Министерство образования и науки Российской Федерации

УНИВЕРСИТЕТ ИТМО

Т.И. Алиев

ЗАДАЧИ И МЕТОДЫ ПРОЕКТИРОВАНИЯ ДИСКРЕТНЫХ СИСТЕМ

Учебное пособие

УНИВЕРСИТЕТ ИТМО

Санкт-Петербург

2015

Алиев Т.И. Задачи и методы проектирования дискретных систем. – СПб: Университет ИТМО, 2015. – 127 с.

В пособии излагаются теоретические вопросы, связанные с математическими моделями технических систем с дискретным характером функционирования и применением аналитических методов расчёта характеристик функционирования систем для анализа эффективности различных вариантов структурно-функциональной организации исследуемых систем и решения задач проектирования с использованием методов оптимизации. Задачи оптимизации рассматриваются на примере относительно простых математических моделей, в качестве которых используются базовые и сетевые модели массового обслуживания.

Учебное пособие предназначено для студентов, обучающихся по магистерским программам направления «Информатика и вычислительная техника», и может быть полезным для выпускников при подготовке выпускных квалификационных работ, в которых требуется выполнить проектирование и исследование систем, представляемых моделями массового обслуживания, например, компьютерной сети или её фрагмента – вычислительной системы (сервера), узла или канала передачи данных.

Рекомендовано к печати Ученым советом факультета компьютерных технологий и управления, 8 декабря 2015 года, протокол № 10.

УНИВЕРСИТЕТ ИТМО

Университет ИТМО – ведущий вуз России в области информационных и фотонных технологий, один из немногих российских вузов, получивших в 2009 году статус национального исследовательского университета. С 2013 года Университет ИТМО – участник программы повышения конкурентоспособности российских университетов среди ведущих мировых научно-образовательных центров, известной как проект «5 в 100». Цель Университета ИТМО – становление исследовательского университета мирового уровня, предпринимательского по типу, ориентированного на интернационализацию всех направлений деятельности.

Содержание

1. Введение 7
2. Основы проектирования дискретных систем 9
   - 2.1. Примеры и модели дискретных систем 9
   - 2.2. Общая постановка задачи проектирования 11
   - 2.3. Типовые подходы к проектированию дискретных систем 13
3. Краткая характеристика методов проектирования 14
   - 3.1. Аналитические методы 15
     - 3.1.1. Метод средних значений 15
     - 3.1.2. Метод марковских случайных процессов 16
   - 3.2. Имитационные методы и средства 18
   - 3.3. Комбинированный подход 19
4. Проектирование систем на основе базовых моделей с однородной нагрузкой 19
   - 4.1. Постановка задачи проектирования систем с однородной нагрузкой 19
   - 4.2. Системы с одним устройством и накопителем неограниченной ёмкости 20
     - 4.2.1. Расчёт характеристик систем с накопителем неограниченной ёмкости 21
     - 4.2.2. Определение нижней границы производительности устройства 25
     - 4.2.3. Определение минимальной производительности устройства с учётом ограничения на среднее время пребывания заявок 26
     - 4.2.4. Определение оптимальной производительности устройства 27
     - 4.2.5. Определение оптимальной производительности устройства с учетом ёмкости накопителя 30
     - 4.2.6. Определение производительности устройства с учётом ограничения на время пребывания 33
     - 4.2.7. Оценка ёмкости накопителя 34
     - 4.2.8. Определение допустимой нагрузки при заданной производительности устройства и ограничении на время пребывания заявок в системе 36
   - 4.3. Системы с одним устройством и накопителем ограниченной ёмкости 36
     - 4.3.1. Расчёт характеристик систем с накопителем ограниченной ёмкости 37

4.3.2. Анализ свойств системы с накопителем ограниченной ёмкости 39
4.3.3. Проектирование систем с накопителем ограниченной ёмкости 42
4.3.1. Сравнительный анализ систем с накопителями ограниченной и неограниченной ёмкости 45
4.4. Системы с несколькими устройствами и накопителем неограниченной ёмкости 47
4.4.1. Расчёт характеристик систем с индивидуальными накопителями неограниченной ёмкости 48
4.4.2. Расчёт характеристик систем с общим накопителем неограниченной ёмкости 49
4.4.3. Определение минимального количества устройств в системе 52
4.4.4. Проблема проектирования систем с несколькими устройствами 53
4.5. Системы с несколькими устройствами и накопителем ограниченной ёмкости 54
4.5.1. Расчёт характеристик систем с несколькими устройствами и индивидуальными накопителями ограниченной ёмкости 54
4.5.2. Расчёт характеристик систем с несколькими устройствами и общим накопителем ограниченной ёмкости 55
4.5.3. Проектирование систем с несколькими устройствами и накопителями ограниченной ёмкости 57
5. Проектирование систем на основе базовых моделей с неоднородной нагрузкой 60
5.1. Системы с одним устройством и неоднородной нагрузкой 60
5.1.1. Расчёт характеристик систем с бесприоритетным обслуживанием заявок 62
5.1.2. Расчёт характеристик систем с относительными приоритетами 63
5.1.3. Расчёт характеристик систем с абсолютными приоритетами 64
5.1.4. Законы сохранения 65
5.1.5. Критерии эффективности для решения задачи распределения приоритетов 66
5.2. Системы со смешанными приоритетами 70
5.2.1. Способы описания дисциплин со смешанными приоритетами 70
5.2.2. Корректность матрицы приоритетов 73
5.2.3. Правила построения корректных канонических матриц приоритетов 75

5.2.4. Расчёт характеристик систем со смешанными приоритетами ... 76
5.2.5. Постановка задачи проектирования систем со смешанными приоритетами ... 76
5.3. Функциональное проектирование систем реального времени ... 77
5.3.1. Постановка задачи функционального проектирования ... 78
5.3.2. Оценка нижней границы производительности процессора ... 80
5.3.3. Синтез дисциплины обслуживания ... 83
5.3.4. Определение оптимальной производительности системы ... 87
5.3.5. Проектирование систем реального времени с вероятностными ограничениями ... 89
5.4. Системы с несколькими устройствами и неоднородной нагрузкой ... 90
5.4.1. Расчёт характеристик систем с бесприоритетным обслуживанием заявок ... 91
5.4.2. Расчёт характеристик систем с относительными приоритетами ... 92
5.4.3. Расчёт характеристик систем с абсолютными приоритетами ... 93
5.4.4. Задачи проектирования систем с несколькими устройствами и неоднородной нагрузкой ... 93
6. Проектирование систем с использованием сетевых моделей ... 94
6.1. Проектирование систем на основе разомкнутых сетевых моделей ... 94
6.1.1. Описание разомкнутых сетевых моделей ... 94
6.1.2. Расчёт коэффициентов передач и интенсивностей потоков заявок в узлах сетевой модели ... 96
6.1.3. Расчёт узловых характеристик разомкнутых моделей ... 97
6.1.4. Расчёт сетевых характеристик разомкнутых моделей ... 98
6.1.5. Способы ликвидации перегрузок в системе и определение допустимой нагрузки в сети ... 98
6.1.6. Задачи проектирования систем на основе разомкнутых сетевых моделей ... 100
6.1.7. Проектирование системы на основе разомкнутой сетевой модели и составного критерия эффективности ... 102
6.1.8. Проектирование системы заданной стоимости на основе разомкнутой сетевой модели ... 104
6.1.9. Проектирование системы с заданным временем пребывания на основе разомкнутой сетевой модели ... 106
6.2. Проектирование систем на основе замкнутых сетевых моделей ... 107
6.2.1. Описание замкнутых сетевых моделей ... 108

6.2.2. Расчёт коэффициентов передач в узлах замкнутой модели 109
6.2.3. Расчёт характеристик замкнутой сетевой модели 110
6.2.4. Рекомендации по проектированию систем на основе замкнутых сетевых моделей 112
6.2.5. Задачи проектирования систем на основе замкнутых сетевых моделей 116
6.2.6. Проектирование системы на основе замкнутой модели и составного критерия эффективности 117
6.2.7. Проектирование системы заданной стоимости на основе замкнутой модели 119
6.2.8. Проектирование системы заданной производительности на основе замкнутой модели 120
7. Вопросы для самопроверки 123
8. Список литературы 128

Введение

Пособие является продолжением пособия «Основы проектирования систем» [1], в котором рассмотрены общие принципы проектирования систем, математические модели дискретных систем, представляемые в виде систем и сетей массового обслуживания, используемых в качестве моделей реальных систем с дискретным характером функционирования. Примерами таких систем могут служить вычислительные системы, компьютерные сети и их компоненты, модели которых представлены в последнем разделе пособия [1]. Естественно, что материал, излагаемый в данном пособии, предполагает знакомство с материалом пособия «Основы проектирования систем».

Цель данного пособия – дать общее представление о задачах проектирования систем с использованием математических моделей на примере моделей массового обслуживания, широко используемых для исследования дискретных систем различного назначения, в том числе, таких как вычислительные системы.

В пособии излагаются теоретические вопросы, связанные с математическими моделями технических систем с дискретным характером функционирования и применением аналитических методов расчёта характеристик функционирования систем для анализа эффективности различных вариантов структурно-функциональной организации исследуемых систем и решения задач системотехнического проектирования с использованием методов оптимизации. При этом задачи оптимизации рассматриваются абстрактно, применительно к сравнительно простым математическим моделям, в качестве которых используются базовые и сетевые модели массового обслуживания, безотносительно реальных систем. Читателю предлагается попытаться самостоятельно применить рассматриваемые методы к системам, модели которых представлены в четвёртом разделе пособия [1], выполнить анализ полученных результатов и сформулировать выводы и рекомендации по проектированию таких систем.

При проектировании реальных систем может потребоваться применение гораздо более сложных моделей и разработка гораздо более громоздких методов, основанных на комбинированном подходе, предполагающем совместное применение аналитических и имитационных методов и средств, которые могут быть реализованы в рамках построения автоматизированных систем проектирования.

Пособие содержит три основных раздела, связанных с проектированием систем, в которых рассматриваются аналитические методы расчёта характеристик функционирования систем и базирующиеся на них задачи оптимизации с использованием трёх видов моделей:

1) базовых моделей систем с одним или несколькими параллельными устройствами и однородной нагрузкой;

2) базовых моделей систем с одним или несколькими параллельными устройствами и неоднородной нагрузкой;

3) сетевых моделей с однородной нагрузкой.

Каждый из этих разделов предваряется кратким изложением аналитических методов расчёта соответствующих моделей, на которых базируются методы оптимизации или формируются рекомендации по проектированию систем, представляемых этими моделями. Более детально с рассматриваемыми в пособии математическими моделями, а также с аналитическими и имитационными методами расчёта характеристик их функционирования можно ознакомиться, например, в [2].

Список литературы содержит ограниченный перечень литературных источников, которые в той или иной мере использовались при написании пособия или могут быть рекомендованы для получения дополнительной информации. Этот список включает учебные пособия и монографии, которые условно можно разбить на две группы, содержащие материал:

- по принципам моделирования дискретных систем и математическим методам расчёта характеристик функционирования систем и сетей массового обслуживания [1-3];
- по моделям и методам исследования вычислительных систем и компьютерных сетей разных классов [4-8].

Учебное пособие предназначено для студентов, обучающихся по магистерским программам направления «Информатика и вычислительная техника», и может быть полезным для выпускников при подготовке выпускных квалификационных работ, в которых требуется выполнить проектирование и исследование систем, представляемых моделями массового обслуживания, например, компьютерной сети или её фрагмента – вычислительной системы (сервера), узла или канала передачи данных.

Основы проектирования дискретных систем

2.1. Примеры и модели дискретных систем

Большое многообразие систем, обладающих разными специфическими особенностями, обусловливает необходимость разработки различных подходов, принципов, методов и средств проектирования, каждый из которых применим к определенному классу систем.

В качестве объектов проектирования ниже рассматриваются дискретные системы, в которых процесс функционирования носит дискретный характер, т.е. переход системы из одного состояния в другое осуществляется скачкообразно, причем множество состояний счетно, т.е. может быть пронумеровано.

Примерами дискретных систем служат системы обслуживания, в которые поступают некоторые подвижные объекты, называемые запросами или заявками, требующие обслуживания (обработки). К таким системам относятся, например, технические системы (информационно-вычислительные и информационно-управляющие системы, компьютерные сети и сетевые устройства – маршрутизаторы, коммутаторы и каналы связи), производственные системы, системы обслуживания клиентов [1, 3, 4, 6]. Системы обслуживания характеризуются наличием структурно-функциональной организации, ориентированной на эффективное выполнение некоторой работы, описываемой в виде нагрузки, реализуемой в системе.

Важной особенностью таких систем является случайный характер функционирования, обусловленный наличием множества случайных факторов в процессе формирования и реализации нагрузки.

Для количественного описания качества функционирования дискретных систем различных классов используется множество показателей эффективности, которые можно разбить на две группы:

1) временны*е* показатели эффективности, описывающие различного рода задержки при обработке и передачи запросов в системе;

2) безразмерные показатели эффективности, определяющие, например, нагрузку и загрузку системы, вероятность потери запросов и т.п.

Абстрагируясь от реальных систем, дискретную систему обслуживания можно представить в виде совокупности некоторых ресурсов (оборудования, устройств), которые распределяются между запросами (заявками, задачами, деталями и т.п.), поступающими из внешней среды или формирующимися внутри системы, которые образуют нагрузку в системе (рисунок 1). Запросы, поступившие в систему и заставшие некоторый ресурс системы занятым обработкой (обслуживанием) ранее поступивших запросов, могут ожидать освобождения ресурса в накопителях, образуя очередь запросов. Поскольку ёмкость накопителей в реальных системах обычно ограничена, то запросы, поступившие в систему и заставшие накопитель заполненным, могут получить отказ в обслуживании и быть потерянными, либо вытеснить из накопителя менее важные (низкоприоритетные) заявки, которые будут потеряны (не обслужены). Обработанные (обслуженные) запросы покидают систему.

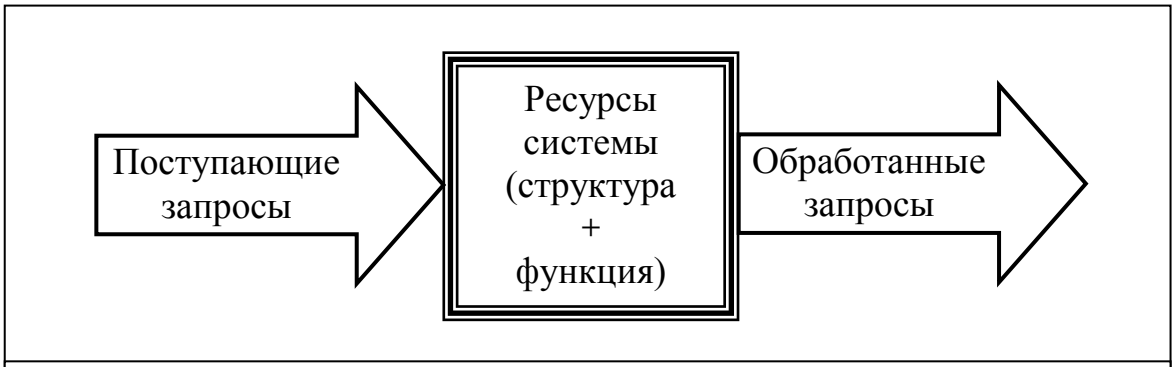

Рисунок 1. Система обслуживания

В качестве математических моделей функционирования таких систем широкое применение находят модели массового обслуживания разных классов (рисунок 2), а именно: базовые модели, представляющие собой системы массового обслуживания (СМО), и сетевые модели, представляющие собой сети массового обслуживания (СеМО), позволяющие эффективно решать задачи анализа свойств реальных систем и задачи синтеза на этапе проектирования.

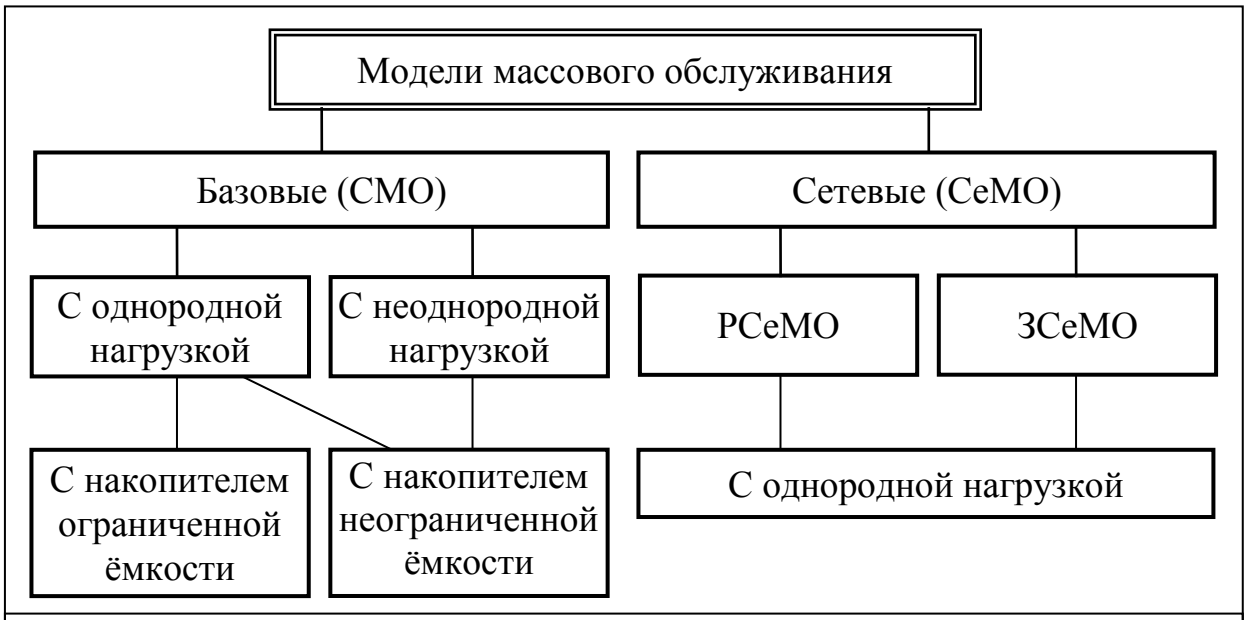

Рисунок 2. Модели дискретных систем

При рассмотрении задач проектирования дискретных систем ниже используются базовые модели с однородной нагрузкой и накопителем ограниченной и неограниченной ёмкости, а также модели с неоднородной нагрузкой и накопителем неограниченной ёмкости. В качестве сетевых моделей дискретных систем используются линейные разомкнутые (РСеМО) и замкнутые (ЗСеМО) сети массового обслуживания с однородным потоком заявок.

Основными показателями эффективности таких систем служат:

- время пребывания запросов в системе, измеряемое от момента поступления запроса до момента выхода его из системы после завершения обслуживания;

- вероятность потери запроса в системе с накопителем ограниченной ёмкости;
- производительность системы, измеряемая как среднее число обслуженных запросов за единицу времени;
- стоимость системы.

Проектирование дискретных систем со стохастическим характером функционирования на системном уровне осуществляется с применением различных моделей и методов в зависимости от класса проектируемой системы, ее особенностей и требований, предъявляемых к качеству функционирования. Задачи проектирования возникают при разработке новой системы или при модернизации (реконструкции) существующей системы. При этом задача проектирования сводится к определению структурно-функциональных параметров системы, обеспечивающих требуемое качество функционирования, задаваемое в виде ограничений, налагаемых на характеристики. В некоторых случаях к задаче проектирования относят также задачу определения вида нагрузки (класса задач, для решения которых система наиболее эффективна) и допустимых значений параметров нагрузки, возлагаемой на систему с заданной структурно-функциональной организацией. Таким образом, можно выделить две группы задач проектирования:

1) структурно-функциональное проектирование;

2) нагрузочное проектирование.

Очевидно, что наиболее востребованными и актуальными являются задачи структурно-функционального проектирования, которым в дальнейшем и уделяется основное внимание. Составляющими структурно-функционального проектирования являются:

1) структурное проектирование, заключающееся в определении состава элементов, конфигурации (топологии) системы, а также требований к значениям структурных параметров;

2) функциональное проектирование, заключающееся в определении стратегии управления потоками заявок, поступающих и обрабатываемых в системе.

Решение перечисленных задач осуществляется с учетом вида ограничений, налагаемых на показатели эффективности системы, среди которых одними из основных являются временные характеристики, например время пребывания запросов в системе, называемое в реальных системах временем задержки (временем реакции, временем ответа и т.п.).

