistics // *Probability in the Engineering and Informational Sciences.* — 2006. — Vol. 20. — Pp. 667–686.

258. *Williamson R., Downs T.* Probabilistic arithmetic. I. Numerical methods for calculation convolutions and dependency bounds // *International Journal of Approximate Reasoning.* — 1990. — Vol. 4, no. 2. — Pp. 89–158.

259. *Harrison P., Zertal S.* Queueing models of RAID systems with maxima of waiting times // *Perform. Eval.* — 2007. — Vol. 64. — Pp. 664–689.

260. *Seaman J., Odell P.* Variance, upper bounds // *Encyclopedia of Statistical Sciences.* — 1988. — Pp. 480–484.

261. *Madala S., Sinclair J.* Performance of synchronous parallel algorithms with regular structures // *IEEE Transactions on Parallel and Distributed Systems.* — 1991. — Vol. 2, no. 1. — Pp. 105–116.

262. *Crow C., Goldberg D., Whitt W.* Two–moment approximations for Maxima // *Oper. Res.* — 2007. — Vol. 55, no. 3. — Pp. 532–548.

263. *Бочаров П.П., Печинкин А.В.* Теория массового обслуживания: Учебник. — М.: РУДН, 1995. — 529 с.

264. *Akar N.* A matrix analytical method for the discrete time Lindley equation using the generalized Schur decomposition // *Proceeding from the 2006 workshop on tools for solving structured Markov chains.* — 2006. — С. 12–es.

265. *De Vuyst S., Bruneel H., Fiems D.* Computationally efficient evaluation of appointment schedules in health care // *European Journal of Operational Research.* — 2014. — Vol. 237. — Pp. 1142–1154.

266. *Takacs L.* Discrete queues with one server // *J. Appl. Probab.* — 1971. — Vol. 8, no. 4. — Pp. 691–707.

267. *Toyoizumi H.* Evaluating mean sojourn time estimates for the $M/M/1$ queue // *Computers and Mathematics with Applications.* — 1992. — Vol. 24, no. 1/2. — Pp. 7–15.

268. *Whitt W.* A review of $L = \lambda W$ and extensions // *Queueing Systems.* — 1991. — Vol. 9. — Pp. 235–268.

269. *Whitt W.* Planning queueing simulations // *Management Science.* — 1989. — Vol. 35, no. 11. — Pp. 1341–1366.

270. *Rykov V.V.* Monotone control of queueing systems with heterogeneous servers // *Queueing Systems.* — 2001. — Vol. 37. — Pp. 391–403.

271. *Viniotis I., Ephremides A.* Extension of the optimality of the threshold policy in heterogeneous multiserver queueing systems // *IEEE Transactions on Automatic Control.* — 1988. — Vol. 1. — Pp. 104–109.

272. *Hyytiä E.* Optimal routing of fixed size jobs to two parallel servers // *INFOR: Information Systems and Operational Research.* — 2013. — Vol. 51, no. 4. — Pp. 215–224.

273. *Lin K.* Decentralized admission control of a queueing system: A game–theoretic model // *Naval Research Logistics.* — 2003. — Vol. 50, no. 7. — Pp. 702–718.

274. *Coffman E.G., Jelenkovic P.* Threshold policies for single–resource reservation systems // *SIGMETRICS Perform. Eval. Rev.* — 2001. — Vol. 28, no. 4. — Pp. 9–10.

275. *Legros B., Jouini O.* Routing in a queueing system with two heterogeneous servers in speed and in quality of resolution // *Stochastic Models.* — 2017. — Vol. 33, no. 3. — Pp. 392–410.

276. *Ширяев А.Н.* Стохастические задачи о разладке. — Москва: МЦНМО, 2016. — 392 с.

277. *Walrand J.* A note on optimal control of a queueing system with two heterogeneous servers // *Systems and Control Letters.* — 1984. — no. 4. — Pp. 131—134.

278. *Koole G.* A simple proof of the optimality of a threshold policy in a two–server queueing system // *Systems and Control Letters.* — 1995. — Vol. 26, no. 5. — Pp. 301–303.

279. On choosing a task assignment policy for a distributed server system / M. Harchol-Balter, M. Crovella, , C. Murta // *J. Parallel Distr. Com.* — 1999. — Vol. 59. — Pp. 204–228.

280. Dispatching problem with fixed size jobs and processor sharing discipline / E. Hyytiä, A. Penttinen, S. Aalto, J. Virtamo // *Proceedings of the 23rd International Teletraffic Congress.* — 2011. — Vol. 59. — Pp. 190–197.

281. *Рыков В.В., Ефросинин Д.В.* К проблеме медленного прибора // *Автомат. и телемех.* — 2009. — № 12. — С. 81–91.

282. *Stockbridge R.H.* A martingale approach to the slow server problem // *J. Appl. Probab.* — 1991. — Т. 28, № 2. — С. 480–486.

283. *Vericourt F., Zhou Y.P.* On the incomplete results for the heterogeneous server problem // *Queueing Systems.* — 2006. — Vol. 52, no. 3. — Pp. 189–191.

284. *Lin W., Kumar P.* Optimal control of a queueing system with two heterogeneous servers // *IEEE Transactions on Automatic Control.* — 1984. — Vol. 29, no. 8. — Pp. 696–703.

285. *Krishnan K.R.* Joining the right queue: a state–dependent decision rule // *IEEE Transactions on Automatic Control.* — 1990. — Vol. 35, no. 1. — Pp. 104–108.

286. *Hyytiä E., Virtamo J.* Dynamic routing and wavelength assignment using first policy iteration // *Proceedings of the 5th IEEE Symposium on Computers and Communications.* — 2000. — Pp. 146–151.

287. *Hyytiä E., Righter R., Aalto S.* Task assignment in a server farm with switching delays and general energy–aware cost structure // *Perform. Eval.* — 2014. — Vol. 75–76. — Pp. 17–35.

288. *Schrage L.* A proof of the optimality of the shortest remaining processing time discipline // *Oper. Res.* — 1968. — Vol. 16, no. 4. — Pp. 687–690.

289. *Grosof I.* Open problem — $M/G/k$-SRPT under medium load // *Stochastic Systems.* — 2019. — Vol. 9, no. 3. — Pp. 297–298.

290. *Grosof I., Scully Z., Harchol-Balter M.* Load Balancing Guardrails: Keeping your heavy traffic on the road to low response times // *Proc. ACM Meas. Anal. Comput. Syst.* — 2019. — Vol. 3, no. 2. — P. 42.

291. *Grosof I.* Open Problem — $M/G/1$ Scheduling with preemption delays // *Stochastic Systems.* — 2019. — Vol. 9, no. 3. — Pp. 311–312.

292. *Grosof I., Harchol-Balter M., Scheller-Wolf A.* Simple near–optimal scheduling for the $M/G/1$ // *Abstracts of the SIGMETRICS/Performance Joint International Conference on Measurement and Modeling of Computer Systems.* — 2020. — Pp. 37–38.

293. *Мейханаджян Л.А., Разумчик Р.В.* Стационарные характеристики системы $M/G/2/\infty$ с одним частным случаем дисциплины инверсионного порядка обслуживания с обобщенным вероятностным приоритетом // *Информ. и её примен.* — 2020. — Т. 14, № 2. — С. 10–15.

294. *Horvath I., Razumchik R., Telek M.* The resampling $M/G/1$ non–preemptive LIFO queue and its application to systems with uncertain service time // *Perform. Eval.* — 2019. — Vol. 134. — Pp. 102000–1–102000–13 (WoS Q2).

295. *Мейханаджян Л.А., Разумчик Р.В.* Система массового обслуживания $Geo/G/1$ с инверсионным порядком обслуживания и ресамплингом в дискретном времени // *Информ. и её примен.* — 2019. — Т. 13, № 4. — С. 60–67.

296. *Milovanova T.A., Meykhanadzhyan L.A., Razumchik R.V.* Bounding moments of sojourn time in $M/G/1$ FCFS queue with inaccurate job size information and additive error: Some observations from numerical experiments // *CEUR Workshop Procee.* — 2018. — Vol. 2236. — Pp. 24–30.

297. *Meykhanadzhyan L., Razumchik R.* New scheduling policy for estimation of stationary performance characteristics in single server queues with inaccurate job size information // *30th International ECMS Conference on Modelling and Simulation Proceedings.* — 2016. — Pp. 710–716.

298. *Милованова Т.А., Разумчик Р.В.* Однолинейная система массового обслуживания с инверсионным порядком обслуживания с вероятностным приоритетом, групповым пуассоновским потоком и фоновыми заявками // *Информ. и её примен.* — 2020. — Т. 14, № 3. — С. 26–34.

299. Стационарные вероятности состояний в системе обслуживания с инверсионным порядком обслуживания и обобщенным вероятностным приоритетом / Л.А. Мейханаджян, Т.А. Милованова, А.В. Печинкин, Р.В. Разумчик // *Информ. и её примен.* — 2015. — Т. 8, № 3. — С. 28–38.

300. *Konovalov M., Razumchik R.* Minimizing mean response time in batch-arrival non-observable systems with single-server FIFO queues operating in parallel // *Communications of the ECMS.* — 2021. — Vol. 35, no. 1. — Pp. 272–278.

301. *Коновалов М.Г., Разумчик Р.В.* Об одном новом способе диспетчеризации для ненаблюдаемых систем с параллельным обслуживанием и дисциплиной FIFO в серверах // *Информационные процессы.* — 2020. — Т. 20, № 3. — С. 205–214.

302. *Konovalov M., Razumchik R.* A simple dispatching policy for minimizing mean response time in non-observable queues with SRPT policy operating in parallel // *Communications of the ECMS.* — 2020. — Vol. 34, no. 1. — Pp. 398–402.

303. *Konovalov M., Razumchik R.* Minimizing mean response time in non-observable distributed systems with processor sharing nodes // *33rd International ECMS Conference on Modelling and Simulation Proceedings.* — 2019. — Vol. 33, no. 1. — Pp. 456–461.

304. *Коновалов М.Г., Разумчик Р.В.* Минимизация среднего времени пребывания в ненаблюдаемых системах с параллельным обслуживанием и дисциплиной справедливого разделения процессора в серверах // *Информационные процессы.* — 2019. — Т. 19, № 3. — С. 327–338.

305. *Коновалов М.Г., Разумчик Р.В.* Комплексное управление в одном классе систем с параллельным обслуживанием // *Информ. и её примен.* — 2019. — Т. 13, № 4. — С. 54–59.

306. *Konovalov M., Razumchik R.* Improving routing decisions in parallel non-observable queues // *Computing.* — 2018. — Vol. 100, no. 10. — Pp. 1059–1079 (WoS Q2).

307. *Konovalov M., Razumchik R.* Using inter-arrival times for scheduling in non-observable queues // *31st International ECMS Conference on Modelling and Simulation Proceedings.* — 2017. — Pp. 667–672.

308. *Коновалов М.Г., Разумчик Р.В.* Управление случайным блужданием с эталонным стационарным распределением // *Информ. и её примен.* — 2018. — Т. 12, № 3. — С. 2–13.

309. *Коновалов М.Г., Разумчик Р.В.* О размещении заданий на двух серверах при неполном наблюдении // *Информ. и её примен.* — 2016. — Т. 10, № 4. — С. 57–67.

310. *Коновалов М.Г., Разумчик Р.В.* Об управлении размером очереди в системе с одним сервером // *Системы и средства информ.* — 2017. — Т. 27, № 4. — С. 4–15.

311. *Konovalov M., Razumchik R.* Iterative algorithm for threshold calculation in the problem of routing fixed size jobs to two parallel servers // *Journal of Telecommunications and Information Technology.* — 2015. — no. 3. — Pp. 32–38.

312. *Konovalov M., Razumchik R.* Simulation of job allocation in distributed processing systems // *International Congress on Ultra Modern Telecommunications and Control Systems and Workshops Proceedings.* — 2015. — Pp. 563–569.

313. *Razumchik R.* Two–priority queueing system with LCFS service, probabilistic priority and batch arrivals // *AIP Conference Proceedings.* — 2019. — Vol. 2116. — Pp. 090011–1–090011–3.

314. *Разумчик Р.В.* Стационарные характеристики системы обслуживания с инверсионным порядком обслуживания, вероятностным приоритетом и групповым поступлением разнородных заявок // *Информ. и её примен.* — 2017. — Т. 11, № 4. — С. 10–18.

315. *Разумчик Р.В.* Стационарные распределения, связанные со временем пребывания в состоянии перегрузки системы $MAP/PH/1/r$ с гистерезисным управлением нагрузкой // *Информ. и её примен.* — 2017. — Т. 11, № 4. — С. 19–25.

316. *Razumchik R.* On $M/G/1$ queue with state-dependent heterogeneous batch arrivals, inverse service order and probabilistic priority // *AIP Conference Proceedings.* — 2017. — Vol. 1863. — Pp. 090006–1–090006–3.

317. *Razumchik R.* Analysis of finite $MAP/PH/1$ queue with hysteretic control of arrivals // *8th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops Proceedings.* — 2016. — Pp. 288–293.

318. *Нагоненко В.А.* О характеристиках одной нестандартной системы массового обслуживания. I // *Изв. АН СССР. Технич. кибернет.* — 1981. — № 1. — С. 187–195.

319. *Нагоненко В.А.* О характеристиках одной нестандартной системы массового обслуживания. II. // *Изв. АН СССР. Технич. кибернет.* — 1981. — № 3. — С. 91–99.

320. *Климов Г.П.* Стохастические системы обслуживания. — М.: Наука, 1966. — 243 с.

321. *Нагоненко В.А.* Системы массового обслуживания с инверсионным порядком обслуживания и вероятностным приоритетом // *Канд. диссертация.* — 1981. — 140 с.

322. *Ивницкий В.А.* О связи стационарных вероятностей состояний систем массового обслуживания в моменты произвольный, поступления и ухода

требований // *Изв. АН СССР. Технич. кибернет.* — 1979. — № 1. — С. 99–109.

323. *Морозов Е.В., Дельгадо Р.* Анализ стационарности регенеративных систем обслуживания // *Автомат. и телемех.* — 2009. — № 70. — С. 42–58.