2.2. Общая постановка задачи проектирования

Этап предварительного проектирования технических систем предполагает выполнение научно-исследовательской работы (НИР), в которой решаются задачи системного проектирования, в процессе которого необходимо, исходя из сведений о назначении системы и требований, предъявляемых к качеству ее функционирования, определить структурную и/или функциональную организацию системы, обеспечивающую реализацию

заданных функций. Для этого необходимо располагать знаниями о том, как влияют различные способы структурной и функциональной организации на характеристики функционирования системы, т.е. решать задачи системного анализа. Системное проектирование технических систем называется системотехническим проектированием.

Результаты НИР оформляются в виде технических предложений – технико-экономических обоснований целесообразности разработки системы и результатов сравнительного анализа различных вариантов реализации системы. Технико-экономическое обоснование структурно-функциональных параметров разрабатываемой системы выполняется на основе количественных оценок, полученных в процессе проведенных экспериментов и математических расчётов. Во многих случаях для этого разрабатываются физические или, чаще всего, математические модели, позволяющие выполнить сравнительный анализ различных вариантов построения системы и выбрать из них наилучший в соответствии с предъявляемыми требованиями.

Постановка задачи структурно-функционального проектирования на системотехническом уровне предполагает задание совокупности исходных данных, которая зависит от назначения системы и, в общем случае, включает:

- перечень структурных и функциональных параметров, определяемых в процессе проектирования;
- перечнем и значениями нагрузочных параметров, описывающих взаимодействие системы с внешней средой и потребность в ресурсах системы для реализации заданных функций;
- перечень характеристик (показателей эффективности) системы, описывающих эффективность проектируемой системы с точки зрения ее назначения;
- требования к проектируемой системе в виде ограничений, налагаемых на характеристики функционирования системы (мощностные, временные, надежностные, экономические);
- критерий эффективности для решения задачи оптимального синтеза, который может быть задан или сформирован на начальном этапе процесса проектирования.

На основе этих данных требуется определить структурную организацию системы, т.е. номенклатуру и состав элементов, а также конфигурацию (топологию) связей между ними, и функциональную организацию системы, то есть режим функционирования системы, обеспечивающий выполнение заданных ограничений и максимизирующий (минимизирующий) прямой (инверсный) критерий эффективности.

Системотехническое проектирование может быть направлено на решение частных задач, а именно:

- определение структурной организации системы при заданных параметрах функциональной организации и нагрузки (структурное проектирование);

- определение функциональной организации при заданных параметрах структурной организации и нагрузки (функциональное проектирование);
- определение предельной нагрузки, которая может быть реализована системой с заданной структурно-функциональной организацией (нагрузочное проектирование).

Процесс проектирования может заключаться в обоснованном выборе нескольких возможных вариантов структурно-функциональной организации разрабатываемой системы или в решении задачи оптимального синтеза на основе сформулированного критерия эффективности и результатах математического моделирования. В случае нескольких возможных вариантов построения системы выбор наилучшего варианта осуществляется по результатам сравнительного анализа основных характеристик рассматриваемых вариантов или с использованием критерия эффективности. Задачи оптимального синтеза решаются на основе математического моделирования и направлены на определение оптимальных значений параметров структурно-функциональной организации системы в соответствии с выбранным критерием эффективности.

2.3. Типовые подходы к проектированию дискретных систем

В зависимости от цели и постановки задачи проектирования дискретных систем с использованием моделей массового обслуживания можно сформулировать несколько типовых подходов к структурно-функциональному проектированию, а именно:

1) проектирование системы минимальной конфигурации;
2) проектирование системы минимальной конфигурации при наличии ограничений;
3) оптимальное проектирование системы;
4) оптимальное проектирование при наличии ограничений;
5) проектирование на основе сравнения нескольких вариантов.

Проектирование системы минимальной конфигурации. Под минимальной конфигурацией будем понимать такую структурно-функциональную организацию системы, при которой стоимость системы минимальна. При этом стоимость системы может определяться как в явном виде, так и неявно. Последнее означает, что стоимость системы не рассчитывается, а минимальность конфигурации обеспечивается только за счет минимально возможного количества устройств в системе и минимальных значений их параметров (быстродействий устройств, ёмкостей памяти, пропускных способностей каналов и т.п.).

Проектирование системы минимальной конфигурации предполагает наличие единственного требования – система должна справляться с возлагаемой на нее нагрузкой, т.е. система не должна быть перегружена. При этом к характеристикам функционирования системы не предъявляются никакие дополнительные требования, т.е. не налагаются предельные ограничения. Результатом проектирования являются граничные значения (нижние границы) структурно-функциональных параметров проектируемой системы.

Проектирование системы минимальной конфигурации при наличии ограничений. Система минимальной конфигурации проектируется с учетом ограничений, предъявляемых к предельным значениям характеристик (показателей эффективности) системы. Таким образом, кроме требования отсутствия перегрузок должны выполняться заданные ограничения на характеристики системы. Как и в предыдущем случае, в качестве критерия эффективности рассматривается стоимость системы, а результатами проектирования могут быть граничные значения (нижние границы) структурно-функциональных параметров проектируемой системы.

Оптимальное проектирование системы. Проектирование системы осуществляется на основе составного критерия эффективности, сформированного путем объединения всех показателей эффективности проектируемой системы в единый аддитивный или мультипликативный критерий. Задача оптимального проектирования (оптимального синтеза) предполагает применение математической модели функционирования системы. Результатом проектирования служат оптимальные значения структурно-функциональных параметров проектируемой системы, при которых достигается экстремум (минимум или максимум) составного критерия эффективности.

Оптимальное проектирование при наличии ограничений. В качестве критерия эффективности выбирается один из показателей эффективности, а на остальные налагаются ограничения. Результатом проектирования служат значения структурно-функциональных параметров проектируемой системы, при которых заданные ограничения выполняются при наименьшем (наибольшем) значении критерия эффективности. Если при этом достигается экстремум (минимум или максимум) критерия эффективности, то найденные значения структурно-функциональных параметров являются оптимальными. Таким образом, в результате проектирования структурно-функциональные параметры проектируемой системы могут оказаться не оптимальными в смысле выбранного критерия эффективности. В этом случае они называются наилучшими или рациональными. Аналогичная ситуация может возникнуть в случае представления в процессе решения задачи оптимального синтеза дискретных значений структурно-функциональных параметров непрерывными величинами, когда полученные дробные значения оптимизируемых параметров округляются до целых значений.

Проектирование на основе сравнения нескольких вариантов. Задача проектирования решается путем сравнительного анализа нескольких рассматриваемых (предлагаемых) вариантов и выбора наилучшего варианта с использованием критерия эффективности или на основе результатов сравнительного анализа основных показателей эффективности.

Краткая характеристика методов проектирования

Системотехническое проектирование дискретных систем проводится на основе математического моделирования с использованием:

- *аналитических* методов теории вероятностей, теории массового обслуживания, методов оптимизации и т.д.;
- *имитационных* экспериментов с применением специальных программных средств и языков моделирования.

Проектирование реальных систем обычно выполняется на основе *комбинированного подхода*, предполагающего совместное применение на разных этапах проектирования аналитических и имитационных методов.

3.1. Аналитические методы

Аналитическое исследование дискретных систем, моделями которых служат системы и сети массового обслуживания, предполагает использование разнообразных математических методов, позволяющих получить конечный результат в явном виде или в виде математических зависимостей (трансцендентных уравнений, систем алгебраических или дифференциальных уравнений и т.п.), решение которых может быть выполнено методами численного анализа.

К числу таких методов относятся:

- метод средних значений;
- метод марковских случайных процессов;
- метод введения дополнительного события;
- диффузионная аппроксимация и т.д.

Ниже рассматриваются первые два метода, используемые в последующих параграфах.

3.1.1. Метод средних значений

Среди методов расчёта характеристик моделей массового обслуживания одним из наиболее простых является метод средних значений, заключающийся в расчёте математических ожиданий, то есть средних значений, характеристик функционирования моделей массового обслуживания с использованием совокупности фундаментальных (базовых) зависимостей, связывающих средние значения параметров и характеристик моделей и справедливых для широкого класса систем. Такими зависимостями в теории массового обслуживания являются следующие:

- среднее время пребывания заявок в системе:

$$u = w + b, \qquad (1)$$

где w – среднее время ожидания заявок в очереди; b – средняя длительность обслуживания заявок в устройстве;

- среднее число заявок, поступающих в систему за время t:

$$m_t = \lambda t, \qquad (2)$$

где λ – интенсивность поступления заявок в систему;

- формулы Литтла для средней длины l очереди заявок и среднего числа заявок m в системе:

$$l = \lambda w; \qquad m = \lambda u; \qquad (3)$$

- среднее время дообслуживания заявки в устройстве:

$$T_0 = \frac{1}{2}\lambda b^{(2)} = \frac{1}{2}\lambda b^2(1+\nu_b^2), \qquad (4)$$

где $b^{(2)}$ и ν_b – соответственно второй начальный момент и коэффициент вариации длительности обслуживания заявок в устройстве, связанные следующей зависимостью: $b^{(2)} = b^2(1+\nu_b^2)$;

- усредненное время для заявок объединенного (суммарного) потока:

$$T = \frac{1}{\Lambda}\sum_{k=1}^{H}\lambda_k T_k, \qquad (5)$$

где T_k – среднее время (обслуживания, дообслуживания, ожидания, пребывания и т.д.) заявок класса k; H – количество классов заявок.

Метод средних значений достаточно успешно применяется для расчёта характеристик на уровне математических ожиданий как базовых, так и сетевых моделей.

3.1.2. Метод марковских случайных процессов

Случайный процесс называется марковским, если вероятность любого состояния в будущем зависит только от его состояния в настоящем и не зависит от того, когда и каким образом процесс оказался в этом состоянии.

Для того чтобы случайный процесс *с непрерывным временем* был марковским, необходимо, чтобы интервалы времени между соседними переходами из состояния в состояние были распределены по экспоненциальному закону [2].

Для описания марковского случайного процесса с дискретными состояниями и непрерывным временем используются следующие параметры:

- перечень состояний $\mathbf{E}_1,\ldots,\mathbf{E}_n$, в которых может находиться случайный процесс;
- матрица интенсивностей переходов $\mathbf{G} = [g_{ij} \mid i, j = \overline{1,n}]$, описывающая переходы случайного процесса между состояниями;
- начальные вероятности $p_1(0),\ldots, p_n(0)$.

Изучение случайных процессов заключается в определении вероятностей того, что в момент времени t система находится в том или ином состоянии. Совокупность таких вероятностей, описывающих состояния системы в различные моменты времени, дают достаточно полную информацию о протекающем в системе случайном процессе.

Если по истечении достаточно большого промежутка времени вероятности состояний стремятся к предельным значениям $p_1,\ldots, p_n$, не зависящим от начальных вероятностей $p_1(0),\ldots, p_n(0)$ и от текущего момента времени t, то говорят, что случайный процесс обладает эргодическим свойством. Таким образом, для процессов, обладающих эргодическим свойством:

$$\lim_{t\to\infty} P(t) = P(\infty) = \mathbf{P},$$

где $\mathbf{P} = (p_1, \ldots, p_n)$ – вектор вероятностей состояний системы, называемых *стационарными вероятностями*.

В системе, описываемой марковским случайным процессом, обладающим эргодическим свойством, при $t \to \infty$ устанавливается некоторый предельный режим, при котором характеристики функционирования системы не зависят от времени. В этом случае говорят, что система работает в установившемся или стационарном режиме.

Для однородного марковского процесса *с непрерывным временем* вероятности состояний на произвольный момент времени t определяются из системы дифференциальных уравнений:

$$\frac{\mathrm{d}\, p_j(t)}{\mathrm{d}t} = \sum_{i=1}^{n} p_i(t)\, g_{ij} \quad (j = \overline{1,n};\ t > 0) \qquad (6)$$

с учетом начальных условий $p_1(0), \ldots, p_n(0)$.

Для систем обладающих эргодическим свойством, имеет место стационарный режим, для которого вероятности состояний $p_1, \ldots, p_n$ при $t \to \infty$ не зависят от начальных вероятностей и текущего момента времени t, и система дифференциальных уравнений (6) для установившегося режима преобразуется в систему линейных алгебраических уравнений:

$$\sum_{i=1}^{n} p_i\, g_{ij} = 0 \quad (j = \overline{1,n}), \qquad (7)$$

которая совместно с нормировочным условием

$$\sum_{i=1}^{n} p_i = 1$$

образует систему, обладающую единственным решением.

Решая полученную систему уравнений аналитически для некоторых систем можно определить значения $p_0, p_1, \ldots, p_n$ стационарных вероятностей состояний марковского процесса, на основе которых могут быть рассчитаны другие характеристики исследуемой системы, например, среднее число заявок в системе, вероятность потери заявки и т.д.

При построении математической модели в каждом конкретном случае на основе описания исследуемой системы формулируются предположения и допущения, необходимые для того, чтобы протекающий в системе случайный процесс был марковским. Разработка марковской модели исследуемой системы в терминах случайных процессов предполагает выполнение следующих этапов:

- кодирование состояний случайного процесса;
- построение размеченного графа переходов;
- формирование матрицы интенсивностей переходов;
- составление системы линейных алгебраических уравнений для расчёта стационарных вероятностей состояний марковского процесса.

Ниже приведены примеры, иллюстрирующие применение метода средних значений и марковских случайных процессов для получения математических

зависимостей характеристик простейших моделей массового обслуживания от структурно-функциональных и нагрузочных параметров, на основе которых решаются задачи проектирования систем.

3.2. Имитационные методы и средства

Имитационные методы проектирования базируются на имитационном моделировании, позволяющем проводить исследование систем любой сложности с любой степенью детализации. Применительно к дискретным системам со стохастическим характером функционирования это означает, прежде всего, возможность исследования свойств систем при любых законах распределений случайных величин, описывающих нагрузку.

Имитационное моделирование в процессе системотехнического проектирования применяется в следующих случаях:

- для установления адекватности аналитических моделей при отсутствии возможности сравнения аналитических результатов с результатами измерений на реальной системе;
- для оценки погрешностей приближенных аналитических методов и граничных оценок;
- для выбора наилучшего варианта построения системы из нескольких возможных вариантов;
- для детального анализа спроектированной системы с целью установления соответствия характеристик спроектированной системы заданным требованиям.

Наиболее эффективным имитационное моделирование оказывается при сравнении несколько вариантов построения системы. В то же время, с использованием имитационного моделирования практически невозможно решать задачи оптимального синтеза систем, описываемых большим числом структурно-функциональных и нагрузочных параметров.

Имитационные методы проектирования дискретных систем разрабатываются с применением программных средств, которые можно разбить на две группы:

- коммерческие средства, характеризующиеся удобным графическим интерфейсом и наличием библиотеки устройств и оборудования, входящего в состав проектируемых систем; недостатком коммерческих средств является их закрытость, что не позволяет, при необходимости, расширить диапазон исследований и выполнить более детальный анализ характеристик функционирования системы, и достаточно высокая стоимость; примером такого средства является система моделирования AnyLogic;
- специализированные средства, представляющие собой системы и языки имитационного моделирования, позволяющие строить модели с реализацией любых необходимых функций и с возможностью получения результатов с любой степенью детализации; примером такого средства является система моделирования GPSS.

3.3. Комбинированный подход

Наиболее универсальным и эффективным подходом к проектированию дискретных систем со стохастическим характером функционирования является комбинированный подход, основанный на совместном применении аналитических и имитационных методов на разных этапах проектирования, а именно:

- аналитические методы на этапах анализа свойств и оптимального синтеза на основе критерия эффективности;
- имитационные методы на этапах сравнительного анализа нескольких вариантов построения системы и детального анализа свойств спроектированной системы.

Аналитические методы целесообразно применять при решении задач, связанных с исследованием свойств системы на основе анализа зависимостей характеристик функционирования системы от значений структурно-функциональных параметров и параметров нагрузки. По результатам анализа формулируются рекомендации по проектированию с учетом требований к качеству функционирования системы. Поскольку аналитические методы обычно разрабатываются для сравнительно простых моделей, результаты могут иметь значительную погрешность. Для оценки погрешностей приближенных аналитических результатов используется имитационное моделирование, в процессе которого выявляются новые свойства системы, которые не могли быть получены аналитическими методами, например свойства, присущие переходному режиму функционирования системы или режиму перегрузок.

Наиболее эффективными аналитические методы оказываются при решении задач синтеза оптимальной системы. Однако, из-за применения приближенных аналитических зависимостей, результаты оптимизации могут существенно отличаться от истинных значений. Уточнение результатов оптимизации выполняется на этапе детального анализа спроектированной системы с использованием имитационного моделирования, позволяющего проводить исследование систем практически любой сложности и с любой степенью детализации.

Проектирование систем на основе базовых моделей с однородной нагрузкой

При решении задач системотехнического проектирования дискретных систем со стохастическим характером функционирования в качестве простейших моделей используются системы массового обслуживания различных классов.

4.1. Постановка задачи проектирования систем с однородной нагрузкой

В системах с однородной нагрузкой циркулирует один класс заявок.

При проектировании реальных технических систем во многих случаях оказывается целесообразным использование в качестве моделей систем массового обслуживания с накопителями неограниченной ёмкости, для

которых при некоторых дополнительных предположениях могут быть получены сравнительно простые аналитические зависимости для расчёта характеристик функционирования системы. Как будет показано ниже, предположение о неограниченной ёмкости накопителя в модели оправдано в тех случаях, когда ёмкость накопителя реальной системы такова, что вероятность потери заявок из-за переполнения накопителя незначительна. Примерами таких систем могут служить системы, управляющие некоторым внешним объектом в реальном времени, для которых не то что потеря заявок, а даже просто превышение допустимого значения времени реакции системы (пребывания заявки в системе) может привести к катастрофическим последствиям.

Если же вероятность потери заявок из-за переполнения накопителя существенна, то для решения задач проектирования необходимо применять модели с накопителями ограниченной ёмкости.

В соответствии с типовыми подходами к проектированию дискретных систем, изложенных в предыдущем параграфе, для систем с однородной нагрузкой, представляемых моделями с накопителями неограниченной ёмкости, задачи структурно-функционального проектирования формулируются в следующих постановках:

- спроектировать систему минимальной конфигурации (стоимости), обеспечивающую отсутствие перегрузок;
- спроектировать систему, обеспечивающую выполнение заданного ограничения на время пребывания запросов в системе;
- спроектировать систему на основе обобщенного критерия эффективности;
- спроектировать систему на основе критерия эффективности с учетом ограничения на время пребывания запросов в системе.

Для систем, представляемых моделями с накопителями ограниченной ёмкости, важной характеристикой является вероятность потери заявок в результате переполнения накопителя. В этом случае задачи структурно-функционального проектирования формулируются в несколько иных постановках:

- спроектировать систему, обеспечивающую выполнение заданного ограничения на вероятность потери заявок;
- спроектировать систему на основе обобщенного критерия эффективности, включающего время пребывания запросов в системе и вероятность потери заявок;
- спроектировать систему на основе критерия эффективности с учетом ограничений на время пребывания запросов в системе и вероятность потери заявок.

4.2. Системы с одним устройством и накопителем неограниченной ёмкости

Положим, что исследуемая система содержит одно устройство, обрабатывающее запросы З (заявки, команды, транзакции, детали и т.п.),

поступающие в систему в случайные моменты времени независимо от того, сколько в системе уже находится запросов, то есть источник запросов – неограниченный. Поступающие запросы образуют однородный поток и создают в системе однородную нагрузку. Средний интервал между запросами равен a. Средняя ресурсоёмкость, измеряемая количеством работы, которое затрачивается на обработку одного запроса, равна θ. Разброс ресурсоёмкости, являющейся в общем случае величиной случайной, задается коэффициентом вариации: $\nu_\theta = \sigma_\theta / \theta$, где σ_θ – среднеквадратическое отклонение ресурсоёмкости. Скорость обработки запросов в устройстве (производительность или быстродействие устройства), измеряемая количеством работы, выполняемой устройством за единицу времени, равна V. Тогда длительность обработки одного запроса, как и ресурсоёмкость, распределена по произвольному закону со средним значением $b = \dfrac{\theta}{V}$ и коэффициентом вариации $\nu_b = \nu_\theta$. В каждый момент времени устройство может обрабатывать только один запрос. Запросы, поступившие в систему и заставшие устройство занятым обработкой ранее поступившего запроса, располагаются в накопителе перед устройством, где ожидают освобождения устройства. Тогда в качестве модели обработки запросов может использоваться одноканальная СМО (с одним обслуживающим устройством) с однородным потоком заявок (З) и накопителем (Н) ограниченной или неограниченной ёмкости (рисунок 3).

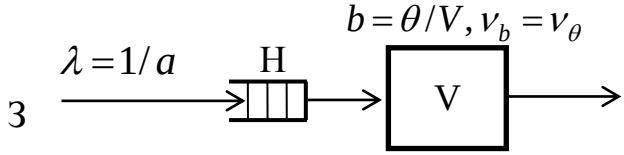

Рисунок 3. Одноканальная СМО

Представленная одноканальная СМО, как показано в [1], может служить моделью: процессорной обработки в однопроцессорной системе; симплексного канала связи, а также упрощенной моделью сервера локальной сети, узла связи (маршрутизатора, коммутатора) и т.д.

4.2.1. Расчёт характеристик систем с накопителем неограниченной ёмкости

Положим, что заявки, поступающие в систему, образуют простейший поток заявок с интенсивностью $\lambda = 1/a$. Длительность обработки одной заявки распределена по произвольному закону со средним значением $b = \dfrac{\theta}{V}$ и коэффициентом вариации $\nu_b = \nu_\theta$. Если ёмкость накопителя перед устройством достаточна для хранения всех поступающих в систему заявок, т.е. любая поступившая в систему заявка всегда найдёт место в накопителе, то в качестве модели обработки заявок может использоваться СМО (рисунок 3), которая в терминах символики Кендалла [2] обозначается как M/G/1, где первый символ

M (Markovian) обозначает, что поток заявок простейший (марковский), а второй символ G (General) – длительность обслуживания заявок в устройстве распределена по произвольному закону общего вида.

Таким образом, ниже в качестве модели рассматривается одноканальная СМО с накопителем неограниченной ёмкости, в которую поступает простейший поток заявок, длительность обслуживания которых распределена по произвольному закону общего вида.

Положим, что нагрузка рассматриваемой системы меньше единицы: $y = \lambda b = \frac{\lambda\theta}{V} < 1$, следовательно, загрузка совпадает с нагрузкой: $\rho = y < 1$.

Математические зависимости для расчёта характеристик простейших базовых моделей, представляющих собой одноканальные СМО с однородным потоком заявок, легко могут быть получены с использованием метода средних значений, изложенного в пункте 3.1.1, путём следующих рассуждений.

Среднее время ожидания заявки, поступившей в систему в некоторый момент времени, будет складываться из среднего времени дообслуживания T_0 заявки, уже находящейся на обслуживании в устройстве, и среднего времени обслуживания T_l всех ранее поступивших заявок и находящихся в очереди:

$$w = T_0 + T_l.$$

Время дообслуживания T_0 определяется выражением (4), а T_l – выражением $T_l = lb$, учитывающим, что в очереди находится в среднем l заявок, каждая из которых в среднем будет обслуживаться в устройстве в течение времени b. Тогда с использованием формулы Литтла (3) для средней длины очереди $(l = \lambda w)$, получим:

$$w = \frac{1}{2}\lambda b^2(1+\nu_b^2) + \lambda b w.$$

Отсюда после некоторых преобразований, с учетом того, что $\rho = \lambda b < 1$, окончательно получим выражение для расчёта среднего времени ожидания заявок в СМО M/G/1, известное как формула Поллачека-Хинчина:

$$w = \frac{\rho b(1+\nu_b^2)}{2(1-\rho)}. \qquad \textbf{(8)}$$

Полученное выражение может быть легко преобразовано для СМО типа M/M/1 с учетом того, что для нее $\nu_b = 1$. Тогда среднее время ожидания:

$$w = \frac{\rho b}{1-\rho}, \qquad (9)$$

и среднее время пребывания:

$$u = w + b = \frac{b}{1-\rho}. \qquad (10)$$

Более детальное исследование СМО M/M/1 предполагает расчёт характеристик функционирования на уровне более высоких моментов, в пределе – на уровне законов распределений случайных величин.

Для нахождения закона распределения числа заявок в системе и, соответственно, длины очереди заявок, воспользуемся методом марковских процессов.

Из описания СМО (простейший поток и экспоненциальная длительность обслуживания заявок) следует, что функционирование системы может быть описано в терминах марковского процесса.

Для кодирования состояний марковского процесса в качестве параметра, описывающего состояние марковского процесса, будем рассматривать количество заявок k, находящихся в СМО (в устройстве и в накопителе). Поскольку в системе в произвольный момент времени может находиться любое сколь угодно большое число заявок, то количество состояний марковского процесса равно бесконечности:

E_0: $k = 0$ – в системе нет ни одной заявки;

E_1: $k = 1$ – в системе находится 1 заявка (на обслуживании в приборе);

E_2: $k = 2$ – в системе находятся 2 заявки (одна – на обслуживании в приборе и вторая ожидает в накопителе);

…

E_k: k – в системе находятся k заявок (одна – на обслуживании в приборе и $(k-1)$ – в накопителе).

…

На рисунке 4 представлен размеченный граф переходов случайного процесса.

В один и тот же момент времени может происходить только одно событие: поступление заявки в систему с интенсивностью λ или завершение обслуживания заявки с интенсивностью $\mu = 1/b$. Размеченный граф переходов содержит бесконечное число состояний.

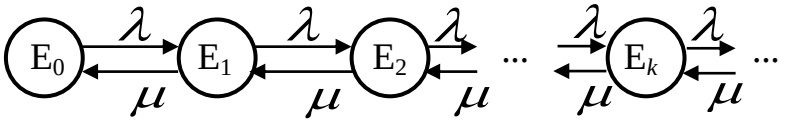

Рисунок 4. Граф переходов марковского процесса

Не выписывая матрицу интенсивностей переходов, составим по графу переходов **систему уравнений** для определения стационарных вероятностей:

$$\begin{cases} \lambda p_0 = \mu p_1 \\ (\lambda+\mu) p_1 = \lambda p_0 + \mu p_2 \\ (\lambda+\mu) p_2 = \lambda p_1 + \mu p_3 \\ \quad \dots \\ (\lambda+\mu) p_k = \lambda p_{k-1} + \mu p_{k+1} \\ \quad \dots \\ p_0 + p_1 + p_2 + \dots + p_k + \dots = 1 \end{cases}$$

Несмотря на то, что система содержит бесконечное число уравнений и, соответственно, бесконечное число неизвестных, нетрудно методом математической индукции получить аналитическое решение в явном виде для расчёта вероятностей состояний одноканальной экспоненциальной СМО с однородным потоком заявок и накопителем неограниченной ёмкости при условии, что нагрузка системы $y < 1$:

$$p_k = y^k (1-y) = \rho^k (1-\rho) \quad (k = 0, 1, 2, \ldots), \qquad (11)$$

где $\rho = \lambda / \mu$ – загрузка системы, совпадающая с нагрузкой, причём $\rho < 1$, что гарантирует отсутствие перегрузок в системе.

Таким образом, вероятность того, что в произвольный момент времени в системе находится k заявок, распределена по геометрическому закону (11) со значением параметра распределения, равным загрузке (нагрузке) системы.

Начальные моменты числа заявок в очереди и в системе рассчитываются по формулам:

$$l^{(n)} = \sum_{k=1}^{\infty} (k-1)^n p_k = (1-\rho) \sum_{k=1}^{\infty} (k-1)^n \rho^k ;$$

$$m^{(n)} = \sum_{k=0}^{\infty} k^n p_k = (1-\rho) \sum_{k=0}^{\infty} k^n \rho^k .$$

Откуда средние значения длины очереди и числа заявок в системе:

$$l = l^{(1)} = \frac{\rho^2}{1-\rho} \qquad \text{и} \qquad m = m^{(1)} = \frac{\rho}{1-\rho}.$$

С использованием формул Литтла (3) легко получить выражения для расчёта среднего времени ожидания заявок и среднего времени пребывания заявок в системе:

$$w = \frac{l}{\lambda} = \frac{\rho b}{1-\rho}; \quad u = \frac{m}{\lambda} = \frac{b}{1-\rho}.$$

Последнее выражение для среднего времени пребывания u совпадает с формулой (10**)**.

Распределение (11) может использоваться для расчёта вероятности превышения числа заявок в системе некоторого предельного значения m^*:

$$\Pr(M > m^*) = \sum_{k=m^*+1}^{\infty} p_k$$

Подставляя (11) в это выражение, получим:

$$\Pr(M > m^*) = (1-\rho) \sum_{k=m^*+1}^{\infty} \rho^k ,$$

где сумма представляет собой геометрическую прогрессию: $\sum_{k=m^*+1}^{\infty} \rho^k = \frac{\rho^{m^*+1}}{1-\rho}$.

Тогда

$$\Pr(M > m^*) = \rho^{m^*+1}.$$

Отсюда можно определить нижнюю границу числа заявок $\hat{m}$, вероятность превышения которого равна δ^*:

$\rho^{\hat{m}+1} = \delta^*$.

Логарифмируя левую и правую часть последнего выражения, после некоторых преобразований получим:

$$\hat{m} = \left\lceil \frac{\lg \delta^*}{\lg \rho} - 1 \right\rceil,$$

где $\lceil x \rceil$ означает ближайшее большее целое по отношению x (округление x до целого в большую сторону).

Полагая $\delta^* = 10^{-n}$ $(n = 2,3,\ldots)$, получим:

$$\hat{m} = -\left\lceil \frac{n}{\lg \rho} + 1 \right\rceil. \qquad (12)$$

При загрузке системы $\rho = 0{,}9$ и вероятности превышения $\delta^* = 0{,}001$ нижняя граница числа заявок в системе $\hat{m} = 65$, а при $\delta^* = 0{,}0001$ – $\hat{m} = 87$.

4.2.2. Определение нижней границы производительности устройства

В качестве примера простейшей задачи проектирования системы минимальной конфигурации рассмотрим задачу определения нижней границы производительности устройства с использованием базовой модели (рисунок 3) с одним устройством, обрабатывающим однородный поток заявок, и накопителем неограниченной ёмкости.

Под нижней границей производительности устройства будем понимать такую производительность, при которой в системе отсутствуют перегрузки.

Нагрузка y и загрузка ρ системы соответственно определяются как

$$y = \lambda b = \frac{\lambda\theta}{V}; \qquad \rho = \min(y; 1).$$

Для того чтобы в системе не было перегрузок, необходимо выполнение следующего условия: $\frac{\lambda\theta}{V} < 1$, из которого следует:

$$\boxed{V > \lambda\theta}. \qquad (13)$$

Выражение (13) можно рассматривать как ограничение, налагаемое на производительность устройства (скорость работы устройства) и обеспечивающее отсутствие перегрузок в системе. Величина $V_0 = \lambda\theta$ представляет собой нижнюю границу производительности устройства. Если производительность устройства будет меньше или равна V_0, то в системе возникнут перегрузки, что приведёт к неограниченному возрастанию длины очереди заявок перед устройством.

Для выбранного значения производительности устройства *V*, удовлетворяющего условию (13), среднее время пребывания заявок в системе с учетом типа модели M/M/1 будет равно:

$$u = \frac{b}{1-\rho} = \frac{\theta}{V - \lambda\theta}. \qquad (14)$$

Пусть значение производительности устройства $V = k\lambda\theta \quad (k > 1)$, тогда загрузка системы $\rho = 1/k$, и выражение (14) примет вид:

$$u = \frac{1}{(k-1)\lambda} = \frac{a}{k-1},$$

т.е. среднее время пребывания заявок в системе не зависит от ресурсоёмкости обработки.

При $k = 1{,}25$ загрузка системы $\rho = 0{,}8$, а среднее время пребывания заявок в системе $u = 4a$ в 4 раза больше среднего интервала между поступающими заявками, а при $k = 2$ – загрузка $\rho = 0{,}5$ и среднее время пребывания $u = a$.

Запишем теперь выражение (14) в следующем виде:

$$u = \frac{\theta}{V - V_0}.$$

где $V_0 = \lambda\theta$ – нижняя граница производительности устройства.

Как и ранее, положим, что $V = k\lambda\theta = kV_0 \quad (k > 1)$. Тогда последнее выражение примет вид:

$$u = \frac{b_д}{k-1},$$

где $b_д = \frac{\theta}{V_0}$ – максимально допустимое время обработки заявок, при котором загрузка системы $\rho = 1$.

4.2.3. Определение минимальной производительности устройства с учётом ограничения на среднее время пребывания заявок

Пусть для описанной выше системы задано ограничение на среднее время пребывания u заявок в системе в виде:

$$u < u^*, \qquad (15)$$

где u^* – допустимое среднее время пребывания заявок в системе.

С учетом (14) неравенство (15) запишется в следующем виде:

$$\frac{\theta}{V - \lambda\theta} < u^*,$$

откуда получим выражение для оценки производительности устройства:

$$\boxed{V > \lambda\theta + \frac{\theta}{u^*}}. \qquad (16)$$

Правая часть полученного выражения представляет собой минимальную производительность $V_{\min}$, при которой выполняется заданное ограничение и которое складывается из двух составляющих: нижней производительности $V_0 = \lambda\theta$, обеспечивающей отсутствие перегрузок в системе, и дополнительной

производительности $V_{\text{доп}} = \frac{\theta}{u^*}$, необходимой для обеспечения заданного ограничения (15).

Характер зависимости минимальной производительности устройства от ограничения на время пребывания заявок в системе показан на рисунке 5. Область допустимых значений производительности лежит выше представленной зависимости $V_{\min}$. Если производительность устройства выбрать в интервале между значениями $V_{\min}$ и V_0, то система будет работать без перегрузок, но заданное ограничение (15) не будет выполняться.

При увеличении допустимого среднего времени пребывания заявок в системе до бесконечности $u^* \to \infty$, что можно трактовать как отсутствие ограничения на время пребывания, минимальная производительность $V_{\min}$ уменьшается и стремится к V_0: $V_{\min} \to V_0$.

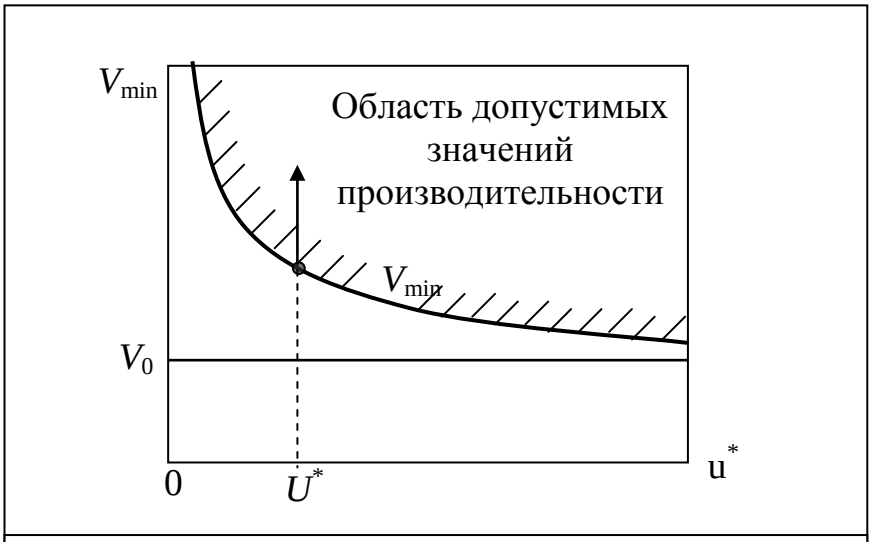

Рисунок 5. Нижняя граница производительности

4.2.4. Определение оптимальной производительности устройства

В качестве показателей эффективности проектируемой системы, моделью которой служит СМО M/M/1, будем рассматривать:

- u – среднее время пребывания заявки в системе;
- S – стоимость системы.

Положим, что стоимость системы зависит только от производительности устройства и определяется выражением:

$$S = \xi V^{\chi},$$

где ξ и χ – стоимостные коэффициенты пропорциональности и нелинейности соответственно.

Для определения оптимальной производительности устройства воспользуемся составным критерием эффективности:

$$F = k_1 u + k_2 S = \frac{k_1 \theta}{V - \lambda\theta} + k_2 \xi V^{\chi} - \min, \qquad (17)$$

где k_1 и k_2 – весовые коэффициенты соответственно времени пребывания заявок в системе и стоимости системы, определяющие степень важности (ценности) соответствующего показателя эффективности, причем $k_1 + k_2 = 1$.

Значения весовых коэффициентов обычно формируются на основе экспертных оценок, что в определенной степени обусловливает их субъективность.

Заметим, что корректность выражения (17) обеспечивается наличием разных размерностей весовых коэффициентов k_1 и k_2, а именно: размерность k_1 – у.е./с, а k_2 – величина безразмерная. В этом случае критерий эффективности F имеет размерность стоимости в у.е., где первое слагаемое представляет собой стоимость (штраф) за задержку при обработке заявки, а второе – стоимость самой системы (устройства).

Проиллюстрируем на графике применение критерия эффективности (17). Очевидно, что с увеличением производительности устройства среднее время пребывания заявок в системе уменьшается, а стоимость системы увеличивается (рисунок 6). Можно ожидать, что при некотором значении производительности $V = V_{\text{opt}}$, представляющем собой оптимальное значение, критерий эффективности F примет минимальное значение: $F = \min$.

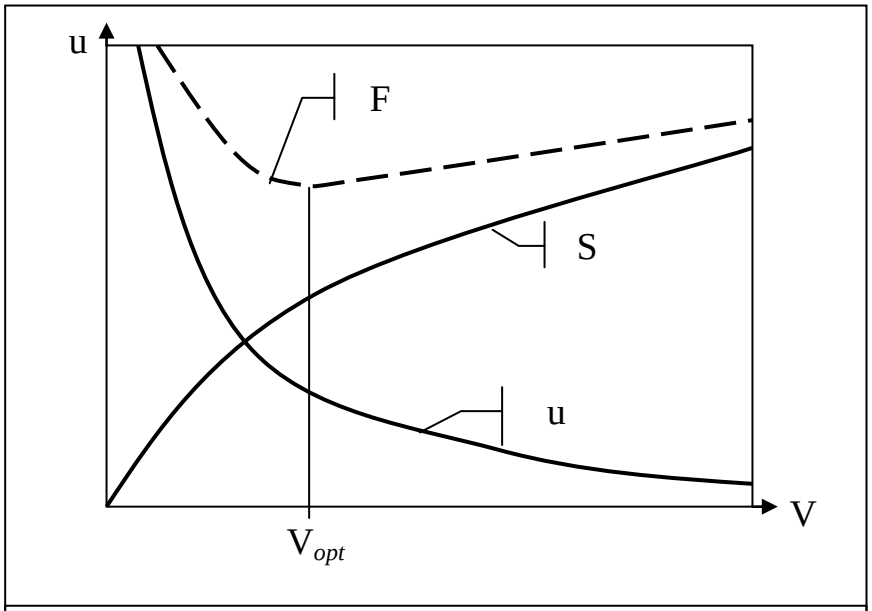

Рисунок 6. Оптимальная производительность

Найдем минимум функции F. Для этого возьмём производную от этой функции по V и приравняем её нулю:

$$\frac{dF}{dV} = -\frac{k_1\theta}{(V-\lambda\theta)^2} + k_2\xi\chi V^{\chi-1} = 0.$$

Из последнего выражения, полагая для простоты $\chi = 1$, найдем оптимальную производительность устройства:

$$\boxed{V_{\text{opt}} = \lambda\theta + \sqrt{\frac{k_1}{k_2}}\sqrt{\frac{\theta}{\xi}}}. \qquad (18)$$

Стоимость спроектированной системы (устройства) будет равна:

$$S_{\text{opt}} = \xi\lambda\theta + \sqrt{\frac{k_1\xi\theta}{1-k_1}}.$$

В последнем выражении первое слагаемое в правой части представляет собой стоимость системы минимальной конфигурации, которая определяется значением нижней границы производительности, а второе слагаемое – это дополнительные затраты, обеспечивающие построение оптимальной системы в соответствии с заданным критерием эффективности (17).

Выполним анализ выражения (18). Оптимальное значение производительности складывается из двух составляющих: производительности $V_0 = \lambda\theta$, обеспечивающей отсутствие перегрузок в системе и некоторой дополнительной производительности $V_{\text{доп}} = \sqrt{\frac{k_1}{k_2}}\sqrt{\frac{\theta}{\xi}}$, обеспечивающей оптимальное решение. Последняя зависит от весовых коэффициентов k_1 и k_2, ресурсоёмкости обработки заявок θ и стоимостного коэффициента пропорциональности ξ. Дополнительная производительность $V_{\text{доп}}$ тем больше, чем больше значение k_1 (означающее, что более важным показателем эффективности является среднее время пребывания заявки в системе) и чем больше ресурсоёмкость θ. И наоборот, чем более важным является стоимость системы S, т.е. чем больше k_2, и чем больше стоимостной коэффициент пропорциональности ξ, тем меньше дополнительная производительность устройства $V_{\text{доп}}$. Заметим, что большое значение ξ означает, что незначительное увеличение производительности устройства приводит к резкому росту его стоимости. Следовательно, вкладывать значительные средства для незначительного увеличения производительности нецелесообразно.