324. *Афанасьева Л.Г., Ткаченко А.В.* Условия стабильности систем с очередью и регенерирующим процессом прерываний обслуживания // *Теория вероятн. и ее примен.* — 2018. — Т. 63, № 4. — С. 623–653.

325. *Neuman F.* Factorizations of matrices and functions of two variables // *Czechoskvak Math. J.* — 1982. — Vol. 32. — Pp. 582–588.

326. *Gauchman H., Rubel L.* Sums of products of functions of x times functions of y // *Linear Algebra Appl.* — 1989. — Vol. 125. — Pp. 19–63.

327. *Даугавет В.А.* О равномерном приближении функции двух переменных, заданной таблично, произведением функций одной переменной // *Ж. вычисл. матем. и матем. физ.* — 1971. — Vol. 11, no. 2. — Pp. 289–303.

328. Tensor product approximation with optimal rank in quantum chemistry / S. Chinnamsetty, M. Espig, B. Khoromskij et al. // *J. Chem. Phys.* — 2007. — Vol. 128. — P. 084110.

329. *Тыртышников Е.Е.* Тензорные аппроксимации матриц, порожденных асимптотически гладкими функциями // *Матем. сб.* — 2003. — Т. 194, № 6. — С. 147–160.

330. *Watson G.A.* A multiple exchange algorithm for multivariate Chebyshev approximation // *SIAM J. Numer. Anal.* — 1975. — Vol. 12, no. 1. — Pp. 46–52.

331. *Oseledets E., Tyrtyshnikov E.* TT–cross approximation for miltidimansional arrays // *Linear Algebra Appl.* — 2010. — Т. 432. — С. 70–88.

332. *Поспелов В.В.* О погрешности приближения функции двух переменных суммами произведений функций одного переменного // *Ж. вычисл. матем. и матем. физ.* — 1978. — Т. 18, № 5. — С. 1307–1308.

333. *Shura-Bura M.R.* Approximation of functions of many variables by functions depending on one variable // *Vychisl. Mat.* — 1957. — no. 2. — Pp. 3–19.

334. *Uschmajew A.* Regularity of tensor product approximations to square integrable functions // *Constructive Approximation.* — 2011. — Vol. 34, no. 3. — Pp. 371–391.

335. *Townsend A., Trefethen L.* An extension of chebfun to two dimensions // *SIAM J. Sci. Comput.* — 2013. — Vol. 35, no. 6. — Pp. 495–518.

336. *Hashemi B., Trefethen L.* Chebfun in three dimensions // *SIAM J. Sci. Comput.* — 2017. — Vol. 39, no. 5. — Pp. C341–C363.

337. *Bebendorf M.* Adaptive cross approximation of multivariate functions // *Constructive Approximation.* — 2011. — Vol. 34, no. 2. — Pp. 149–179.

338. *Мейханаджян Л.А.* Системы с инверсионным обслуживанием и обобщенным вероятностным приоритетом и их применение к оценке показателей эффективности систем распределенных вычислений // *Канд. диссертация.* — 2016. — 125 с.

339. *Jerri A.* Introduction to integral equations with applications. — N. Y.: John Wiley & Sons, 1999. — 433 pp.

340. *Linz P.* Analytical and numerical methods for Volterra equations. — Philadelphia: SIAM, 1985. — 240 pp.

341. *Полянин А.Д., Манжиров А.В.* Справочник по интегральным уравнениям. — М.: Факториал, 2000. — 384 с.

342. Numerical recipes. 3rd ed. / W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. — N. Y.: Cambridge University Press, 2007. — 1256 pp.

343. *Avi-Itzhak B., Brosh E., Levy H.* SQF: A slowdown queueing fairness measure // *Perform. Eval.* — 2007. — Vol. 64, no. 9. — Pp. 1121–1136.

344. *Harchol-Balter M., Sigman K., Wierman A.* Asymptotic convergence of scheduling policies with respect to slowdown // *Perform. Eval.* — 2002. — Vol. 49, no. 1–4. — Pp. 241–256.

345. *Harchol-Balter M., Sigman K., Wierman A.* Understanding the slowdown of large jobs in an $M/GI/1$ system // *ACM SIGMETRICS Perform. Eval. Rev.* — 2002. — Vol. 30, no. 3. — Pp. 9–11.

346. Numerical inverse Laplace transformation using concentrated matrix exponential distributions / G. Horvth, I. Horvath, S.A. Almousa, M. Telek // *Perform. Eval.* — 2020. — Vol. 137. — P. 102067.

347. *Наумов В.А., Самуйлов К.Е.* О марковских и рациональных потоках случайных событий. II // *Информ. и её примен.* — 2020. — Vol. 14, no. 4. — Pp. 37–46.

348. *Abate J., Whitt W.* A unified framework for numerically inverting Laplace transforms // *INFORMS Journal on Computing.* — 2006. — Vol. 18, no. 4. — Pp. 408–421.

349. *Севастьянов Б.А.* Ветвящиеся процессы. — М.: Наука, 1971. — 436 с.

350. *Ватутин В.А., Зубков А.М.* Ветвящиеся процессы. I // *Итоги науки и техн. Сер. Теор. вероятн. Мат. стат. Теор. кибернет.* — 1985. — Т. 23. — С. 3–67.

351. *Parthasarathy P.R., Vijayalakshmi V.* On quadrature approximation for the busy period density function of an $M/M/1$ queueing system // *American Journal of Mathematical and Management Sciences.* — 1998. — Vol. 18, no. 3–4. — Pp. 277–289.

352. *Liu J., Jiang X., Horiguchi S.* Recursive formula for the moments of queue length in the $M/M/1$ queue // *IEEE Communications Letters.* — 2008. — Vol. 12, no. 9. — Pp. 690–692.

353. *Клейнрок Л.* Теория массового обслуживания. — М.: Машиностроение, 1979. — 432 с.

354. *Harchol-Balter M.* Performance modeling and design of computer systems: Queueing theory in action. — New York: Cambridge University Press, 2013. — 548 pp.

355. *Morozov E.V.* Coupling and stochastic monotonicity of queueing processes. — Петрозаводск: ПетрГУ, 2013. — 72 pp.

356. *Thorrison H.* Coupling, stationarity, and regeneration. — N.Y.: Springer, 2000. — 517 pp.

357. *Lindvall T.* Lectures on the coupling method. — N.Y.: Wiley, 1992. — 257 pp.

358. *Кокс Д., Смит В.* Теория восстановления / Пер. с англ. под ред. Ю.К. Беляева. — М.: Сов. радио, 1967. — 300 с.

359. *Боровков А.А.* Асимптотические методы в теории массового обслуживания. — М.: Наука, 1980. — 381 с.

360. *Генис Я.Г.* Оценки скорости сходимости в предельных теоремах для системы $M/G/1$ при загрузках, близких к единице // *Проблемы передачи информ.* — 1978. — Т. 14, № 1. — С. 103–108.

361. *Азаров Т.А., Хусаинов Я.М.* Предельные теоремы для системы массового обслуживания с абсолютным приоритетом в условиях большой загрузки // *Изв. АН УзССР. Сер. физ.-мат.н.* — 1974. — № 6. — С. 53–55.