Рассмотрим два крайних случая, когда:

1) $k_1 = 0$ и $k_2 = 1$;

2) $k_1 = 1$ и $k_2 = 0$.

Первый случай ($k_1 = 0$ и $k_2 = 1$) соответствует проектированию системы, к которой предъявляется только одно требование – минимизировать стоимость системы, при этом отсутствует ограничение к задержке заявок. В этом случае $V_{\text{доп}} = 0$, и оптимальное решение совпадает с нижней границей производительности: $V_{\text{опт}} = V_0 = \lambda\theta$, обеспечивающей только требование отсутствия перегрузок.

Во втором случае ($k_1 = 1$ и $k_2 = 0$) проектирование системы направлено на обеспечение минимальной задержки заявок, при этом стоимость системы не имеет никакого значения. Очевидно, что наилучшим решением будет система с бесконечно большой производительностью $V_{\text{опт}} = \infty$ и, соответственно с бесконечной стоимостью $S = \infty$.

Если показатели эффективности u и S – равноценны, то весовые коэффициенты $k_1 = k_2$, и, следовательно, в критерии эффективности (17) они могут быть опущены. Тогда выражение (18) для расчёта оптимальной производительности устройства примет вид:

$$V_{\text{opt}} = \lambda\theta + \sqrt{\frac{\theta}{\xi}},$$

а стоимость такой системы составит:

$$S_{\text{opt}} = \xi\lambda\theta + \sqrt{\xi\theta}.$$

4.2.5. Определение оптимальной производительности устройства с учетом ёмкости накопителя

В предыдущем параграфе при решении задачи проектирования системы учитывалась только стоимость устройства, зависящая от его производительности. Однако в реальных системах, построенных, например, на основе вычислительных машин, стоимость накопителя может быть соизмерима и даже превосходить стоимость устройства. Другими словами, при проектировании таких систем следует учитывать и стоимость накопителя.

Очевидно, что при невысокой производительности устройства его стоимость будет небольшой, но при этом в системе будет скапливаться большое число заявок, для хранения которых потребуется накопитель большой ёмкости и, следовательно, большой стоимости. И, наоборот, при высокой производительности и, соответственно, высокой стоимости устройства заявки будут быстро обрабатываться и не будут скапливаться в системе. Для их хранения потребуется накопитель небольшой ёмкости и, следовательно, небольшой стоимости. Сказанное иллюстрируется на рисунке 7, где предполагается линейная зависимость стоимости от производительности устройства, а зависимость стоимости накопителя прямо пропорциональна максимальному числу заявок в системе, которая, как будет показано ниже, имеет степенной характер. Таким образом, существует некоторое значение производительности V_{opt}, называемое оптимальным, при котором стоимость системы S, складывающаяся из стоимости устройства $S_{\text{У}}$ и стоимости накопителя $S_{\text{Н}}$, будет минимальной.

Рассмотрим задачу проектирования систем с учетом ёмкости накопителя, необходимой для хранения поступающих в систему заявок. Стоимость системы складывается из стоимости устройства и стоимости накопителя (стоимости хранения заявок в накопителе):

$$S = S_{\text{У}} + S_{\text{Н}}.$$

Пусть, как и ранее, в качестве модели системы используется СМО типа M/M/1, а стоимость устройства определяется как $S_{\text{У}} = \xi V^{\chi}$.

Стоимость накопителя определяется его ёмкостью, используемой для хранения заявок, и зависит от максимального количества заявок, которые могут находиться в накопителе:

$$S_{\text{Н}} = s_0 \hat{m},$$

где $\hat{m} = f_0 m$ – максимально возможное количество заявок, одновременно находящихся в накопителе и определяемое через среднее число заявок в системе m и некоторый коэффициент $f_0 > 1$, зависящий от закона распределения числа запросов в системе, допустимой вероятности превышения максимального количества заявок в накопителе и загрузки системы.

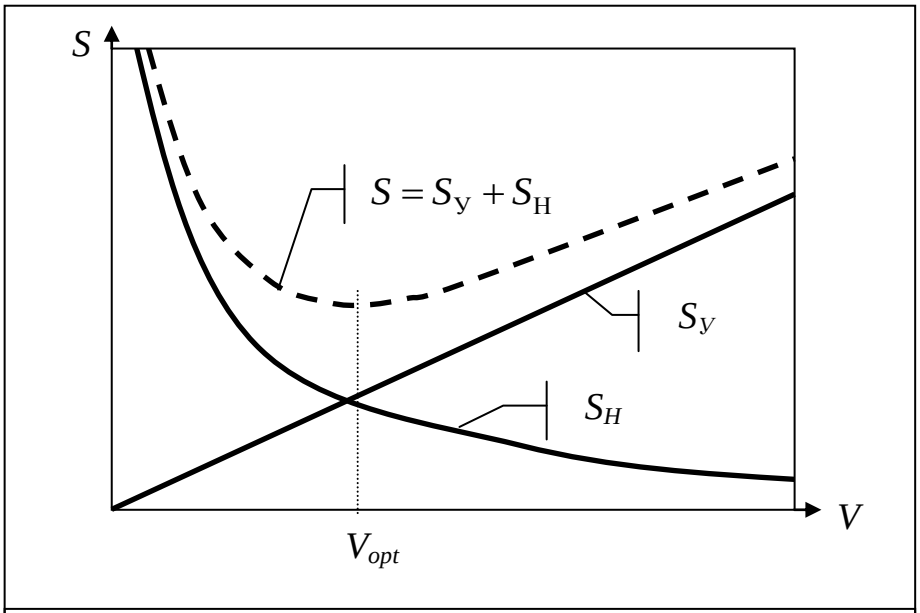

Рисунок 7. Оптимизация по критерию «стоимость»

Тогда с учетом (3) и (14) получим:

$$S_{\text{Н}} = s_0 f_0 m = s_0 f_0 \lambda u = s_0 f_0 \frac{\lambda\theta}{V - \lambda\theta}.$$

Здесь предполагается, что обрабатываемая в устройстве заявка также занимает место в накопителе, например, обрабатываемые в центральном процессоре вычислительной системы данные занимают место в памяти так же, как и ожидающие обработки.

Таким образом, стоимость системы будет равна:

$$S = \xi V^{\chi} + \frac{s_0 f_0 \lambda\theta}{V - \lambda\theta}. \qquad (19)$$

Используя выражение (19) в качестве критерия эффективности, можно решать задачи проектирования, связанные с определением оптимальной производительности устройства системы. При этом постановка задачи оптимизации формулируется следующим образом: определить производительность устройства V, при котором стоимость системы, определяемая выражением (19) минимальна: S – min.

С целью упрощения математических преобразований положим, что стоимость устройства связана с его производительностью линейной зависимостью, то есть стоимостной коэффициент нелинейности $\chi = 1$.

Для нахождения минимума выражения (19) вычислим производную от S по V и приравняем её нулю:

$$\frac{dS}{dV} = \xi - \frac{s_0 f_0 \lambda\theta}{(V-\lambda\theta)^2} = 0.$$

Решая полученное уравнение, найдем оптимальную производительность устройства:

$$\boxed{V_{\text{opt}} = \lambda\theta + \sqrt{\frac{s_0 f_0 \lambda\theta}{\xi}}}. \qquad (20)$$

Тогда максимальная ёмкость накопителя, которая также является и оптимальной в соответствии со стоимостным критерием эффективности (19), будет равна:

$$E_{\max} = E_{\text{opt}} = \sqrt{\frac{\xi f_0 \lambda\theta}{s_0}}.$$

Загрузка системы и среднее время пребывания заявок в оптимальной системе соответственно равны:

$$\rho = \frac{\lambda\theta}{V_{\text{opt}}} = \frac{\sqrt{\xi\lambda\theta}}{\sqrt{\xi\lambda\theta} + \sqrt{s_0 f_0}};$$

$$u = \frac{\theta}{V_{\text{opt}}(1-\rho)} = \sqrt{\frac{\xi\theta}{s_0 f_0 \lambda}}.$$

Легко видеть, что загрузка $\rho < 1$.

Подставляя (20) в (19) с учетом $\chi = 1$, определим минимальную стоимость системы:

$$S_{\min} = \xi\lambda\theta + 2\sqrt{\xi s_0 f_0 \lambda\theta},$$

где $S_{\text{У}} = \xi\lambda\theta + \sqrt{\xi s_0 f_0 \lambda\theta}$ – стоимость устройства, а $S_{\text{Н}} = \sqrt{\xi s_0 f_0 \lambda\theta}$ – стоимость накопителя.

В заключение оценим значение коэффициента f_0 для модели M/M/1.

Для этого воспользуемся выражением (11) для расчёта нижней границы числа заявок в системе.

При загрузке системы $\rho = 0{,}1$ нижняя граница числа заявок $\hat{m} = \lceil n-1 \rceil$, а среднее число заявок в системе: $m = \frac{\rho}{1-\rho} = \frac{1}{9}$. Тогда с учетом того, что n принимает только целые значения ($n = 1, 2, \ldots$):

$$\frac{\hat{m}}{m} = 9(n-1) \qquad \text{или} \qquad \hat{m} = 9(n-1)m.$$

Следовательно, коэффициент $f_0 = 9(n-1)$, если загрузка системы $\rho = 0{,}1$. Таким образом, при $n = 3$ максимальное число заявок в системе будет больше среднего в 18 раз, а при $n = 4$ – в 27 раз.

При загрузке системы $\rho = 0{,}9$ нижняя граница числа заявок $\hat{m} = \left\lceil \frac{n}{0{,}046} - 1 \right\rceil$, а среднее число заявок в системе: $m = \frac{\rho}{1-\rho} = 9$. Тогда отношение:

$$\frac{\hat{m}}{m} = \frac{\lceil 21{,}74\,n - 1 \rceil}{9} \quad \text{или} \quad \hat{m} = \lceil 2{,}42\,n - 0{,}11 \rceil m.$$

Следовательно, коэффициент $f_0 = \lceil 2{,}42\,n - 0{,}11 \rceil$, если загрузка системы $\rho = 0{,}9$. Таким образом, при $n = 3$ максимальное число заявок в системе будет больше среднего в 8 раз, а при $n = 4$ – в 10 раз.

На рисунке 8 представлены зависимости коэффициента f_0 для вероятностей $\delta^* = 0{,}001$ и $\delta^* = 0{,}0001$ при изменении загрузки от 0,1 до 0,9.

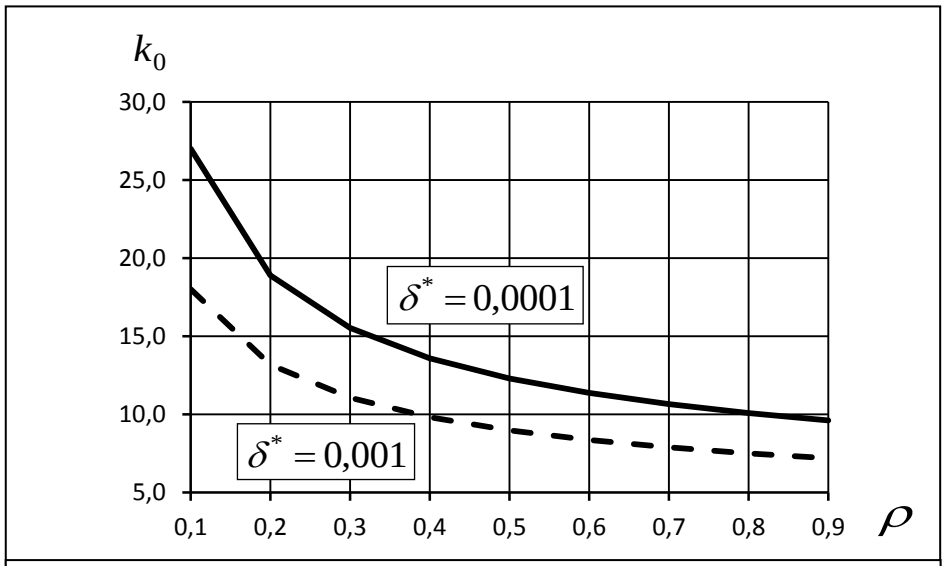

Рисунок 8. Зависимость коэффициента f_0 от загрузки

Как видно из графика, большие значения коэффициент f_0 имеет при маленькой загрузке системы, и f_0 уменьшается с увеличением загрузки ρ.

4.2.6. Определение производительности устройства с учётом ограничения на время пребывания

В двух предыдущих параграфах решались задачи оптимального проектирования систем с одним устройством. При этом к показателям эффективности не предъявлялись никакие требования. При наличии ограничений на предельные значения показателей эффективности необходимо выполнить их проверку.

Проиллюстрируем на рисунке 9 задачу определения оптимальной производительности устройства при наличии ограничения (15) на среднее время пребывания заявок в системе.

Наличие ограничения (15) позволяет выделить допустимую область значений V, точнее, минимальную производительности V'. Если при некотором заданном значении ограничения u_1^* минимальная

производительность $V_1^{'} \le V_{opt}$, то в качестве конечного результата принимается значение $V = V_{opt}$. В противном случае, если при некотором заданном значении ограничения u_2^* минимальная производительность $V_2^{'} > V_{opt}$, в качестве конечного результата следует принять значение $V = V_2^{'}$, поскольку оптимальное значение производительности не обеспечивает выполнение основного требования к системе: $u < u_2^*$.

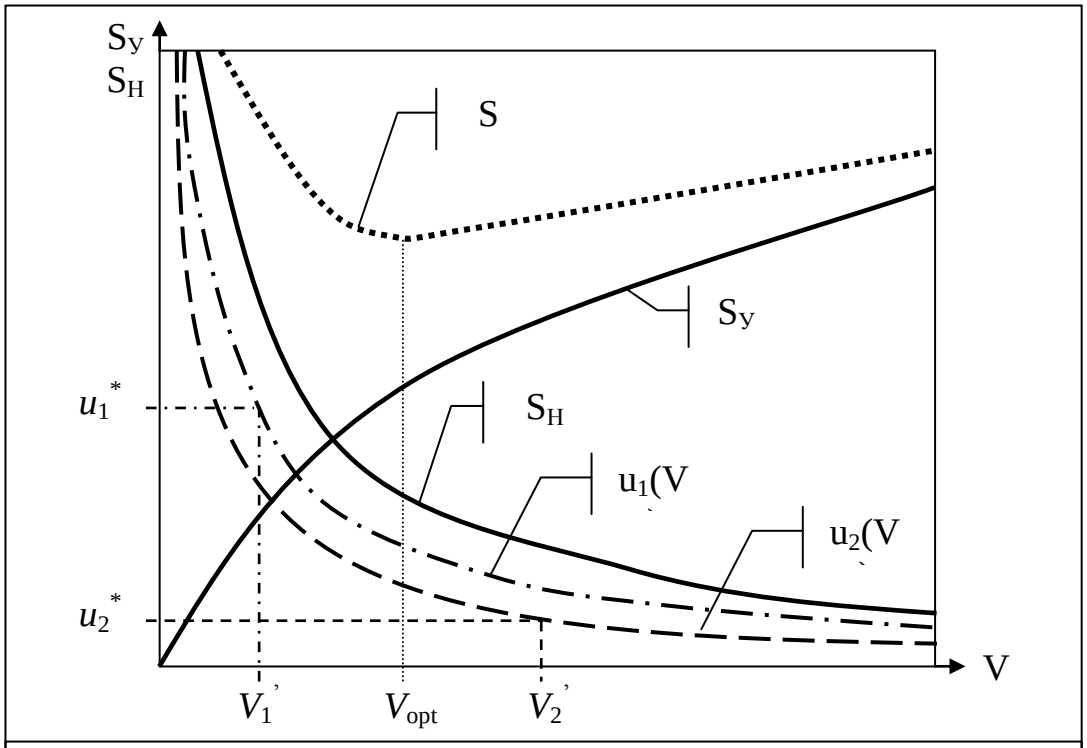

Рисунок 9. Учет ограничений на время пребывания

4.2.7. Оценка ёмкости накопителя

Одной из задач проектирования систем с использованием простейших базовых моделей является оценка ёмкости накопителя, обеспечивающей заданный уровень потерь поступающих заявок из-за переполнения накопителя.

Как и выше, для решения этой задачи необходимо знать закон распределения числа заявок в системе: $p_k = \Pr(K = k)$ – вероятность нахождения в системе ровно k заявок. Тогда задача определения ёмкости накопителя формулируется следующим образом: определить ёмкость накопителя E, при которой вероятность превышения этой ёмкости, то есть вероятность того, что количество K заявок в системе окажется больше ёмкости E накопителя, не превысит заданного значения δ^*:

$$\Pr(K > E) \le \delta^*. \qquad (21)$$

Очевидно, что

$$\Pr(K > E) = \sum_{k=E+1}^{\infty} p_k = 1 - \sum_{k=0}^{E} p_k . \qquad (22)$$

Отсюда, рассматривая задачу на границе ограничения δ^*, получим уравнение, решением которой будет минимальное значение E ёмкости накопителя:

$$\sum_{k=E+1}^{\infty} p_k = \delta^*. \qquad (23)$$

Для модели M/M/1 закон распределения числа заявок в системе определяется выражением (11). Подставляя (11) в (23), получим:

$$(1-\rho)\sum_{k=E+1}^{\infty} \rho^k = \delta^*,$$

откуда с учетом того, что сумма в последнем выражении представляет собой геометрическую прогрессию, после некоторых преобразований получим:

$$\rho^{E+1} = \delta^*.$$

Решение этого уравнения имеет вид

$$E = \left\lceil \frac{\lg \delta^*}{\lg \rho} - 1 \right\rceil. \qquad (24)$$

Если ограничение δ^* задается в виде $\delta^* = 10^{-n}$, где n принимает целочисленные значения $(n = 1, 2, ...)$, то выражение (24) может быть представлено следующим образом:

$$E = -\left\lceil \frac{n}{\lg \rho} + 1 \right\rceil.$$

На рисунке 10 показана зависимость ёмкости накопителя, рассчитанной по формуле (24) от загрузки системы при разных значениях вероятности превышения: $\delta^* = 10^{-3};\ 10^{-4};\ 10^{-5}$.

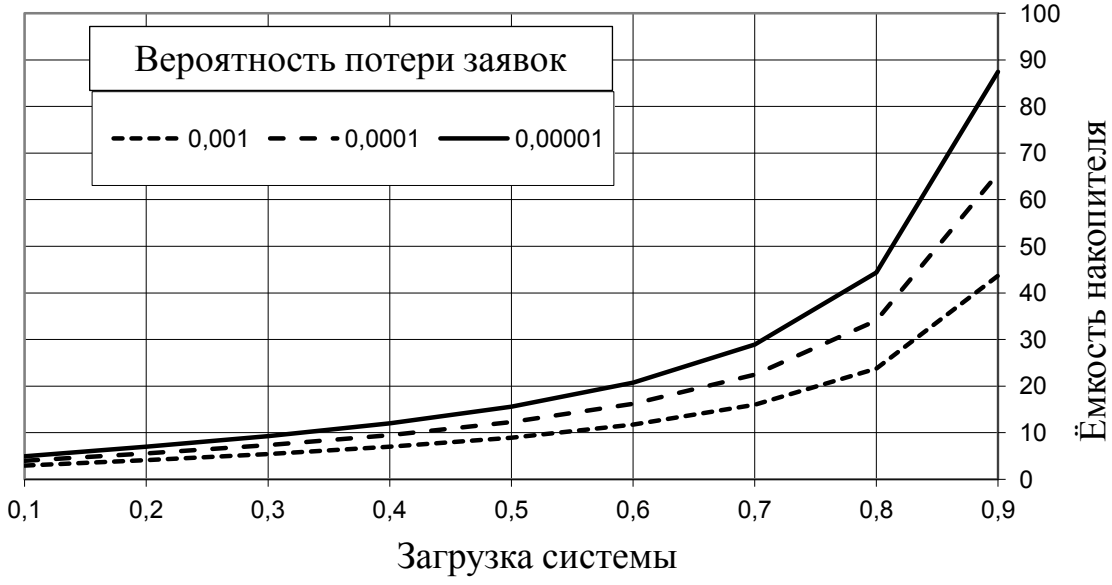

Рисунок 10. Зависимость ёмкости накопителя от загрузки системы

Как видно из представленных зависимостей, для сохранения заданного уровня превышения δ^* ёмкость накопителя существенно должна быть увеличена при увеличении загрузки, начиная со значения $\rho = 0{,}6$. В частности, при увеличении загрузки от значения $\rho = 0{,}8$ до значения $\rho = 0{,}9$ ёмкость накопителя должна быть увеличена примерно в 2 раза.

4.2.8. Определение допустимой нагрузки при заданной производительности устройства и ограничении на время пребывания заявок в системе

Под *допустимой нагрузкой* будем понимать предельную (максимальную) интенсивность поступления в систему заявок, средняя ресурсоёмкость обработки которых известна и равна θ.

Для решения поставленной задачи воспользуемся неравенством (16), откуда получим:

$$\lambda < \frac{V}{\theta} - \frac{1}{u^*} = \frac{u^* V - \theta}{u^* \theta}.$$

Для того чтобы обеспечить $\lambda > 0$, из последнего выражения вытекают необходимые условия (требования), которые должны выполняться в проектируемой системе:

$V > \frac{\theta}{u^*}$ или, что то же самое, $u^* > \frac{\theta}{V}$.

Последнее требование в виде второго неравенства становится очевидным, если учесть, что правая часть этого неравенства представляет собой среднюю длительность обработки одной заявки, то есть ограничение на время пребывания заявок в системе не может быть меньше длительности обработки одной заявки.

4.3. Системы с одним устройством и накопителем ограниченной ёмкости

Положим теперь, что система с одним устройством имеет накопитель, ёмкость которого ограничена и равна E (рисунок 11). Заявки заносятся в накопитель в порядке поступления. Если накопитель заполнен, то очередная поступившая в систему заявка получит отказ и будет потеряна. Заявки выбираются из накопителя на обслуживание в соответствии с дисциплиной FIFO, т.е. в порядке поступления. Длительность обслуживания заявок распределена по экспоненциальному закону со средним значением b и коэффициентом вариации $\nu_b = 1$.

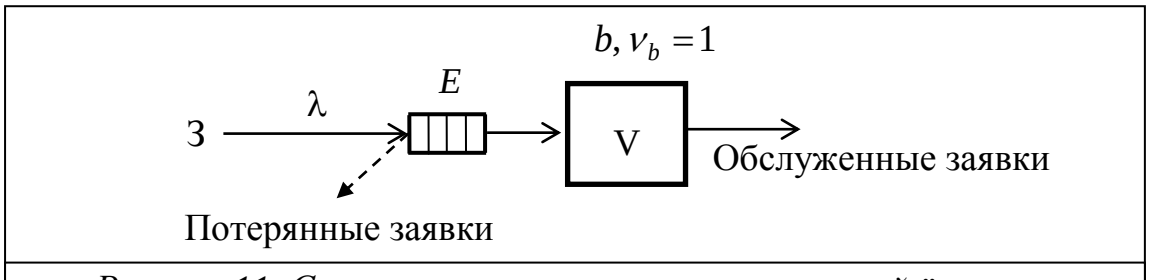

Рисунок 11. Система с накопителем ограниченной ёмкости

4.3.1. Расчёт характеристик систем с накопителем ограниченной ёмкости

Положим, что в рассматриваемую систему поступает простейший поток с интенсивностью λ, длительность обслуживания которых распределена по экспоненциальному закону (коэффициент вариации $\nu_b = 1$) со средним значением b (модель M/M/1/E), причём нагрузка, в отличие от предыдущей модели, может принимать любые положительные значения: $y = \lambda b > 0$.

Рассчитаем вероятности состояний системы в случае ограниченной ёмкости накопителя. Для этого воспользуемся выражением (11). Поскольку число заявок в системе не может превышать значения $(E+1)$, то число состояний будет ограничено, и тогда вероятность состояния k, означающего, что в системе находится ровно k заявок:

$$p_k = A y^k (1-y) \quad (k = 0, 1, 2, \ldots, E+1),$$

где A – нормировочный коэффициент, определяемый из условия:

$$\sum_{k=0}^{E+1} p_k = 1,$$

откуда получим:

$$A = \left[(1-y) \sum_{k=0}^{E+1} y^k \right]^{-1} = \left[1 - y^{E+2} \right]^{-1}.$$

Тогда вероятности состояний системы в случае ограниченной ёмкости накопителя могут быть рассчитаны по формуле:

$$p_k = \frac{y^k (1-y)}{1-y^{E+2}} \quad (k = 0, 1, 2, \ldots, E+1). \qquad (25)$$

Вероятность потери заявки из-за ограниченной ёмкости накопителя равна вероятности того, что в системе находится $(E+1)$ заявка, т.е. система заполнена:

$$\pi_{\text{п}} = p_{E+1} = \frac{y^{E+1}(1-y)}{1-y^{E+2}} \qquad (y \neq 1).$$

Для случая $y = 1$ путем предельного перехода получим:

$$\pi_{\text{п}} = \frac{1}{E+2} \qquad (y = 1).$$

Таким образом, вероятность потери заявки рассчитывается по формуле:

$$\pi_{\text{п}} = \begin{cases} \dfrac{y^{E+1}(1-y)}{1-y^{E+2}}, & y \neq 1; \\ \dfrac{1}{E+2}, & y = 1. \end{cases} \qquad (26)$$

При расчёте загрузки системы следует учитывать только обслуженные заявки: $\rho = \lambda_{\text{о}} b$, где $\lambda_{\text{о}} = (1-\pi_{\text{п}})\lambda$ – интенсивность потока обслуженных заявок, откуда с учетом (26) после некоторых преобразований окончательно получим:

$$\rho = \begin{cases} \dfrac{y(1-y^{E+1})}{1-y^{E+2}}, & y \neq 1; \\ \dfrac{E+1}{E+2}, & y = 1. \end{cases} \qquad (27)$$

Среднее число заявок в системе определяется через вероятности состояний:

$$m = \sum_{k=1}^{E+1} k\, p_k = \frac{1-y}{1-y^{E+2}} \sum_{k=1}^{E+1} k\, y^k ,$$

откуда после некоторых преобразований получим:

$$m = \begin{cases} \dfrac{y}{1-y^{E+2}} \left[\dfrac{1-y^{E+1}}{1-y} - (E+1) y^{E+1} \right], & y \neq 1; \\ \dfrac{E+1}{2}, & y = 1. \end{cases} \qquad (28)$$

Остальные характеристики могут быть рассчитаны с использованием фундаментальных соотношений $(1)-(3)$, в частности, среднее время пребывания заявок в системе:

$$u = \frac{m}{(1-\pi_{\text{п}})\lambda} = \frac{m}{(1-\pi_{\text{п}})y} b ,$$

Подставляя (26) и (28) в последнее выражение, после некоторых преобразований получим:

$$u = \begin{cases} \dfrac{1-(E+2)y^{E+1}+(E+1)y^{E+2}}{(1-y)(1-y^{E+1})} b, & y \neq 1; \\ \dfrac{E+2}{2} b, & y = 1. \end{cases} \qquad (29)$$

Формулы (25) - (28) для расчёта характеристик функционирования систем с одним устройством и накопителем ограниченной ёмкости получены в предположении, что обслуживаемая в устройстве заявка не занимает места в накопителе. В то же время во многих реальных системах, построенных, например, на основе компьютера, обрабатываемый запрос, как и ожидающие обработки, находится в накопителе (памяти) системы. Для таких систем число состояний уменьшится на единицу и формулы (25) - (28) примут вид:

$$p_k = \frac{y^k(1-y)}{1-y^{E+1}} \quad (k = 0, 1, 2, \ldots, E);$$

$$\pi_{\text{п}} = \begin{cases} \dfrac{y^E(1-y)}{1-y^{E+1}}, & y \neq 1; \\ \dfrac{1}{E+1}, & y = 1; \end{cases} \qquad (30)$$

$$\rho = \begin{cases} \dfrac{y(1-y^E)}{1-y^{E+1}}, & y \neq 1; \\ \dfrac{E}{E+1}, & y = 1; \end{cases} \tag{31}$$

$$m = \begin{cases} \dfrac{y}{1-y^{E+1}}\left[\dfrac{1-y^E}{1-y} - E y^E\right], & y \neq 1; \\ \dfrac{E}{2}, & y = 1; \end{cases} \tag{32}$$

$$u = \begin{cases} \dfrac{1-(E+1)y^E + E y^{E+1}}{(1-y)(1-y^E)} b, & y \neq 1; \\ \dfrac{E+1}{2} b, & y = 1. \end{cases} \tag{33}$$

4.3.2. Анализ свойств системы с накопителем ограниченной ёмкости

Положим, что в систему с одним устройством поступает *простейший* поток заявок с интенсивностью λ, длительность обслуживания которых распределена по *экспоненциальному* закону (коэффициент вариации $\nu_b = 1$) со средним значением b. В качестве модели используется система с накопителем ограниченной ёмкости (M/M/1/E), причём нагрузка системы может принимать любые положительные значения: $y = \lambda b > 0$.

Вероятность потери заявки из-за ограниченной ёмкости накопителя определяется выражением (26), которое запишем в несколько ином виде:

$$\pi_{\text{п}} = \begin{cases} \dfrac{y^{E+1}(y-1)}{y^{E+2}-1}, & y \neq 1 \\ \dfrac{1}{E+2}, & y = 1. \end{cases}$$

На рисунке 12 показаны зависимости загрузки системы ρ и вероятности потери заявок $\pi_{\text{п}}$ от нагрузки y при ёмкости накопителя $E = 5$.

При увеличении нагрузки до бесконечности ($y \to \infty$) получим вполне ожидаемый результат – вероятность потери заявок стремится к единице:

$$\pi_{\text{п}} = \lim_{y\to\infty} \frac{1-1/y}{1-1/y^{E+2}} = 1.$$

При увеличении ёмкости накопителя до бесконечности ($E \to \infty$) вероятность потери заявок:

$$\pi_{\text{п}} = \lim_{E\to\infty} \frac{y-1}{y-1/y^{E+1}} = \begin{cases} 0, & y \leq 1 \\ 1-\dfrac{1}{y}, & y > 1. \end{cases}$$

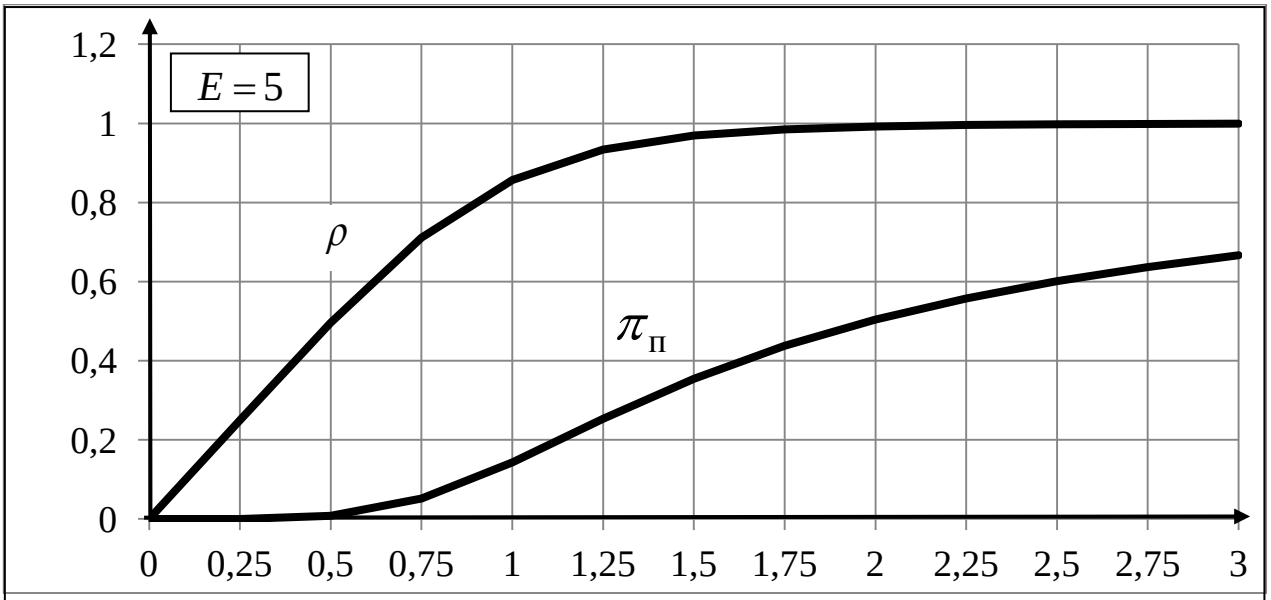

Рисунок 12 . Зависимости загрузки системы и вероятности потери заявок от нагрузки

Загрузка системы рассчитывается по формуле (27), а среднее число заявок в системе и среднее время пребывания заявок в системе – по формулам (28) и (29) соответственно.

При условии, что нагрузка системы $y \neq 1$, определим ёмкость накопителя, при которой вероятность потери заявок $\pi_{\text{п}}$ не превысит значения π^*:

$$\frac{y^{E+1}(1-y)}{1-y^{E+2}} \leq \pi^*.$$

После некоторых преобразований получим:

$$E \geq \log_y \frac{\pi^*}{1-(1-\pi^*)y} - 1.$$

Из последнего выражения могут быть получены ограничения, налагаемые на значение:

- нагрузки y при заданной вероятности потери заявок π^* (рисунок 13,а): $y \leq \frac{1}{1-\pi^*}$ (правая часть представляет собой верхнюю границу нагрузки, при которой вероятность потери заявок $\pi_{\text{п}}$ равна заданному значению π^*),
- вероятности потери заявок $\pi_{\text{п}}$ при заданной нагрузке системы y^* (рисунок 13,б): $\pi_{\text{п}} \geq 1 - \frac{1}{y^*}$ (правая часть представляет собой нижнюю границу

вероятности потери заявок, при которой может быть реализована заданная нагрузка y^*).

На рисунке 13 заштрихованы области допустимых значений нагрузки (рисунок 13,а) и вероятности потери заявок (рисунок 13,б).

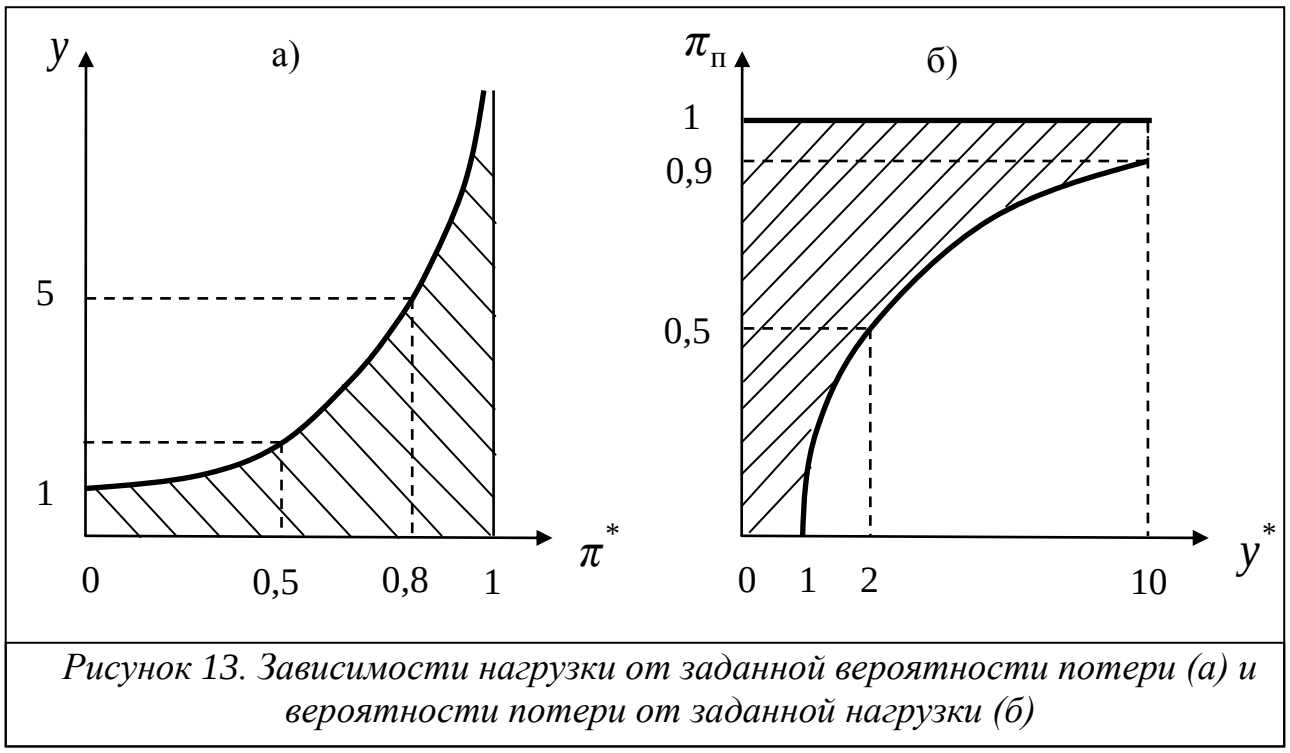

Рисунок 13. Зависимости нагрузки от заданной вероятности потери (а) и вероятности потери от заданной нагрузки (б)

При нагрузке системы $y=1$ из условия $\frac{1}{E+2}\le\pi^*$ найдем ёмкость накопителя, при которой вероятность потери заявок $\pi_{\text{п}}$ не превысит заданного значения π^*:

$$E\ge\frac{1}{\pi^*}-2.$$

Таким образом, при заданном ограничении на вероятность потери заявок нижняя граница ёмкости накопителя может быть рассчитана как

$$E=\begin{cases}\left\lceil \log_y\dfrac{\pi^*}{1-(1-\pi^*)y}-1\right\rceil, & y\ne 1;\\ \left\lceil\dfrac{1}{\pi^*}-2\right\rceil, & y=1,\end{cases} \qquad (34)$$

где $\lceil x\rceil$ означает округление x до целого в большую сторону.

Особый интерес представляют высоконагруженные и перегруженные системы, нагрузка которых соответственно близка к единице и, превышая единицу, стремится к бесконечности $(y\to\infty)$. В последнем случае вероятность потери заявок при ограниченной ёмкости накопителя будет стремиться к 1: $\pi_{\text{п}}\to 1$.

Рассмотрим теперь случай, когда нагрузка больше 1, а ёмкость накопителя увеличивается до бесконечности ($y>1;\ E\to\infty$). Тогда вероятность потери заявок:

$$\pi_{\text{п}}=\lim_{E\to\infty}\frac{y^{E+1}(1-y)}{1-y^{E+2}}=\frac{y-1}{y}.$$

В таблице 1 представлены значения вероятности потерь заявок $\pi_{\text{п}}$ и интенсивности обслуженных заявок $\lambda_{\text{вых}}$ при различных значениях нагрузки y (интенсивность поступления заявок в систему равна 1) в перегруженных системах. При этом загрузка всех систем $\rho=1$.

Таблица 1. Вероятности потерь в перегруженных системах

y	2	5	10	50	100	1000
$\pi_{\text{п}}$	0,5	0,8	0,9	0,98	0,99	0,999
$\lambda_{\text{вых}}$	0,5	0,2	0,1	0,02	0,01	0,001
ρ	1,0	1,0	1,0	1,0	1,0	1,0

4.3.3. Проектирование систем с накопителем ограниченной ёмкости

Для систем с накопителем ограниченной ёмкости одной из основных характеристик, определяющих эффективность функционирования, является вероятность потери заявок $\pi_{\text{п}}$ из-за переполнения накопителя. Очевидно, что эта вероятность зависит от двух параметров системы: ёмкости накопителя E и производительности устройства V: $\pi_{\text{п}}=f(E,V)$, причем, чем больше ёмкость E и производительность V, тем меньше вероятность потери заявок $\pi_{\text{п}}$.

На рисунке 14 показан характер зависимостей вероятности потери заявок $\pi_{\text{п}}$ от ёмкости накопителя E (рисунок 14,а) и производительности устройства V (рисунок 14,б).