362. *Kingman J. F. C.* On queues in heavy traffic // *Journal of the Royal Statistical Society. Series B (Methodological).* — 1962. — Vol. 24, no. 2. — Pp. 383–392.

363. *Limic V.* A LIFO queue in heavy traffic // *Ann. Appl. Probab.* — 2001. — no. 11. — Pp. 301–331.

364. *Limic V.* On the behavior of LIFO preemptive resume queues in heavy traffic // *Electron. Commun. Probab.* — 2000. — no. 5. — Pp. 13–27.

365. *Королев В.Ю.* О сходимости распределений случайных сумм независимых случайных величин к устойчивым законам // *Теория вероятн. и ее примен.* — 1997. — Т. 42, № 4. — С. 818–820.

366. *Королев В.Ю.* О точности нормальной аппроксимации для распределений сумм случайного числа независимых случайных величин // *Теория вероятн. и ее примен.* — 1988. — Т. 33, № 3. — С. 577–581.

367. *Королев В.Ю.* Обобщенные гиперболические распределения как предельные для случайных сумм // *Теория вероятн. и ее примен.* — 2013. — Т. 58, № 1. — С. 117–132.

368. *Григорьева М.Е., Королев В.Ю., Соколов И.А.* Предельная теорема для геометрических сумм независимых неодинаково распределенных случайных величин и ее применение к прогнозированию вероятности катастроф

в неоднородных потоках экстремальных событий // *Информ. и её примен.* — 2013. — Т. 7, № 4. — С. 11–19.

369. *Королев В.Ю., Дорофеева А.В.* Оценки функций концентрации случайных сумм при ослабленных моментных условиях // *Теория вероятн. и ее примен.* — 2017. — Т. 62, № 1. — С. 104–121.

370. *Королев В.Ю., Бенинг В.Е., Шоргин С.Я.* Математические основы теории риска. — М.: Физматлит, 2011. — 591 с.

371. *Gnedenko B.V., Korolev V.Yu.* Random summation: Limit theorems and applications. — Boca Raton: CRC Press, 1996. — 267 pp.

372. *Куглов В.М., Королев В.Ю.* Предельные теоремы для случайных сумм. — М.: изд–во Московского университета, 1990. — 270 с.

373. *Szynal D.* On limit distribution theorem for sums of a random number of random variables appearing in the study of rarefaction of a recurrent process // *Zastosow. Mat.* — 1976. — Vol. 15. — Pp. 277–288.

374. On a class of distributions stable under random summation / L.B. Klebanov, A.V. Kakosyan, S.T. Rachev, G. Temnov // *J. Appl. Probab.* — 2012. — Vol. 49, no. 2. — Pp. 303–318.

375. *Fraser D.* Stein's method for compound geometric approximation // *J. Appl. Probab.* — 2010. — Vol. 47, no. 1. — Pp. 146–156.

376. *Кристоф Г., Монахов М. М., Ульянов В.В.* Разложения Чебышева–Эджворта и Корниша–Фишера второго порядка для распределений статистик, простроенных по выборкам случайного размера // *Зап. научн. сем. ПОМИ.* — 2017. — Т. 466. — С. 167–207.

377. *Хохлов Ю.С.* Псевдоустойчивые распределения и их области притяжения // *Фундамент. и прикл. матем.* — 1996. — Т. 2, № 4. — С. 1143–1154.

378. *Сенатов В.В.* Центральная предельная теорема: Точность аппроксимации и асимптотические разложения. — М.: URSS, 2009. — 352 с.

379. *Белов Е.Г.* Предельные теоремы для сумм случайного числа случайных слагаемых и их применение в теории массового обслуживания // *Канд. диссертация.* — 1989. — 97 с.

380. *Белов Е.Г., Печинкин А.В.* Применение одной предельной теоремы в задачах массового обслуживания // *Труды всесоюзной школы–семинара "Теория массового обслуживания".* — 1981. — С. 157–159.

381. *Печинкин А.В.* Предельные распределения для сумм случайного числа случайных слагаемых // *Канд. диссертация.* — 1973. — 90 с.

382. *Szekli K.* On the concavity of the waiting–time distribution in some $GI/G/1$ queues // *J. Appl. Probab.* — 1986. — Vol. 23, no. 2. — Pp. 555–561.

383. *Daley D.J.* Queueing output processes // *Advances in Applied Probability.* — 1976. — Vol. 8, no. 2. — Pp. 395–415.

384. *Nazarathy Y.* The variance of departure processes: puzzling behavior and open problems // *Queueing Systems.* — 2011. — Vol. 68. — Pp. 385–394.

385. *Brumelle S.L.* Some inequalities for parallel–server queues // *Oper. Res.* — 1971. — Vol. 19, no. 2. — Pp. 402–413.

386. *Arjas E., Lehtonen T.* Approximating many server queues by means of single server queues // *Math. Oper. Res.* — 1978. — Vol. 3, no. 3. — Pp. 205–223.

387. *Павлов А.В.* Дисциплины с приоритетом коротким требованиям и идентичное обслуживание. — Москва: ФГБОУ ВО "МИРЭА — Российский технологический университет", 2014. — 119 с.

388. *Наумов В.А.* Исследование некоторых многофазных систем массового обслуживания // *Канд. диссертация.* — 1978. — 98 с.

389. *Neuts M.F.* Structured stochastic matrices of $M/G/1$ type and their applications. — N.Y.: Marcel Dekker, 1989. — 512 pp.

390. *Наумов В.А.* Об однолинейной системе с ограниченным накопителем и заявками нескольких видов // *Модели систем распределения информации и их анализ.* — 1982. — С. 77–82.

391. *Asmussen S.* Matrix–analytic models and their analysis // *Scandinavian Journal of Statistics.* — 2000. — Vol. 27, no. 2. — Pp. 193–226.

392. *He Q.M.* Fundamentals of matrix–analytic methods. — N.Y.: Springer, 2014. — 349 pp.

393. *Бочаров П.П., Наумов В.А.* О некоторых системах массового обслуживания конечной емкости // *Проблемы передачи информации.* — 1977. — Т. 13, № 4. — С. 96–104.

394. *Наумов В.А., Самуйлов В.А., Гайдамака Ю.В.* Мультипликативные решения конечных цепей Маркова. — М.: РУДН, 2015. — 159 pp.

395. *Lipsky L.* Queueing theory: A linear algebraic approach. 2nd ed. — N.Y.: Springer, 2009. — 548 pp.

396. *Gu M., Eisenstat S.* A stable and efficient algorithm for the rank–one modification of the symmetric eigenproblem // *SIAM Journal on Matrix Analysis and Applications.* — 1994. — Vol. 15, no. 4. — Pp. 1266–1276.

397. *Башарин Г.П., Харкевич М.А., Шнепс М.А.* Массовое обслуживание в телефонии. — М.: Наука., 1968. — 248 с.

398. *Kwong M.K., Zettl A.* New proofs and extensions of Sylvester's and Johnson's inertia theorems to non–hermitian matrices // *Proc. Am. Math. Soc.* — 2011. — Vol. 139. — Pp. 3795–3806.

399. *Marcus M., Minc H.* A aurvey of matrix theory and matrix inequalities. — Courier Corporation, 1992. — 180 pp.

400. *Meyer C.D.* Matrix analysis and applied linear algebra. — SIAM, 2010. — 730 pp.

401. *Wilkinson J.H.* The algebraic eigenvalue problem. — Oxford: Clarendon Press, 1965.

402. *Borges C.F., Gragg W.B.* Divide and conquer for generalized real symmetric definite tridiagonal eigenproblems // *Proceedings of 92nd Shanghai International Numerical Algebra and its Applications Conference.* — 1992. — Pp. 70–76.

403. *Bunch J.R., Nielsen C. P., Sorensen D.C.* Rank–one modification of the symmetric eigenproblem // *Numerische Mathematik.* — 1978. — Vol. 31, no. 1. — Pp. 31–48.

404. *Воеводин В.В., Кузнецов Ю.Л.* Матрицы и вычисления. — М.: Наука, 1984. — 320 с.

405. *Карлин С.* Основы теории случайных процессов. — Москва: Мир, 1971.

406. *Zeifman A.* Upper and lower bounds on the rate of convergence for non-homogeneous birth and death processes // *Stochastic Processes and Their Applications*. — 1995. — Vol. 59, no. 1. — Pp. 157–173.

407. Facilitating numerical solutions of inhomogeneous continuous time markov chains using ergodicity bounds obtained with logarithmic norm method / A. Zeifman, Y. Satin, I. Kovalev et al. // *Mathematics*. — 2021. — Vol. 9, no. 1. — P. 42.

408. *Kempa W.M.* Analytical model of a wireless sensor network (WSN) node operation with a modified threshold–type energy saving mechanism // *Sensors (Basel)*. — 2019. — Vol. 19, no. 14. — P. 3114.

409. *Кац Б. А., Кац Р. А., Швидкая Г. Д.* О выборе дисциплины диспетчеризации по минимаксному критерию // *Автомат. и телемех.* — 1984. — Т. 6. — С. 70–77.

410. *Dong L., Huanyuan S., Dinda P.* Size–based scheduling policies with inaccurate scheduling information // *Proceedings of the 12th Annual International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems*. — 2004. — Pp. 31–38.

411. SRPT scheduling for web servers / M. Harchol-Balter, N. Bansal, B. Schroeder, M. Agrawal // *Proceedings of the 7th International Workshop on Job Scheduling Strategies for Parallel Processing*. — 2001. — Pp. 11–20.

412. Improving peer–to–peer performance through server–side scheduling / Y. Qiao, F. Bustamante, P. Dinda et al. // *ACM Trans. Comput. Syst.* — 2008. — Vol. 26, no. 4. — Pp. 1–30.

413. HFSP: Bringing size–based scheduling to Hadoop / M. Pastorelli, D. Carra, M. Dell'Amico, P. Michiardi // *IEEE Trans. Cloud Comput.* — 2017. — Vol. 5, no. 1. — Pp. 43–56.

414. *Dell'Amico M., Carra D., Michiardi P.* PSBS: Practical size–based scheduling // *IEEE Trans. Computers.* — 2016. — Vol. 65, no. 7. — Pp. 2199–2212.

415. *Баранов А.В., Николаев Д.С.* Применение машинного обучения для прогнозирования времени выполнения суперкомпьютерных заданий // *Программные продукты и системы.* — 2020. — Vol. 33, no. 22. — Pp. 218–228.

416. *Штойян Д.* Качественные свойства и оценки стохастических моделей. — М.: Мир, 1979.

417. *Klefsjo B.* The hnbue and hnwue classes of life distributions // *Naval Research Logistics.* — 1982. — Vol. 29, no. 2. — Pp. 331–344.

418. *Rolski T.* Mean residual life // *Bull. Internat. Statist. Inst.* — 1975. — Vol. 46. — Pp. 266–270.

419. *Klefsjo B.* A useful ageing property based on the Laplace transform // *J. Appl. Probab.* — 1983. — Vol. 20, no. 3. — Pp. 615–626.

420. *Chandhuri G.* Coefficient of variation for the $\mathscr{L}$–class of life distributions // *Commun. Statist. Theory Methods.* — 1993. — Vol. 22, no. 9. — Pp. 12619–2622.

421. *Lin G.D.* Characterizations of the $\mathscr{L}$-class of life distributions // *Statistics and Probability Letters.* — 1998. — Vol. 40. — Pp. 259–266.

422. *Foss S., Korshunov D., Zachary Z.* An introduction to heavy–tailed and subexponential distributions. — New York: Springer-Verlag, 2013. — 166 pp.

423. *Markovich N.* Nonparametric analysis of univariate heavy–tailed data: Research and practice. — Chichester: John Wiley & Sons, 2007. — 336 pp.

424. *Klar B.* A note on the $\mathscr{L}$–class of life distributions // *J. Appl. Probab.* — 2002. — Vol. 39, no. 1. — Pp. 11–19.

425. *Fang K., Kotz S., Ng K.* Symmetric multivariate and related distributions. — London: Chapman & Hall, 1990. — 220 pp.

426. *Vanegas L.H., Paula G.A.* Log–symmetric distributions: Statistical properties and parameter estimation // *Journal of the American Statistical Association.* — 2016. — Vol. 30, no. 2. — Pp. 196–220.

427. *Walker S.G.* On a lower bound for the Jensen inequality // *SIAM J. Math. Anal.* — 2014. — Vol. 46, no. 5. — Pp. 3151–3157.

428. *de Bruijn N. G.* Asymptotic methods in analysis. — New York: Dover Publications Inc., 1981. — 224 pp.

429. *Зубков А.М.* Двусторонние неравенства для преобразований Лапласа // *Теория вероятн. и ее применен.* — 1998. — Т. 43, № 4. — С. 767–773.

430. *Hu C., Lin G.* Some inequalities for Laplace transforms // *J. Math. Anal. Appl.* — 2008. — Vol. 340. — Pp. 675–686.

431. *Barlow R.E., Marshall A.W.* Bounds for distributions with monotone hazard rate // *Ann. Math. Statist.* — 1964. — Vol. 35, no. 3. — Pp. 1234–1257.

432. *Волченкова И.В., Клебанов Л.Б.* О характеризации распределений симметрично зависимых случайных величин // *Зап. научн. сем. ПОМИ.* — 2017. — Т. 466. — С. 81–95.

433. *Khokhlov Yu. S., Lukashenko O.V., Morozov E.V.* On a lower asymptotic bound of the overflow probability in a fluid queue with a heterogeneous fractional input // *Journal of Mathematical Sciences.* — 2019. — Vol. 237. — Pp. 667—-672.

434. *Пугачев В.С., Синицин И.Н.* Стохастические дифференциальные системы. Анализ и фильтрация. — М.: Наука, 1990. — 642 с.

435. *Фельдбаум А.А.* Основы теории оптимальных автоматических систем. — М.: Наука. Гл. ред. физ.-мат.лит., 1966. — 624 с.

436. *Панков А.Р.* Оптимизация алгоритмов оценивания параметров стохастических систем в условиях неопределенности // *Автомат. и телемех.* — 1985. — Т. 7. — С. 110–120.

437. *Синицин И.Н., Синицин В.И.* Лекции по теории нормальной и эллипсоидальной аппроксимации распределений в стохастических системах. — М.: ТОРУС ПРЕСС, 2013. — 488 с.