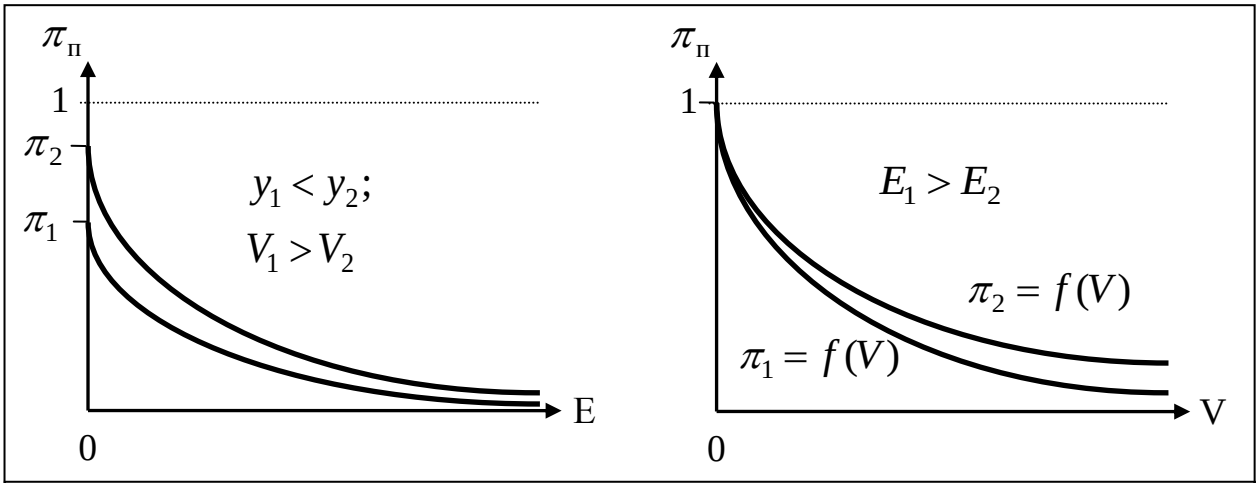

Рисунок 14. Зависимость вероятности потерь от ёмкости (а) и производительности (б) системы

Легко убедиться, что вероятность потери заявок $\pi_{\text{п}}$ при отсутствии накопителя ($E=0$) определяется как $\pi_i = \dfrac{\lambda_i}{\lambda_i + \mu_i} = \dfrac{y_i}{1+y_i}$ $(i=1,2)$. Отсюда следует очевидный вывод: чем больше нагрузка, создаваемая в системе поступающими заявками, тем больше вероятность потери заявок (рисунок 14,а). Поскольку нагрузка y_i связана с производительностью устройства V зависимостью $y_i = \dfrac{\lambda_i \theta_i}{V}$, можно сделать вывод, что с увеличением производительности уменьшается вероятность потери заявок $\pi_{\text{п}}$, что иллюстрируется вторым графиком (рисунок 14,б). Здесь же показано, что вероятность потери заявок при одной и той же производительности меньше для системы с накопителем большей ёмкости.

Таким образом, вероятность потери заявок в системе можно уменьшить за счёт увеличения производительности V устройства и ёмкости E накопителя. Однако следует учитывать, что при этом увеличивается стоимость системы, которая может быть рассчитана следующим образом:

$$S = \xi V^{\chi} + s_0 E,$$

где $S_{\text{У}} = \xi V^{\chi}$ – стоимость устройства, связанная пропорциональной зависимостью с его производительностью (ξ и χ – стоимостные коэффициенты пропорциональности и нелинейности соответственно); $S_{\text{Н}} = s_0 E$ – стоимость накопителя (s_0 – стоимость единицы памяти).

В общем случае задача проектирования системы с одним устройством и накопителем ограниченной ёмкости может быть сформулирована в следующих двух постановках:

- определить производительность V устройства и ёмкость E накопителя, обеспечивающие выполнение ограничения на вероятность потери заявок $\pi_{\text{п}} < \pi^*$ при минимальной стоимости системы $S - \min$;
- определить производительность V устройства и ёмкость E накопителя, обеспечивающие выполнение ограничения на стоимость системы $S < S^*$ при минимальной вероятности потери заявок $\pi_{\text{п}} - \min$.

Для решения задачи проектирования в двух представленных постановках используются аналитические зависимости (25) – (28) или (30) – (33). Однако, получить решение задачи оптимального синтеза в явном аналитическом виде не представляется возможным. Одним из подходов к решению задачи в связи с небольшим количеством оптимизируемых параметров (производительность устройства V и ёмкость накопителя E) может быть последовательный перебор значений ёмкости накопителя E. При этом для каждого значения E определяется минимальная производительность устройства V, при которой выполняются заданные ограничения

Одной из важных характеристик функционирования технических систем является время пребывания заявок в системе, на среднее значение которого

тоже может налагаться ограничение в виде: $u < u^*$. Если в результате проектирования это ограничение не выполняется, необходимо увеличить производительность устройства, что приведёт к увеличению стоимости системы. При этом вероятность потери заявок уменьшится и, следовательно, можно попытаться уменьшить стоимость системы за счёт уменьшения ёмкости накопителя.

Таким образом, к основным характеристикам, определяющим эффективность функционирования системы с накопителем ограниченной ёмкости, относятся:

- вероятность потери заявок $\pi_{\text{п}}$;
- среднее время пребывания заявок в системе u;
- стоимость системы S.

При отсутствии ограничений на характеристики функционирования в качестве обобщённого критерия эффективности может использоваться функция следующего вида:

$$F = k_1\pi_{\text{п}} + k_2 u + k_3 S, \qquad (35)$$

где k_1, k_2, k_3 - весовые коэффициенты, определяющие степень ценности (важности) соответствующей характеристики для проектируемой системы и удовлетворяющие условию $k_1 + k_2 + k_3 = 1$.

В этом случае задача проектирования формулируется следующим образом: определить ёмкость накопителя E и производительность устройства V, при которых критерий эффективности (35) принимает минимальное значение: $F - \min$.

На рисунке 15 иллюстрируется задача определения оптимальных значений ёмкости накопителя и производительности устройства, обеспечивающих минимум критерия эффективности. Здесь представлены графики, характеризующие зависимость вероятности потерь $\pi_{\text{п}}(E,V)$, среднего времени пребывания заявок в системе $u(E)$ и $u(V)$, стоимости системы $S_{\text{у}} = \xi V$ (предполагается, что стоимостной коэффициент нелинейности $\chi = 1$) и критерия эффективности $F = k_1\pi_{\text{п}} + k_2 u + k_3 S$ от вектора оптимизируемых параметров (E,V). Значения $(E,V)_{\text{opt}}$, соответствующие минимуму критерия эффективности F, являются оптимальными.

Отметим, что зависимости среднего времени пребывания заявок в системе $u(E)$ и $u(V)$, представленные на рисунке 15 пунктирными линиями, ведут себя по разному при увеличении соответственно ёмкости накопителя E и производительности устройства V: среднее время пребывания заявок в системе растёт с увеличением ёмкости накопителя и уменьшается с увеличением производительности устройства. При этом зависимость $u(E,V)$ от вектора (E,V) может иметь различный характер – возрастающий, убывающий или любой другой.

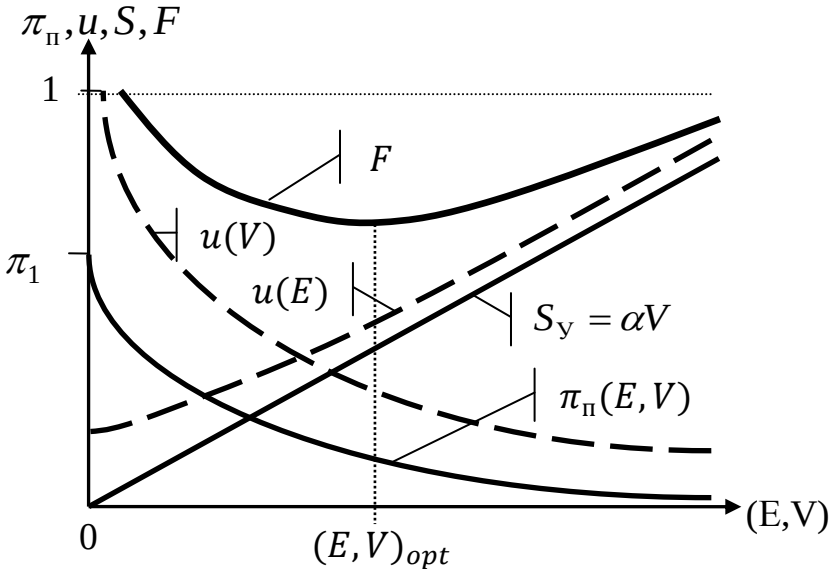

Рисунок 15. К задаче оптимизации ёмкости накопителя и производительности системы

4.3.1. Сравнительный анализ систем с накопителями ограниченной и неограниченной ёмкости

Анализ представленных выше математических зависимостей характеристик функционирования систем с одним устройством свидетельствует о том, что более простым и, следовательно, более быстрым для проведения оценочных расчётов, является расчёт систем с накопителем неограниченной ёмкости. Это делает целесообразным использование модели с накопителем неограниченной ёмкости даже в тех случаях, когда моделируемая система имеет накопитель ограниченной ёмкости. Можно предположить, что такая замена оправдана, если ёмкости накопителя в моделируемой системе достаточно большая, при этом мала вероятность её переполнения и, следовательно, потери заявок незначительны. При этом возникает вопрос: при каких условиях такая замена адекватна и насколько полученные характеристики будут отличаться от характеристик реальной системы?

Для ответа на поставленный вопрос выполним сравнительный анализ систем с одним устройством и накопителями ограниченной и неограниченной ёмкости соответственно.

Очевидно, что такая замена возможна только в том случае, если система не перегружена. Поэтому положим, что нагрузка системы с накопителем ограниченной ёмкости ($E < \infty$) не превышает единицы: $y < 1$. По формуле (34) определим нижнюю границу ёмкости накопителя $E(\pi^*)$, при которой вероятность потери заявок не превысит значений $\pi^* = 10^{-3}; 10^{-4}$ и 10^{-5} при изменении нагрузки y в интервале от 0,5 до 0,999. Для полученных систем M/M/1/E рассчитаем характеристики функционирования, в качестве которых будем рассматривать средние значения времени ожидания ($w_{\text{M/M/1/E}}$) и времени

пребывания ($u_{\text{M/M/1/E}}$) заявок. Эти же характеристики ($w_{\text{M/M/1}}$и $u_{\text{M/M/1}}$) рассчитаем в предположении о неограниченной ёмкости накопителя, используя в качестве модели систему M/M/1.

В таблицах 2, 3 и 4 представлены результаты расчёта характеристик ($w_{\text{M/M/1/E}}$ и $u_{\text{M/M/1/E}}$) систем M/M/1/E с накопителями ограниченной ёмкости $E(10^{-3})$, $E(10^{-4})$ и $E(10^{-5})$ соответственно и систем M/M/1 с накопителями неограниченной ёмкости ($w_{\text{M/M/1}}$ и $u_{\text{M/M/1}}$) при различных значениях нагрузки системы y и заданной вероятности потерь π^*. Для оценки погрешностей результатов расчёта характеристик функционирования (среднего времени пребывания заявок в системе и среднего времени ожидания), возникающих при замене системы с накопителем ограниченной ёмкости моделью с накопителем неограниченной ёмкости, используются следующие величины (для простоты принято, что среднее время обслуживания заявок равно 1):

$$\delta_u\% = \frac{u_{\text{M/M/1}} - u_{\text{M/M/1/E}}}{u_{\text{M/M/1/E}}}100\% \quad \text{и} \quad \delta_w\% = \frac{w_{\text{M/M/1}} - w_{\text{M/M/1/E}}}{w_{\text{M/M/1/E}}}100\%.$$

Таблица 2. Сравнение систем с вероятностью потери заявок $\pi^* = 10^{-3}$

y	$E(10^{-3})$	$u_{\text{M/M/1}}$	$u_{\text{M/M/1/E}}$	$\delta_u\%$	$w_{\text{M/M/1}}$	$w_{\text{M/M/1/E}}$	$\delta_w\%$
0,5	8	2,0	1,98	1%	1,0	0,98	2%
0,8	23	5,0	4,886	2,3%	4,0	3,886	2,9%
0,9	43	10,0	9,569	4,3%	9,0	8,569	5%
0,95	76	20	18,488	7,6%	19	17,488	8,6%
0,99	241	100	76,694	23,3%	99	75,694	30,8%

Как видно из таблицы 2, погрешность, возникающая при расчёте характеристик функционирования в результате замены системы с накопителем ограниченной ёмкости ($E < \infty$) системой с накопителем неограниченной ёмкости ($E = \infty$), при условии, что вероятность потери заявок в исходной системе менее 10^{-3}, не превышает 5% в области нагрузок от 0 до 0,9 и 10% – в области нагрузок до 0,95. При этом погрешность расчёта среднего времени пребывания заявок в системе меньше погрешности расчёта среднего времени ожидания заявок: $\delta_u\% < \delta_w\%$.

Погрешности $\delta_u\%$ и $\delta_w\%$, возникающие при условии, что вероятность потери заявок в исходной системе менее 10^{-4} (таблица 3), не превышают 5% в области нагрузок от 0 до 0,99. Как и выше, погрешность расчёта среднего времени пребывания заявок в системе меньше погрешности расчёта среднего времени ожидания заявок: $\delta_u\% < \delta_w\%$.

Таблица 3. Сравнение систем с вероятностью потери заявок $\pi^* = 10^{-4}$

y	$E(10^{-4})$	$u_{\text{M/M/1}}$	$u_{\text{M/M/1/E}}$	$\delta_u\%$	$w_{\text{M/M/1}}$	$w_{\text{M/M/1/E}}$	$\delta_w\%$
0,5	12	2,0	1,998	0,1%	1,0	0,998	0,2%
0,8	34	5,0	4,986	0,3%	4,0	3,986	0,4%
0,9	66	10,0	9,942	0,6%	9,0	8,942	0,7%
0,95	122	20	19,776	1,2%	19	18,776	1,2%
0,99	464	100	95,616	4,6%	99	94,616	4,7%
0,999	2443	1000	767,97	30,2%	999	766,97	30,3%

Погрешности $\delta_u\%$ и $\delta_w\%$, возникающие при условии, что вероятность потери заявок в исходной системе менее 10^{-5} (таблица 4), не превышают 5% в области нагрузок от 0 до 0,999.

Таким образом, использование системы M/M/1 для расчёта характеристик функционирования систем с накопителями ограниченной ёмкости оправдано и позволяет получить верхнюю границу характеристик функционирования (среднего времени ожидания и среднего времени пребывания) с приемлемой погрешностью менее 5% в широкой области изменения нагрузки, а именно: если вероятность потери заявок в реальной системе менее 10^{-3} и нагрузка не превышает 0,8, а если вероятность потери менее 10^{-5}, то нагрузка системы может достигать значения 0,99. В то же время следует помнить, что использование системы M/M/1 не позволяет оценить вероятность потери заявок.

Таблица 4. Сравнение систем с вероятностью потери заявок $\pi^* = 10^{-5}$

y	$E(10^{-5})$	$u_{\text{M/M/1}}$	$u_{\text{M/M/1/E}}$	$\delta_u\%$	$w_{\text{M/M/1}}$	$w_{\text{M/M/1/E}}$	$\delta_w\%$
0,5	15	2,0	2,0	0%	1,0	1,0	0%
0,8	47	5,0	4,999	0,02%	4	3,999	0,025%
0,9	93	10,0	9,995	0,05%	9	8,995	0,06%
0,95	179	20	19,982	0,09%	19	18,982	0,1%
0,99	756	100	99,624	0,4%	99	98,624	0,4%
0,999	5300	1000	973,5	2,7%	999	972,5	2,8%
0,9999	23977	10000	7602,3	31,5%	9999	7601,3	31,5%

4.4. Системы с несколькими устройствами и накопителем неограниченной ёмкости

В системах с несколькими устройствами поступающие заявки могут обрабатываться любым свободным устройством. Если все устройства заняты, то заявки ожидают в накопителе освобождения устройства. При этом могут

использоваться два разных способа организации накопителей перед устройствами:

1) накопители заявок формируются перед каждым устройством системы, при этом поступившая в систему заявка случайным образом направляется в один из накопителей; такие системы называются системами с индивидуальными накопителями и представляются в виде совокупности одноканальных СМО, число которых равно числу устройств и, следовательно, числу накопителей (рисунок 16,а);

2) перед всеми устройствами системы формируется один общий накопитель, в который заносятся все поступающие заявки, при этом заявки из накопителя на обработку поступают в первое освободившееся устройство; такие системы называются системами с общим накопителем и представляются в виде многоканальной СМО, в которой число обслуживающих приборов равно числу устройств в системе (рисунок 16,б).

К системам с индивидуальными накопителями относятся многомашинные вычислительные комплексы, а к системам с общим накопителем – многопроцессорные вычислительные комплексы, моделями которых могут служить системы, представленные на рисунке 16.

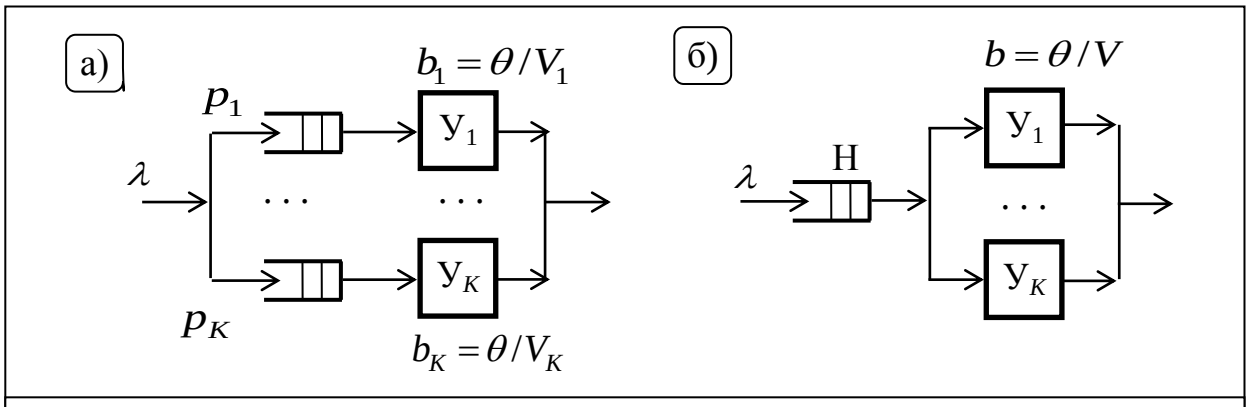

Рисунок 16. Системы с индивидуальными накопителями (а) и общим накопителем (б) неограниченной ёмкости

4.4.1. Расчёт характеристик систем с индивидуальными накопителями неограниченной ёмкости

Системы с несколькими устройствами и с индивидуальными накопителями неограниченной ёмкости представляются в виде совокупности одноканальных систем массового обслуживания (рисунок 16, а). Поступающие с интенсивностью λ в систему заявки с вероятностью p_i $(i = \overline{1, K})$ направляются к устройству с номером i, причём должно выполняться условие:

$$\sum_{i=1}^{K} p_i = 1.$$

Это условие означает, что поступившая заявка с вероятностью 1 попадает в один из K накопителей.

Интенсивность поступления заявок к устройству i определяется как $\lambda_i = p_i \lambda$. В дальнейшем расчёт характеристик системы сводится к расчёту K

независимых одноканальных СМО с однородным потоком заявок, как это описано в пункте 4.2.1, причём не исключается и вариант, при котором разные устройства имеют разные производительности V_i $(i=\overline{1,K})$, следовательно, затрачивают разное время $b_i=\theta/V_i$ $(i=\overline{1,K})$ на обработку заявок, то есть $b_i \neq b_j$ $(i \neq j;\ i,j=\overline{1,K})$.

В случае равновероятного распределения заявок по K накопителям, интенсивность поступления заявок ко всем устройствам системы одинакова и равна $\lambda_i=\frac{\lambda}{K}$. Если при этом все устройства идентичны, то расчёт характеристик системы сводится к расчёту одной одноканальной СМО с однородным потоком заявок, поскольку характеристики всех устройств будут одинаковы. В противном случае, если разные устройства имеют разные производительности $b_i \neq b_j$ $(i \neq j;\ i,j=\overline{1,K})$, расчёт выполняется для всех K одноканальных СМО.

4.4.2. Расчёт характеристик систем с общим накопителем неограниченной ёмкости

В качестве моделей систем с несколькими устройствами и с общим накопителем используются многоканальные модели (СМО), содержащие K идентичных устройств и накопитель неограниченной ёмкости. В систему поступает поток заявок с интенсивностью λ (рисунок 16,б). Длительность обработки заявок является случайной величиной со средним значением b. Выбор заявок из накопителя на обработку осуществляется в соответствии с дисциплиной обслуживания в порядке поступления (ОПП) по правилу «первым пришёл – первым обслужен» (FIFO – First In First Out). При этом если в момент поступления какой-то заявки свободны несколько устройств, то поступившая заявка случайным образом направляется к одному из свободных устройств.

В качестве основной характеристики функционирования системы будем рассматривать среднее время ожидания заявок в накопителе.

Точный метод расчёта характеристик обслуживания заявок в многоканальной модели разработан при следующих предположениях:

- поток заявок – простейший;
- длительность обработки заявок в каждом из устройств распределена по экспоненциальному закону со средним значением $b=\frac{\theta}{V}$, где θ – средняя ресурсоёмкость обработки одного запроса; V – производительность одного устройства многоканальной модели;
- все K устройств – идентичны, и любая заявка может быть обслужена любым устройством;
- ёмкость накопителя – не ограничена;

- в системе отсутствуют перегрузки, то есть загрузка системы меньше единицы: $\rho = \frac{\lambda b}{K} = \frac{\lambda \theta}{KV} < 1$ или, что то же самое, нагрузка системы меньше числа устройств: $y < K$.

Получим аналитические зависимости для расчёта характеристик функционирования системы с использованием метода марковских процессов.

На основе графа переходов марковского процесса построим систему линейных алгебраических уравнений для вероятностей состояний:

$$\begin{cases} y\, p_0 = p_1; \\ (y+k)\, p_k = y\, p_{k-1} + (k+1)\, p_{k+1} \qquad (k = \overline{1, K-1}); \\ (y+K)\, p_k = y\, p_{k-1} + K\, p_{k+1} \qquad (k = \overline{K, \infty}), \end{cases}$$

где p_k – вероятность состояния k (нахождения в системе ровно k заявок); $y = \lambda / \mu = \lambda b$ – нагрузка, создаваемая в системе; $\mu = 1/b$ – интенсивность обслуживания заявок одним устройством.

Отсюда легко получить, что вероятность состояния k определяется как

$$p_k = \begin{cases} \frac{y^k}{k!}\, p_0 \qquad (k = \overline{1, K-1}); \\ \frac{K^K}{K!}\left(\frac{y}{K}\right)^k p_0 \quad (k = \overline{K, \infty}). \end{cases}$$

Используя нормировочное условие

$$\sum_{k=0}^{\infty} p_k = 1,$$

найдём вероятность того, что в системе нет ни одной заявки, т.е. система простаивает:

$$p_0 = \left[\sum_{k=0}^{K-1} \frac{y^k}{k!} + \frac{K^K}{K!} \sum_{k=K}^{\infty} \left(\frac{y}{K}\right)^k \right]^{-1}$$

или в другой записи:

$$p_0 = \begin{cases} \left[\sum_{k=0}^{K-1} \frac{y^k}{k!} + \frac{y^K}{(K-1)!(K-y)} \right]^{-1}, & y < K; \\ 0, & y \geq K \ . \end{cases} \qquad (36)$$

Вероятность P того, что все K устройств заняты обслуживанием заявок, определяется как:

$$P = \sum_{k=K}^{\infty} p_k = \sum_{k=K}^{\infty} \frac{K^K}{K!} \left(\frac{y}{K}\right)^k p_0 = \begin{cases} \frac{y^K}{(K-1)!(K-y)}\, p_0, & y < K; \\ 1, & y \geq K \ . \end{cases} \qquad (37)$$

Очевидно, что случай $y \geq K$ означает перегрузку системы, при которой система не справляется с работой и все характеристики, такие как время

ожидания, число заявок в очереди и т.п., с увеличением времени работы системы будут стремиться к бесконечности. Поэтому ниже представлены только зависимости для расчёта характеристик систем, работающих без перегрузки, нагрузка которых не превышает количества устройств: $y < K$.

Средняя длина очереди заявок определяется через вероятности состояний:

$$l = \sum_{k=1}^{\infty} k\, p_{K+k} = \frac{y^K}{K!}\, p_0 \sum_{k=1}^{\infty} k \left(\frac{y}{K}\right)^k .$$

С учётом того, что

$$\sum_{k=1}^{\infty} k \left(\frac{y}{K}\right)^k = \frac{y/K}{(1-y/K)^2} = \frac{Ky}{(K-y)^2},$$

окончательно получим:

$$l = \sum_{k=1}^{\infty} k\, p_{K+k} = \frac{y^{K+1}}{(K-1)!(K-y)^2}\, p_0 . \qquad (38)$$

Среднее время ожидания заявок в системе в соответствии с формулой Литтла (3) и с учётом (36) будет равно:

$$w = \frac{l}{\lambda} = \frac{y^K b}{(K-1)!(K-y)^2}\, p_0 = \frac{Pb}{K-y} . \qquad (39)$$

Представленные выше выражения (36) – (39) показывают зависимость рассматриваемых характеристик от нагрузки многоканальной системы $y = \lambda b$, причём $y < K$.

Эти же зависимости могут быть представлены как функции загрузки системы $\rho = \frac{y}{K} < 1$ путём замены $y = K\rho$.

В этом случае среднее время ожидания заявок равно:

$$w = \frac{Pb}{K(1-\rho)} , \qquad (40)$$

где P – вероятность того, что все K устройств заняты обслуживанием заявок.

Вероятность P определяется как:

$$P = \frac{(K\rho)^K}{K!(1-\rho)}\, p_0 , \qquad (41)$$

где p_0 – вероятность простоя многоканальной СМО, то есть вероятность того, что в системе нет заявок:

$$p_0 = \left[\frac{(K\rho)^K}{K!(1-\rho)} + \sum_{i=0}^{K-1} \frac{(K\rho)^i}{i!} \right]^{-1} .$$

Остальные характеристики могут быть рассчитаны с использованием фундаментальных соотношений (1) – (3).

4.4.3. Определение минимального количества устройств в системе

Под минимальным количеством устройств в системе будем понимать такое количество, при котором в системе отсутствуют перегрузки.

Нагрузка y и загрузка ρ системы с несколькими устройствами определяются соответственно как:

$$y = \lambda b = \frac{\lambda\theta}{V}; \qquad \rho = \min\left(\frac{y}{K}; 1\right).$$

Для того чтобы в системе с несколькими устройствами не было перегрузок, необходимо выполнение условия: $\frac{y}{K} < 1$, из которого следует:

$$KV > \lambda\theta. \tag{42}$$

Выражение (42) можно рассматривать как ограничение, налагаемое на суммарную производительность $V_\Sigma = KV$ системы, обеспечивающее отсутствие перегрузок в системе. Величина $V_0 = \lambda\theta$ представляет собой минимально возможную суммарную производительность системы. Если суммарная производительность системы V_Σ будет меньше или равна V_0, то в системе возникнут перегрузки, что приведёт со временем к неограниченному возрастанию длины очереди заявок перед устройствами.

Для систем с идентичными устройствами на основе выражения (42) можно определить количество параллельных устройств K в системе при заданной производительности V одного устройства:

$$K > \frac{\lambda\theta}{V}. \tag{43}$$

Выражение (43) определяет ограничение, налагаемое на количество устройств с известной производительностью. Величина, определяемая как ближайшее большее целое по отношению к $\frac{\lambda\theta}{V}$, представляет собой минимальное количество устройств, необходимое для того, чтобы система справлялась с заданной нагрузкой.

Кроме того задача проектирования может заключаться в определении производительности V одного устройства при заданном количестве устройств K в системе:

$$V > \frac{\lambda\theta}{K}. \tag{44}$$

Для систем с индивидуальными накопителями и известным количеством K устройств разной производительности V_i $(i = \overline{1, K})$ условие отсутствия перегрузок имеет вид: $\sum_{i=1}^{K} V_i > V_0$, где $V_0 = \lambda\theta$ – нижняя граница суммарной производительности системы, причем для каждой из K подсистем дополнительно должно выполняться условие: $V_i > p_i\lambda\theta$, из которого вытекает требование к доле поступающего в систему с интенсивностью λ трафика

(вероятности p_i), направляемого в i-ю подсистему (к i-му устройству): $p_i < \dfrac{V_i}{\lambda\theta}$ $(i = \overline{1,K})$.

4.4.4. Проблема проектирования систем с несколькими устройствами

Очевидно, что при условии сохранения нагрузки с увеличением числа параллельно работающих устройств времена ожидания и пребывания заявок в системе уменьшаются, поскольку пропорционально количеству устройств уменьшается загрузка каждого устройства. В пределе при $K \to \infty$ время ожидания стремится к нулю, а время пребывания достигает своего наименьшего значения, равного длительности обслуживания заявок. При этом необходимо помнить, что увеличение количества устройств в системе увеличивает её стоимость, но в то же время повышает надежность системы.

При проектировании реальных систем часто возникает следующая проблема. Можно построить систему с одним мощным высокопроизводительным устройством или же с несколькими менее мощными устройствами, производительность которых в сумме равна производительности одного мощного устройства. Какой из этих вариантов является предпочтительным?

Естественно, что с позиции надежности предпочтение следует отдать варианту с несколькими устройствами. При этом возникает вопрос: а какой вариант обеспечивает меньшее время пребывания заявок в системе?

На рисунке 17 показана зависимость времени пребывания и времени ожидания заявок в системе от количества устройств K при условии, что их суммарная производительность (быстродействие) остается постоянной, т.е. $V_\Sigma = KV_K = \text{const}$, где V_K – производительность одного устройства при наличии в системе K устройств.

Из представленных графиков видно, что среднее время ожидания w заявок уменьшается с увеличением числа устройств, однако время пребывания u заявок в системе увеличивается. Последнее объясняется тем, что с увеличением количества устройств производительность каждого из них для сохранения суммарной производительности системы уменьшается пропорционально K и, следовательно, линейно увеличивается длительность обслуживания заявки в устройстве. При этом скорость увеличения длительности обслуживания больше скорости уменьшения времени

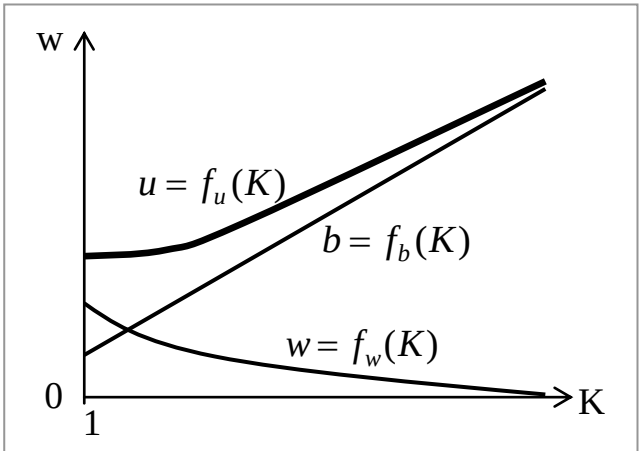

Рисунок 17. Зависимость времени пребывания от количества устройств при сохранении суммарной производительности

ожидания, что в сумме приводит к увеличению времени пребывания заявок в системе. В пределе при $K \to \infty$ время пребывания заявок асимптотически стремится к длительности обслуживания заявок. Таким образом, при проектировании технических систем следует иметь в виду, что с точки зрения задержек (времени пребывания заявок) более эффективной является система с одним быстрым устройством, чем система с несколькими медленными устройствами, суммарная производительность которых равна производительности быстрого устройства в первой системе. Основным достоинством системы с несколькими устройствами является более высокая надёжность, проявляющаяся в том, что при выходе из строя одного или даже нескольких устройств, например процессоров вычислительной системы, система продолжает функционировать, хотя и с меньшей эффективностью, что проявляется в увеличении времени пребывания заявок в системе.

4.5. Системы с несколькими устройствами и накопителем ограниченной ёмкости

В системах с несколькими устройствами и накопителями ограниченной ёмкости, как и в случае неограниченной ёмкости, могут использоваться два способа организации накопителей перед устройствами (рисунок 18):

1) с накопителями ёмкостью $E_1,\ldots,E_K$ перед каждым из K устройств (системы с индивидуальными накопителями), как показано на рисунке 18,а), причем устройства могут иметь разные производительности $V_1,\ldots,V_K$ или быть идентичными;

2) с одним общим для всех устройств накопителем ёмкостью E (системы с общим накопителем), как показано на рисунке 18,б), причем все K устройств идентичны, т.е. имеют одинаковую производительность.

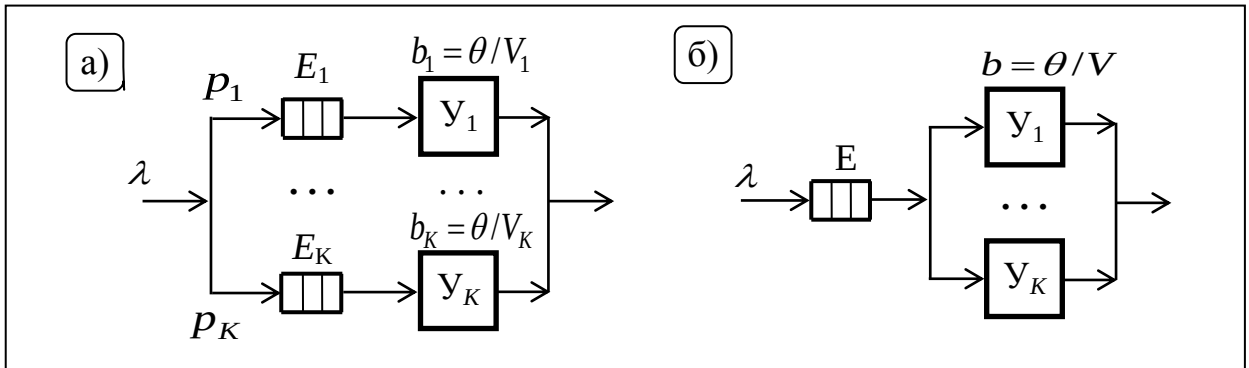

Рисунок 18. Системы с индивидуальными накопителями (а) и с общим накопителем (б) ограниченной ёмкости

4.5.1. Расчёт характеристик систем с несколькими устройствами и индивидуальными накопителями ограниченной ёмкости

Системы с несколькими устройствами и индивидуальными накопителями ограниченной ёмкости можно рассматривать как совокупность K подсистем с одним устройством и накопителем ограниченной ёмкости (рисунок 18,а).

Поступающие с интенсивностью λ в систему заявки с вероятностью p_i $(i = \overline{1, K})$ направляются к устройству (подсистеме) с номером i, причём

$$\sum_{i=1}^{K} p_i = 1,$$

то есть поступившая заявка с вероятностью 1 попадает в один из K накопителей.

Интенсивность поступления заявок к устройству i равна $\lambda_i = p_i \lambda$, а средняя длительность обработки заявок в устройстве $b_i = \frac{\theta}{V_i}$, где θ – ресурсоёмкость обработки заявок, а V_i – производительность i-го устройства $(i = \overline{1, K})$.

Таким образом, расчёт характеристик функционирования системы с несколькими устройствами и индивидуальными накопителями ограниченной ёмкости сводится к расчёту K подсистем с одним устройством и накопителями ограниченной ёмкости с использованием представленных выше зависимостей (25) – (28) или (30) – (33), если заявка, находящаяся на обслуживании в устройстве, занимает место в накопителе.

4.5.2. Расчёт характеристик систем с несколькими устройствами и общим накопителем ограниченной ёмкости

Рассмотрим систему, содержащую K идентичных обслуживающих устройств и общий для всех устройств накопитель ограниченной ёмкости (рисунок 18,б). В систему поступает поток заявок с интенсивностью λ, длительность обработки которых является случайной величиной со средним значением b. Выбор заявок из накопителя на обслуживание осуществляется в соответствии с бесприоритетной дисциплиной обслуживания в порядке поступления по правилу «первым пришёл – первым обслужен» (FIFO). При этом если в момент поступления заявки свободны несколько устройств, поступившая заявка случайным образом направляется к одному из них.

Ведём следующие предположения:

- поток заявок – простейший;
- длительность обработки заявок в каждом из устройств распределена по экспоненциальному закону со средним значением $b = \frac{\theta}{V}$, где θ – средняя ресурсоёмкость обработки одного запроса; V – производительность одного устройства системы;
- все K устройств – идентичны, и любая заявка может быть обслужена любым устройством;
- ёмкость накопителя – ограничена и равна E;
- нагрузка в системе может принимать любое значение: $y = \lambda b > 0$.

Получим аналитические зависимости для расчёта характеристик функционирования системы с использованием метода марковских процессов.

Граф переходов марковского процесса будет иметь конечное число состояний, и система линейных алгебраических уравнений для вероятностей состояний примет вид:

$$\begin{cases} yp_0 = p_1 \\ (y+k)p_k = yp_{k-1} + (k+1)p_{k+1}, & (k = \overline{1, K-1}) \\ (y+K)p_k = yp_{k-1} + Kp_{k+1}, & (k = \overline{K, K+E-1}) \\ Kp_{K+E} = yp_{K+E-1}, \end{cases}$$

где p_k – вероятность состояния k (нахождения в системе ровно k заявок); $y = \lambda/\mu = \lambda b$ – нагрузка, создаваемая в системе; $\mu = 1/b$ – интенсивность обслуживания заявок одним устройством.

Отсюда легко получить, что вероятность состояния k определяется как

$$p_k = \begin{cases} \dfrac{y^k}{k!} p_0 & (k = \overline{1, K-1}); \\ \dfrac{K^K}{K!} \left(\dfrac{y}{K}\right)^k p_0 & (k = \overline{K, K+E}). \end{cases}$$

Используя нормировочное условие: $\sum_{k=0}^{K+E} p_k = 1$, найдём вероятность того, что в системе нет ни одной заявки, т.е. система простаивает:

$$p_0 = \left[\sum_{k=0}^{K-1} \frac{y^k}{k!} + \frac{K^K}{K!} \sum_{k=K}^{K+E} \left(\frac{y}{K}\right)^k \right]^{-1}.$$

После некоторых преобразований окончательно получим:

$$p_0 = \begin{cases} \left[\sum_{k=0}^{K-1} \dfrac{y^k}{k!} + \dfrac{y^K (K^{E+1} - y^{E+1})}{K! K^E (K-y)} \right]^{-1}, & y \neq K \\ \left[\sum_{k=0}^{K-1} \dfrac{y^k}{k!} + \dfrac{(E+1) K^K}{K!} \right]^{-1}, & y = K \end{cases}. \qquad (45)$$

Вероятность P того, что все K устройств заняты обслуживанием заявок, определяется как:

$$P = \sum_{k=K}^{K+E} p_k = \sum_{k=K}^{K+E} \frac{K^K}{K!} \left(\frac{y}{K}\right)^k p_0 = \begin{cases} \dfrac{y^K (K^{E+1} - y^{E+1})}{K! K^E (K-y)} p_0, & y \neq K \\ \dfrac{(E+1) K^K}{K!} p_0, & y = K \end{cases}. \qquad (46)$$

Вероятность потери заявки из-за ограниченной ёмкости накопителя определяется следующим образом:

$$\pi_{\text{п}} = p_{K+E} = \frac{y^{K+E}}{K! K^E} p_0. \qquad (47)$$

Соответственно, загрузка системы рассчитывается как

$$\rho = \frac{(1-\pi_{\text{п}})\lambda b}{K} = \left(1 - \frac{y^{K+E}}{K! K^E} p_0\right) \frac{y}{K}, \qquad (48)$$

где $\lambda_{\text{о}} = (1-\pi_{\text{п}})\lambda$ – интенсивность потока обслуженных заявок.

Среднее число заявок в системе и средняя длина очереди определяются через вероятности состояний:

$$m = \sum_{k=1}^{K+E} k\,p_k = \left[\sum_{k=1}^{K-1}\frac{y^k}{(k-1)!} + \frac{K^K}{K!}\sum_{k=K}^{K+E} k\left(\frac{y}{K}\right)^k\right] p_0; \qquad (49)$$

$$l = \sum_{k=1}^{E} k\,p_{K+k} = p_0\,\frac{K^K}{K!}\sum_{k=1}^{E} k\left(\frac{y}{K}\right)^{K+k} = p_0\,\frac{y^K}{K!}\sum_{k=1}^{E} k\left(\frac{y}{K}\right)^k.$$

С учётом того, что

$$\sum_{k=1}^{N} k x^k = \frac{x}{1-x}\left[\frac{1-x^N}{1-x} - Nx^N\right],$$

последнее выражение при $y \neq K$ после некоторых преобразований примет вид:

$$l = \frac{y^{K+1}}{K!(K-y)K^E}\left[\frac{K(K^E - y^E)}{K-y} - Ey^E\right] p_0,$$

а при $y = K$ получим:

$$l = \frac{E(E+1)K^K}{2K!}\,p_0.$$

Окончательно выражение для расчёта средней длины очереди заявок примет вид:

$$l = \begin{cases} \dfrac{y^{K+1}}{K!(K-y)K^E}\left[\dfrac{K(K^E - y^E)}{K-y} - Ey^E\right] p_0, & y \neq K; \\ \dfrac{E(E+1)K^K}{2K!}\,p_0, & y = K. \end{cases} \qquad (50)$$

Среднее время пребывания заявок в системе и среднее время ожидания рассчитываются с использованием формул Литтла (3) с учётом потерь заявок:

$$u = \frac{m}{(1-\pi_{\text{п}})\lambda}, \qquad w = \frac{l}{(1-\pi_{\text{п}})\lambda}. \qquad (51)$$

4.5.3. Проектирование систем с несколькими устройствами и накопителями ограниченной ёмкости

Для систем с несколькими устройствами и накопителями ограниченной ёмкости, как и выше, основной характеристикой, определяющей эффективность её функционирования, является вероятность потери заявок $\pi_{\text{п}}$, зависящая от трёх параметров системы: ёмкости накопителей E, количества устройств K в системе и производительности устройств V: $\pi_{\text{п}} = f(E, K, V)$, причём, чем больше ёмкость E, количество устройств K и производительность V, тем меньше вероятность потери заявок $\pi_{\text{п}}$.