438. *Tsafrir D., Etsion Y., Feitelson D.* Modeling user runtime estimates // *Lecture Notes in Computer Science.* — 2005. — Vol. 3834. — Pp. 1–35.

439. *Шерр А.* Анализ вычислительных систем с разделением времени. — М.: Мир, 1987. — 152 с.

440. *Arlitt M., Jin T.* Workload characterization of the 1998 World Cup web site // *IEEE Network.* — 2000. — Vol. 14, no. 3. — Pp. 30–37.

441. *Markovitch N.M., Krieger U.R.* On–line estimation of heavy–tailed traffic characteristics in Web data mining // *Teletraffic Science and Engineering.* — 2003. — Vol. 5. — Pp. 571–580.

442. *Harchol-Balter M.* Open problems in queueing theory inspired by datacenter computing // *Queueing Systems.* — 2021. — Vol. 97. — Pp. 3–37.

443. *Королев В.Ю., Соколов И.А.* Математические модели неоднородных потоков экстремальных событий. — М.: ТОРУС ПРЕСС, 2008. — 192 с.

444. *Belkin B.* First passage to a general threshold for a process corresponding to sampling at Poisson times // *J. Appl. Probab.* — 1971. — Vol. 8, no. 3. — Pp. 573–588.

445. *Tang Y., Lam Y.* A δ-shock maintenance model for a deteriorating system // *European Journal of Operational Research.* — 2006. — Vol. 168. — Pp. 541–556.

446. *Kirmani S.N.U.A., Wesolowski J.* Time spent below a random threshold by a Poisson driven sequence of observations // *J. Appl. Probab.* — 2003. — Vol. 40, no. 3. — Pp. 807–814.

447. *Lee M.T., Whitmore G.A.* Stochastic processes directed by randomized time // *J. Appl. Probab.* — 1993. — Vol. 30. — Pp. 302–314.

448. *Mallor F., Omey E.* Shocks, runs, and random sums: Asymptotic behavior of the tail of the distribution function // *Journal of Mathematical Sciences.* — 2002. — Vol. 111, no. 2. — Pp. 3559–3565.

449. *Mallor F., Omey E.* Shocks, runs and random sums // *J. Appl. Probab.* — 2001. — Vol. 38, no. 2. — Pp. 438–448.

450. Asymptotic behavior of total times for jobs that must start over if a failure occurs / S. Asmussen, P. Fiorini, L. Lipsky et al. // *Math. Oper. Res.* — 2008. — Vol. 33, no. 4. — Pp. 932–944.

451. *Wesolowski J., Ahsanullah M.* Distributional properties of exceedance statistics // *Annals of the Institute of Statistical Mathematics.* — 1998. — Vol. 50. — Pp. 543–565.

452. *Gut A., Huesler J.* Extreme shock models // *Extremes.* — 1999. — Vol. 2. — Pp. 295–307.

453. *Thangaraj V., Stanley A.D.J.* General shock models with random threshold // *Optimization.* — 1990. — Vol. 21, no. 4. — Pp. 629–636.

454. *Shanthikumar J. G., Sumita U.* Shock models associated with correlated renewal sequences // *J. Appl. Probab.* — 1983. — Vol. 20, no. 3. — Pp. 600–614.

455. *Galambos J., Hagwood C.* An unreliable server characterization of the exponential distribution // *J. Appl. Probab.* — 1994. — Vol. 31, no. 1. — Pp. 274–279.

456. *Brandt A., Brandt M.* A sample path relation for the sojourn times in $G/G/1PS$ // *Queueing Systems.* — 2006. — Vol. 52. — Pp. 281–286.

457. *Whitt W.* The Marshall and Stoyan bounds for $IMRL/G/1$ queues are tight // *Oper. Res. Lett.* — 1982. — Vol. 1, no. 6. — Pp. 209–213.

458. *Sigman K.* Exact simulation of the stationary distribution of the FIFO $M/G/c$ queue: the general case for $\rho < c$ // *Queueing Systems.* — 2012. — no. 70. — Pp. 37–43.

459. *Xiong Y., Murdoch D., Stanford D.* Perfect and nearly perfect sampling of work–conserving queues // *Queueing Systems.* — 2015. — Vol. 80. — Pp. 197–222.

460. *Пешкова И.В., Румянцев А.С.* Методы регенеративного моделирования для анализа многосерверных систем обслуживания // *Труды Карельского научого центра РАН.* — 2018. — № 7. — С. 68–82.

461. *Morozov E.V., Steyart B.* Stability analysis of regenerative queueing models. — Cham: Springer, 2021. — 185 pp.

462. *Morozov E. V.* The tightness in the ergodic analysis of regenerative queueing processes // *Queueing Systems.* — 1997. — no. 27. — Pp. 179–203.

463. *Гнеденко Б.В.* Курс теории вероятностей. — М.: ГИТТЛ, 1954. — 411 с.

464. *Chukova S., Dimitrov B.* On distributions having the almost–lack–of–memory property // *J. Appl. Probab.* — 1992. — Vol. 29, no. 3. — Pp. 691–698.

465. *Whitt W.* Approximations for the $GI/G/m$ queue // *Production and Operations Management.* — 1993. — Vol. 2, no. 3. — Pp. 114–161.

466. *Wolff R.W., Wang C.* Idle period approximations and bounds for the $GI/G/1$ queue // *Advances in Applied Probability.* — 2003. — Vol. 35, no. 3. — Pp. 773–792.

467. *Chen Y., Whitt W.* Set–valued performance approximations for the $GI/GI/K$ queue given partial information // *Probability in the Engineering and Informational Sciences.* — 2020. — Pp. 1–23.

468. *Delbrouck L.E.N.* Approximations for compound geometric distributions with applications in queueing theory // *J. Appl. Probab.* — 1978. — Vol. 15, no. 1. — Pp. 202–208.

469. *Filippopoulos D., Karatza H.* An $M/M/2$ parallel system model with pure space sharing among rigid jobs // *Math. Comput. Model.* — 2007. — Vol. 45. — Pp. 491–530.

470. *Морозов Е.В., Румянцев А.С.* Модели многосерверных систем для анализа вычислительного кластера // *Труды Карельского научного центра РАН.* — 2011. — № 5. — С. 75–85.

471. *Brill P.H., Green L.* Queues in which customers receive simultaneous service from a random number of servers: A system point approach // *Management Science.* — 1984. — Vol. 30, no. 1. — Pp. 51–68.

472. *Kim S.* $M/M/s$ queueing system where customers demand multiple server use // *PhD Thesis.* — 1979.

473. *Rumyantsev A., Morozov E.* Stability criterion of a multiserver model with simultaneous service // *Ann. Oper. Res.* — 2015. — Pp. 1–11.

474. *Chakravarthy S.R., Karatza H.D.* Two–server parallel system with pure space sharing and Markovian arrivals // *Computers and Opetaions Research.* — 2013. — Vol. 40, no. 1. — Pp. 510–519.

475. *Nakagawa T., Osaki Sh.* The discrete Weibull distribution // *IEEE Trans. Reliab.* — 1975. — Vol. 24. — Pp. 300–301.

476. *Schassberger R.* Wartenschlangen. — Wien/New York: Springer-Verlag, 1973. — 228 pp.

477. *Kolodziej J.* Evolutionary hierarchical multi–criteria metaheuristics for scheduling in large–scale grid systems. — 2012. — Vol. 419. — 191 pp.

478. *Yang X.-S.* Nature–inspired optimization algorithms. — 1st edition. — NLD: Elsevier Science Publishers B.V., 2014. — 300 pp.

479. *Карацуба Е.А.* Быстрое вычисление константы Каталана через приближения, полученные преобразованиями типа куммеровских // *Дискрет. матем.* — 2013. — Т. 25, № 4. — С. 74–87.

480. Age of information: An introduction and survey / R. Yates, Y. Sun, D. Brown et al. // *IEEE Journal on Selected Areas in Communications.* — 2021. — Vol. 39, no. 5. — Pp. 1183–1210.

481. *Kriouile S.* Asymptotically optimal scheduling schemes for large networks // *PhD Thesis.* — 2021. — 175 pp.

482. *Kosta A.* Age of information aware communication systems: Modeling and performance analysis // *PhD Thesis.* — 2020. — 51 pp.

483. *Lakshminarayana N.B., Lee J., Kim H.* Age based scheduling for asymmetric multiprocessors // *Proceedings of the Conference on High Performance Computing Networking, Storage and Analysis.* — 2009. — Pp. 1–12.

484. *He Q., Dan G., Fodor V.* Joint assignment and scheduling for minimizing age of correlated information // *IEEE/ACM Transactions on Networking.* — 2019. — Vol. 27, no. 5. — Pp. 1887–1900.

485. AoI scheduling with maximum thresholds / C. Li, S. Li, Y. Chen et al. // *Proceedings of the IEEE Conference on Computer Communications.* — 2020. — Pp. 436–445.

486. Алгоритмы параллельного выполнения заданий сервисных приложений в распределенной среде / В. А. Дьячков, В. Н. Захаров, В. А. Козмидиади и др. // *Системы и средства информ.* — 2008. — Т. 18. — С. 49–70.

487. *Зацаринный А.А., Ионенков Ю.С., Сучков А.П.* Некоторые аспекты оценки эффективности облачных технологий // *Системы и средства информ.* — 2018. — Т. 28, № 3. — С. 104–117.

488. *Колпаков Р. М., Посыпкин М. А.* Об эффективной стратегии распараллеливания при решении задач о сумме подмножеств методом ветвей и границ // *Дискрет. матем.* — 2019. — Т. 31, № 4. — С. 20–37.

489. *Абрамов С. М., Кузнецов А. А., Роганов В.А.* Кроссплатформенная версия Т–системы с открытой архитекурой // *Выч. мет. программирование.* — 2007. — Т. 8, № 1. — С. 18–23.

490. *Ahn J., Han T.* An analytical method for parallelization of recursive functions // *Parallel Processing Letters.* — 2000. — Vol. 10, no. 04. — Pp. 359–370.

491. *Schervish M.J.* Applications of parallel computation to statistical inference // *Journal of the American Statistical Association.* — 1988. — Vol. 83, no. 404. — Pp. 976–983.

492. Метод распараллеливания прогонки на гибридных ЭВМ / А.Н. Быков, А.М. Ерофеев, Е.А. Сизов, А.А. Федоров // *Выч. мет. программирование.* — 2013. — Т. 14, № 2. — С. 43–47.

493. *Штейнберг Б.Я.* Распараллеливание рекуррентных циклов с условными операторами // *Автомат. и телемех.* — 1995. — Т. 9. — С. 176–184.

494. *Головкин Б.А.* Методы и средства параллельной обработки информации // *Итоги науки и техн. Сер. Теор. вероятн. Мат. стат. Теор. Кибернет.* — 1979. — Т. 17. — С. 85–193.

495. *Воеводин В.В.* Параллелизм в сложных программных комплексах (почему сложно создавать эффективные прикладные пакеты) // *Чебышевский сб.* — 2017. — Т. 18, № 3. — С. 188–201.

496. Optimizing for tail sojourn times of cloud clusters / M. Bjorkqvist, N. Gautam, L. Chen, W. Binder // *IEEE Transactions on Cloud Computing.* — 2018. — Vol. 6. — Pp. 156–167.

497. Beyond processor sharing / S. Aalto, U. Ayesta, S. Borst et al. // *SIGMETRICS Perform. Eval. Rev.* — 2007. — Vol. 34, no. 4. — Pp. 36–43.

498. *Vlasiou M.* Lindley–type recursions // *PhD Thesis.* — 2006. — 201 pp.

499. *Мистрюков А.В., Ушаков В.Г.* Достаточные условия эргодичности приоритетных систем массового обслуживания // *Информ. и её примен.* — 2018. — Т. 12, № 2. — С. 24–28.

500. *Kin W., Chan V.* Generalized Lindley–type recursive representations for multiserver tandem queues with blocking // *ACM Transactions on Modeling and Computer Simulation.* — 2010. — Vol. 20, no. 4. — Pp. 1–19.

501. *Karpelevich F.I., Kelbert M.Ya., Suhov Yu. M.* Higher–order Lindley equations // *Stochastic Processes and their Applications.* — 1994. — Vol. 53. — Pp. 65–96.

502. *Krivulin N.K.* A recursive equations based representation for the $G/G/m$ queue // *Applied Mathematics Letters.* — 1994. — Vol. 7, no. 3. — Pp. 73–77.

503. *Liu Z., Baccelli F.* Generalized precedence–based queueing systems // *Math. Oper. Res.* — 1992. — Vol. 17, no. 3. — Pp. 615–639.

504. *Baccelli F., Liu Z.* On the execution of parallel programs on multiprocessor systems — a queueing theory approach // *J. ACM.* — 1990. — Vol. 37. — Pp. 373–414.

505. *Wu D., Takagi H.* Processor–sharing and random-service queues with semi–-markovian arrivals // *J. Appl. Probab.* — 2005. — Vol. 42, no. 2. — Pp. 478–490.

506. *Borkar V., Pattathil S.* Whittle indexability in egalitarian processor sharing systems // *Annals of Operations Research.* — 2017. — Pp. 1–21.

507. *Боровков А.А.* Эргодичность и устойчивость случайных процессов. — Москва: Эдиториал УРСС, 1999. — 440 с.

508. *Прохоров Ю.В., Розанов Ю.А.* Теория вероятностей. — Москва: Наука, 1973. — 496 с.

509. *Лисейкин В.Д.* Обзор методов построения структурных адаптивных сеток // *Ж. вычисл. матем. и матем. физ.* — 1996. — Т. 36, № 1. — С. 3–41.

510. Space discretization methods / B. Courbet, C. Benoit, V. Couaillier et al. // *AerospaceLab.* — 2011. — Vol. 45, no. 8. — Pp. 1–14.

511. *Boulle M.K.* A statistical discretization method of continuous attributes // *Machine Learning.* — 2004. — Vol. 55. — Pp. 53–69.

512. *Uther W., Veloso M.* Tree based discretization for continuous state space reinforcement learning // *Proceedings of the 15th National Conference on Artificial Intelligence.* — 1998. — Pp. 769–774.

513. *Robert C., Guihenneuc-Jouyaux C.* Discretization of continuous Markov chains and Markov chain Monte–Carlo convergence assessment // *Journal of the American Statistical Association.* — 1998. — Vol. 93, no. 443. — Pp. 1055–1067.

514. *Евтушенко Ю.Г.* Численный метод поиска глобального экстремума функций (перебор на неравномерной сетке) // *Ж. вычисл. матем. и матем. физ.* — 1971. — Т. 11, № 6. — С. 1390–1403.

515. *Гаранжа В.А., Кудрявцева Л.Н., Цветкова В.О.* Построение гибридных расчетных сеток Вороного. Алгоритмы и нерешенные проблемы // *Ж. вычисл. матем. и матем. физ.* — 2019. — Vol. 59, no. 12. — Pp. 2024–2044.

516. *Whitt W.* Approximating a point process by a renewal process: Two basic methods // *Oper. Res.* — 1982. — Vol. 30, no. 1. — Pp. 125–147.

517. *Albin S.* Approximating a point process by a renewal process, II: Superposition arrival processes to queues // *Oper. Res.* — 1984. — Vol. 32, no. 5. — Pp. 1133–1162.

518. *Sassen S.A.E., Tijms H.C., Nobel R.D.* A heuristic rule for routing customers to parallel servers // *Statistica Neerlandica.* — 1997. — Vol. 51, no. 1. — Pp. 107–121.

519. *Feitelson D., Tsafrir D., Krakov D.* Experience with using the parallel workloads archive // *J. Parallel Distr. Com.* — 2014. — Vol. 74, no. 10. — Pp. 2967–2982.

520. *Feitelson D., Rudolph L.* Metrics and benchmarking for parallel job scheduling // *Lecture Notes in Computer Science.* — 1998. — Vol. 1459. — Pp. 1–24.

521. *Buchholz P., Kriege J.* Fitting correlated arrival and service times and related queueing performance // *Queueing Systems.* — 2017. — Vol. 85. — Pp. 337–359.

522. *Melamed B., Whitt W.* On arrivals that see time averages // *Oper. Res.* — 1990. — Vol. 38, no. 1. — Pp. 156–172.

523. *Glynn P., Melamed B., Whitt W.* Estimating customer and time averages // *Oper. Res.* — 1993. — Vol. 41, no. 2. — Pp. 400–408.

524. *Krishnan K.R., Ott T.J.* State–dependent routing for telephone traffic: Theory and results // *Proceedings of the IEEE Conference on Decision and Control.* — 1986. — Vol. 25. — Pp. 2124–2128.

525. *Krishnan K.R.* Joining the right queue: A state–dependent decision rule // *IEEE Transactions on Automatic Control.* — 1990. — Vol. 35, no. 1. — Pp. 104–108.

526. *Samuelsson S.G., Hyytiä E.* Applying reinforcement learning to basic routing problem // *Lecture Notes in Computer Science.* — 2018. — Vol. 10932. — Pp. 238–249.

527. *Chen H., Marden J.R., Wierman A.* On the impact of heterogeneity and back–end scheduling in load balancing designs // *IEEE INFOCOM.* — 2009. — Pp. 2267–2275.

528. *Козлов В.В.* Оптимальная дисциплина обслуживания для систем массового обслуживания // *Сб. Научные труды Кубанского унив–та.* — 1977. — № 247. — С. 33–37.

529. *Schrage L.E., Miller L.W.* The queue $M/G/1$ with the shortest remaining processing time discipline // *Oper. Res.* — 1966. — Vol. 14. — Pp. 670–684.

530. *Печинкин А.В.* О верхней и нижней оценках средней очереди в системе с дисциплиой Шраге // *Техника средств связи. Сер. СС.* — 1980. — № 3. — С. 24–28.

531. *Kimura T.* Approximating the mean waiting time in the $GI/G/s$ queue // *J. Oper. Res. Soc.* — 1991. — Vol. 42, no. 11. — Pp. 959–970.

532. *Kimura T.* Refining Cosmetatos' approximation for the mean waiting time in the $M/D/s$ queue // *J. Oper. Res. Soc.* — 1991. — Vol. 42, no. 7. — Pp. 595–603.

533. *Kimura T.* Approximations for multi–server queues: System interpolations // *Queueing Systems.* — 1994. — Vol. 17. — Pp. 347–382.

534. *Guo X, Lu Y., Squillante M.* Optimal probabilistic routing in distributed parallel queues // *ACM SIGMETRICS Perform. Eval. Rev.* — 2004. — Vol. 32, no. 2. — Pp. 53–54.

535. *Whitt W.* Approximating a point process by a renewal process: The view through a queue. An indirect approach // *Management Science.* — 1981. — Vol. 27, no. 6. — Pp. 619–636.

536. Queueing networks and Markov chains: Modeling and performance evaluation with computer science applications / G. Bolch, S. Greiner, H. de Meer, K.S. Trivedi. — 2nd edition. — Hoboken, NJ: John Wiley & Sons, 2006. — 896 pp.

537. *Kraemer W., Langenbach-Belz M.* Approximate formulae for the delay in the queueing system $GI/G/1$ // *Proceedings of the 8th International Teletraffic Congress.* — 1976. — Pp. 2351–8.

538. *Isham V.* Dependent thinning of point processes // *J. Appl. Probab.* — 1980. — Vol. 17, no. 4. — Pp. 987–995.

539. *Ivnitskii V., Moiseev A.* New results for a thinned renewal process // *Communications in Computer and Information Science.* — 2016. — Vol. 638. — Pp. 132–139.

540. *Justicz J., Scheinerman E.R., Winkler P.M.* Random intervals // *The American Mathematical Monthly.* — 1990. — Vol. 97, no. 10. — Pp. 881–889.

541. *Stevens W.L.* Solution to a geometrical problem in probability // *Annals of Human Genetics.* — 2011. — Vol. 9, no. 4. — Pp. 315–320.

542. *Barton D.E., David F.N.* Some notes on ordered random intervals // *Journal of the Royal Statistical Society. Series B (Methodological).* — 1956. — Vol. 18, no. 1. — Pp. 79–94.

543. Scheduling split intervals / R. Bar-Yehuda, M. Halldorsson, J. Naor et al. // *Proceedings of the thirteenth annual ACM-SIAM symposium on Discrete algorithms.* — 2002. — Pp. 732–741.

544. *Рыбко А.Н., Шлосман С.Б.* Пуассоновская гипотеза: компбинаторный аспект // *Проблемы передачи информ.* — 2005. — Т. 41, № 3. — С. 51–57.

545. *Феллер В.* Введение в теорию вероятностей и ее приложения. Т. 2. — М.: Мир, 1967. — 752 с.

546. *Asmussen S.* Light traffic equivalence in single–server queues // *The Annals of Applied Probability.* — 1992. — Vol. 2, no. 3. — Pp. 555–574.

547. *Daley D.J., Rolski T.* A light traffic approximation for a single–server queue // *Math. Oper. Res.* — 1984. — Vol. 9, no. 4. — Pp. 624–628.

548. *Босов А. В.* Управление выходом стохастической дифференциальной системы по квадратичному критерию. V. Случай неполной информации о состоянии // *Информ. и её примен.* — 2020. — Т. 14, № 2. — С. 19–25.