Для системы с K устройствами и индивидуальными накопителями перед каждым устройством потери заявок из-за переполнения накопителей

возникают в каждой подсистеме «устройство-накопитель». В общем случае каждая подсистема i характеризуется своей:

- интенсивностью поступления заявок $\lambda_i = p_i\lambda$, где λ – интенсивность поступления заявок в систему; p_i – вероятность того, что поступившая заявка будет направлена в подсистему i;
- производительностью (быстродействием) V_i ($i = \overline{1,K}$).

Следовательно, во всех подсистемах вероятности потерь могут быть разными: $\pi_1,...,\pi_K$. Тогда вероятность потери $\pi_{\text{п}}$ заявок в системе с K устройствами и индивидуальными накопителями перед каждым устройством:

$$\pi_{\text{п}} = \sum_{i=1}^{K} p_i\pi_i .$$

Действительно, вероятность потери в подсистеме i определяется как $\pi_i = \dfrac{\lambda_{\text{п}_i}}{p_i\lambda}$, где $\lambda_{\text{п}_i} = \pi_i p_i\lambda$ – интенсивность потерянных заявок в подсистеме i. Суммарная интенсивность потерянных заявок в системе будет равна $\lambda_{\text{п}} = \sum_{i=1}^{K}\lambda_{\text{п}_i} = \sum_{i=1}^{K}\pi_i p_i\lambda$, а вероятность потери $\pi_{\text{п}}$ заявок в системе с K устройствами и индивидуальными накопителями перед каждым устройством определяется как $\pi_{\text{п}} = \dfrac{\lambda_{\text{п}}}{\lambda} = \sum_{i=1}^{K} p_i\pi_i$.

Очевидно, что уменьшение вероятности потери заявок в системе за счёт увеличения количества устройств K, их производительности V_i и ёмкости E_i накопителей увеличивает стоимость системы, которая в данном случае может быть рассчитана как:

$$S = \sum_{i=1}^{K}\xi V_i^{\chi} + s_0 E ,$$

где $S_{\text{у}_i} = \xi V_i^{\chi}$ – стоимость i-го устройства, зависящая от производительности V_i (ξ и χ - стоимостные коэффициенты пропорциональности и нелинейности соответственно); $E = \sum_{i=1}^{K} E_i$ – суммарная ёмкость всех накопителей (E_i – ёмкость накопителя перед i-м устройством); s_0 - стоимость единицы ёмкости.

Задача проектирования системы с несколькими устройствами и индивидуальными накопителями ограниченной ёмкости сводится фактически к задаче проектирования нескольких подсистем с одним устройством и накопителем ограниченной ёмкости, которая может быть решена с использованием подходов, изложенных в пункте 4.3.3. При этом возможны два случая:

- число подсистем K, входящих в состав проектируемой системы, и вероятности поступления заявок в ту или иную подсистему заданы;

• число подсистем K необходимо определить в процессе проектирования.

В первом случае задача проектирования сводится системы к задаче проектирования K разных подсистем с одним устройством и накопителем ограниченной ёмкости.

Во втором случае проектирование системы обычно проводится в предположении, что все подсистемы, входящие в состав системы, идентичны, то есть устройства имеют одну и ту же производительность V, а вероятности попадания поступившей заявки в каждую из подсистем одинаковы: $p_i = p = 1/K \quad (i = \overline{1,K})$. Таким образом, первоначальная задача проектирования состоит в определении количества подсистем K, после чего рассчитываются значения ёмкости каждого накопителя и производительности устройства. Следует отметить, что вероятность потери заявок в системе совпадает с вероятностью потери заявок в подсистеме, поскольку вероятности попадания поступившей в систему заявки и вероятности потери заявок во всех подсистемах одинаковы: $\pi_i = \pi \quad (i = \overline{1,K})$ и, следовательно,

$$\pi_{\text{п}} = \sum_{i=1}^{K} p_i \pi_i = \sum_{i=1}^{K} \frac{1}{K} \pi = \pi .$$

Задача проектирования системы с несколькими устройствами и общим накопителем ограниченной ёмкости может быть сформулирована в нескольких постановках:

• при заданной производительности V одного устройства определить количество устройств K и ёмкость E накопителя, обеспечивающие выполнение ограничения на вероятность потери заявок $\pi_{\text{п}} < \pi^*$ при минимальной стоимости системы $S - \min$;

• при заданной производительности V одного устройства определить количество устройств K и ёмкость E накопителя, обеспечивающие выполнение ограничения на стоимость системы $S < S^*$ при минимальной вероятности потери заявок $\pi_{\text{п}} - \min$;

• определить количество устройств K, их производительность V и ёмкость E накопителя, обеспечивающие выполнение ограничения на вероятность потери заявок $\pi_{\text{п}} < \pi^*$ при минимальной стоимости системы $S - \min$;

• определить количество устройств K, их производительность V и ёмкость E накопителя, обеспечивающие выполнение ограничения на стоимость системы $S < S^*$ при минимальной вероятности потери заявок $\pi_{\text{п}} - \min$.

Для решения задачи проектирования в представленных постановках могут быть использованы аналитические зависимости (45) – (51). Поскольку получить решение задачи оптимального синтеза в явном аналитическом виде не представляется возможным, решение для первой постановки задачи проектирования может быть выполнено, например, следующим образом.

На первом этапе проектируется система с одним устройством и накопителем ограниченной ёмкости, как это изложено в пункте 4.3.3, с использованием аналитических зависимостей (25) – (29). При этом определяются ёмкость накопителя E и суммарная производительность устройств V, при которых выполняются заданные ограничения.

На втором этапе перебираются варианты построения системы с несколькими устройствами $K = 2, 3, \ldots$. Для каждого варианта рассматриваются две модели.

В первой модели производительность одного устройства принимается равной $V_1 = V/K$, где V – суммарная производительность устройств, рассчитанная на первом этапе. Затем с использованием зависимостей (45) – (51) рассчитываются характеристики функционирования, и подбирается ёмкость накопителя E, при которой выполняется заданное ограничение на вероятность потери заявок $\pi_{\text{п}} < \pi^*$.

Во второй модели ёмкость накопителя принимается равной ёмкости E, полученной на первом этапе, и с использованием зависимостей (45) – (54) подбирается производительность одного устройства V, при которой выполняется заданное ограничение на вероятность потери заявок $\pi_{\text{п}} < \pi^*$.

Из рассмотренных двух моделей в качестве наилучшей выбирается модель меньшей стоимости.

На третьем этапе сравниваются стоимости всех рассмотренных вариантов построения системы с несколькими устройствами $K = 2, 3, \ldots$ и в качестве окончательного решения выбирается вариант с минимальной стоимостью.

При наличии ограничения на среднее значение времени пребывания заявок в системе в виде $u < u^*$ может возникнуть необходимость увеличить производительность или количество устройств, что приведёт к увеличению стоимости системы. При этом вероятность потери заявок уменьшится и, следовательно, можно попытаться уменьшить стоимость системы за счёт уменьшения ёмкости накопителя.

Проектирование систем на основе базовых моделей с неоднородной нагрузкой

5.1. Системы с одним устройством и неоднородной нагрузкой

Во многих реальных системах поступающие в систему запросы относятся к разным классам, образуя неоднородный поток запросов. При этом к качеству обслуживания запросов разных классов могут предъявляться разные требования.

В качестве моделей таких систем могут использоваться как модели с неоднородным потоком заявок, так и с однородным потоком. Последнее предполагает сведение неоднородного потока заявок к однородному, создающему в системе такую же нагрузку, как и неоднородный поток. Сведение к однородному потоку заявок возможно при выполнении следующих условий:

- между запросами разных классов отсутствуют приоритеты, причем заявки всех классов обслуживаются в порядке поступления, т.е. в соответствии с бесприоритетной дисциплиной FIFO;
- требования, предъявляемые к качеству обслуживания заявок, отсутствуют или одинаковы для всех классов заявок.

Сведение неоднородного потока заявок к однородному состоит в пересчете параметров нагрузки. Положим, что в систему поступают заявки H классов с интенсивностями $\lambda_1,\ldots,\lambda_H$, длительности обслуживания которых заданы средними значениями $b_1,\ldots,b_H$ и вторыми начальными моментами $b_1^{(2)},\ldots,b_H^2$. Параметры однородного потока, создающего эквивалентную нагрузку могут быть рассчитаны по следующим формулам:

- интенсивность суммарного потока заявок: $\Lambda=\sum_{h=1}^{H}\lambda_h$;
- средняя длительность обслуживания заявки: $B=\frac{1}{\Lambda}\sum_{h=1}^{H}\lambda_h b_h$;
- второй начальный момент длительности обслуживания заявки: $B^{(2)}=\frac{1}{\Lambda}\sum_{h=1}^{H}\lambda_h b_h^{(2)}$.

Таким образом, в качестве модели системы с неоднородным потоком запросов может использоваться модель с однородным потоком с параметрами нагрузки Λ, B и $B^{(2)}$.

В тех случаях, когда в процессе построения модели реальной системы не удается неоднородный поток заявок свести к однородному, в качестве моделей систем используются СМО с неоднородным потоком заявок (рисунок 19). Как сказано выше, невозможность сведения к однородному потоку заявок может быть обусловлена разными причинами, в частности, наличием приоритетов между заявками, задаваемых в виде дисциплины обслуживания (ДО), необходимостью дифференцирования характеристик заявок разных классов, особенно в случае существенного различия длительностей их обработки в системе.

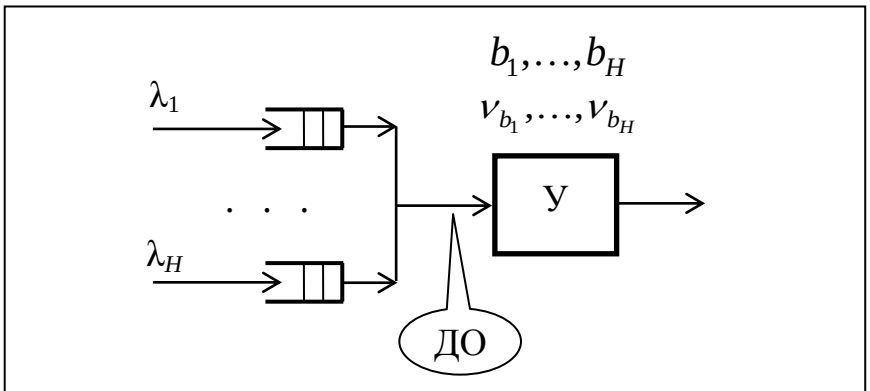

Рисунок 19. Система с неоднородной нагрузкой

Для получения аналитических зависимостей для расчёта характеристик функционирования системы с неоднородной нагрузкой при применении различных дисциплин обслуживания заявок воспользуемся методом средних значений.

Положим, что в систему поступают H классов заявок, образующие простейшие потоки с интенсивностями $\lambda_1,\ldots,\lambda_H$. Длительности обработки заявок разных классов распределены по произвольным законам со средними значениями $b_1,\ldots,b_H$ и коэффициентами вариации $\nu_{b_1},\ldots,\nu_{b_H}$. Накопители для каждого класса заявок имеют неограниченную ёмкость, то есть система без потерь. Заявки выбираются из накопителей на обслуживание в соответствии с заданной дисциплиной обслуживания (ДО). В качестве таких ДО ниже рассматриваются следующие дисциплины:

- бесприоритетная (БП);
- с относительными приоритетами (ОП);
- с абсолютными приоритетами (АП);
- со смешанными приоритетами (СП).

5.1.1. Расчёт характеристик систем с бесприоритетным обслуживанием заявок

Положим, что заявки разных классов выбираются из очереди на обслуживание в соответствии с бесприоритетной дисциплиной в порядке поступления (FIFO).

Среднее время ожидания w_k заявки класса k, поступившей в систему в некоторый момент времени, будет складываться из среднего времени дообслуживания T_0 заявки, находящейся на обслуживании в устройстве, среднего времени обслуживания T_1 всех ранее поступивших заявок всех классов:

$$w_k = T_0 + T_1 \quad (k = \overline{1,H}). \tag{52}$$

Выражение для T_0 с учетом (4) и (5) может быть записано как

$$T_0 = \frac{1}{2}\sum_{i=1}^{H} \lambda_i b_i^2 (1+\nu_{b_i}^2). \tag{53}$$

Выражение для T_1 с учетом того, что в очереди находится в среднем $l_i = \lambda_i w_i$ заявок класса i, каждая из которых в среднем будет обслуживаться в устройстве в течение времени b_i, определяется как

$$T_1 = \sum_{i=1}^{H} l_i\, b_i = \sum_{i=1}^{H} \lambda_i\, w_i\, b_i = \sum_{i=1}^{H} \rho_i\, w_i\,. \tag{54}$$

Подставив (53) и (54) в (52), получим:

$$w_k = \frac{1}{2}\sum_{i=1}^{H} \lambda_i b_i^2 (1+\nu_{b_i}^2) + \sum_{i=1}^{H} \rho_i\, w_i\,.$$

Поскольку правая часть полученного выражения не зависит от номера класса заявок (индекса) k, можно сделать вывод о том, что времена ожидания заявок всех классов одинаковы: $w_k = w = \text{const}$. Тогда

$$w_k^{БП} = w^{БП} = \frac{\sum_{i=1}^{H} \lambda_i b_i^{\,2} (1 + \nu_{bi}^{\,2})}{2(1-R)} \qquad (k = \overline{1, H}), \tag{55}$$

где $R = \sum_{i=1}^{H} \lambda_i b_i = \sum_{i=1}^{H} \rho_i$ – суммарная загрузка системы, причем $R < 1$ (условие отсутствия перегрузок).

5.1.2. Расчёт характеристик систем с относительными приоритетами

Приоритеты называются относительными, если они учитываются только в момент выбора заявки на обслуживание и не сказываются на работе системы в период обслуживания заявки любого класса (приоритета). Относительность приоритета связана со следующим. После завершения обслуживания какой-либо заявки из очереди на обслуживание выбирается заявка класса с наиболее высоким приоритетом, поступившая ранее других заявок этого класса (такого же приоритета). Если в процессе её обслуживания в систему поступят заявки с более высоким приоритетом, то обслуживание рассматриваемой заявки не будет прекращено, то есть эта заявка, захватив устройство, оказывается как бы более приоритетной. Таким образом, приоритет относителен в том смысле, что он имеет место лишь в момент выбора заявок на обслуживание и отсутствует, если устройство занято обслуживанием какой-либо заявки.

Введение относительных приоритетов (ОП) позволяет уменьшить по сравнению с ДО БП время ожидания высокоприоритетных заявок.

Положим, что между заявками разных классов установлены относительные приоритеты, причем заявки класса с меньшим номером имеют более высокий относительный приоритет.

Среднее время ожидания w_k заявки класса k, поступившей в систему в некоторый момент времени, будет складываться из среднего времени дообслуживания T_0 заявки, находящейся на обслуживании в устройстве, среднего времени обслуживания T_k всех ранее поступивших заявок с таким же или с более высоким приоритетом и находящихся в очереди, среднего времени обслуживания T_{k-1} всех заявок с более высоким приоритетом, которые поступят в систему за время ожидания рассматриваемой заявки:

$$w_k = T_0 + T_k + T_{k-1}. \tag{56}$$

Время дообслуживания T_0 определяется в соответствии с выражением (53).

Выражение для T_k с учетом того, что в очереди находится в среднем $l_i = \lambda_i w_i$ заявок класса i, каждая из которых в среднем будет обслуживаться в устройстве в течение времени b_i, определяется как

$$T_k = \sum_{i=1}^{k} l_i\, b_i = \sum_{i=1}^{k} \lambda_i\, w_i\, b_i = \sum_{i=1}^{k} \rho_i\, w_i\,. \qquad (57)$$

Выражение для T_{k-1} с учетом того, что согласно (2) за время ожидания w_k рассматриваемой заявки в систему поступит в среднем $m_i = \lambda_i w_k$ заявок класса i с более высоким приоритетом, чем k, каждая из которых в среднем будет обслуживаться в устройстве в течение времени b_i, будет определяться как

$$T_{k-1} = \sum_{i=1}^{k-1} m_i\, b_i = \sum_{i=1}^{k-1} \lambda_i\, w_k\, b_i = w_k \sum_{i=1}^{k-1} \rho_i = R_{k-1} w_k\,, \qquad (58)$$

где $R_{k-1} = \sum_{i=1}^{k-1} \rho_i$ – загрузка, создаваемая заявками классов $1,\ldots,k-1$, имеющих более высокий приоритет, чем класс k $(k = \overline{1,H})$.

Подставив (53), (57) и (58) в (56), получим:

$$w_k = \frac{1}{2}\sum_{i=1}^{H} \lambda_i b_i^2 (1+\nu_{b_i}^2) + \sum_{i=1}^{k} \rho_i\, w_i + R_{k-1} w_k\,,$$

откуда

$$w_k = \frac{\frac{1}{2}\sum_{i=1}^{H} \lambda_i b_i^2 (1+\nu_{b_i}^2) + \sum_{i=1}^{k} \rho_i\, w_i}{1 - R_{k-1}}.$$

Применяя к последнему выражению метод математической индукции, получим формулу для расчёта среднего времени ожидания для дисциплины обслуживания с относительными приоритетами:

$$w_k^{ОП} = \frac{\sum_{i=1}^{H} \lambda_i {b_i}^2 (1+\nu_{b_i}{}^2)}{2(1-R_{k-1})(1-R_k)} \qquad (k = \overline{1,H}), \qquad (59)$$

где $R_k = \sum_{i=1}^{k} \rho_i$ – загрузка, создаваемая заявками классов $1,\ldots,k$, имеющих более высокий или такой же приоритет, что и класс k $(k = \overline{1,H})$.

5.1.3. Расчёт характеристик систем с абсолютными приоритетами

Иногда время ожидания заявок некоторых классов необходимо уменьшить в такой степени, которая недостижима при использовании ДО ОП. Можно предположить, что время ожидания уменьшится, если при поступлении высокоприоритетной заявки обслуживание ранее поступившей заявки с низким приоритетом прерывается, и устройство переходит к обслуживанию высокоприоритетной заявки. Приоритет, прерывающий обслуживание низкоприоритетной заявки, называется абсолютным, а соответствующая дисциплина – дисциплиной обслуживания с абсолютными приоритетами (ДО АП).

Прерванная заявка может быть потеряна или возвращена в накопитель, где она будет ожидать дальнейшего обслуживания. В последнем случае возможны два варианта продолжения обслуживания прерванной заявки:

- обслуживание с начала, то есть прерванная заявка будет обслуживаться заново с самого начала;
- дообслуживание, когда обслуживание прерванной заявки в устройстве будет выполняться с прерванного места.

В дальнейшем, если не оговорено иное, будем предполагать дообслуживание прерванной заявки.

Положим, что между заявками разных классов установлены абсолютные приоритеты, причем заявки класса с меньшим номером имеют более высокий приоритет. Используя метод средних значений, можно показать, что среднее время ожидания заявок класса k будет определяться по формуле:

$$w_k^{A\Pi} = \frac{\sum_{i=1}^{k} \lambda_i b_i^{\,2}(1+\nu_{bi}^{\,2})}{2(1-R_{k-1})(1-R_k)} + \frac{R_{k-1} b_k}{1-R_{k-1}} \qquad (k=\overline{1,H}), \tag{60}$$

где R_{k-1} и R_k – суммарные загрузки системы, создаваемые заявками с абсолютным приоритетом не ниже $(k-1)$ и k соответственно:

$$R_{k-1} = \sum_{i=1}^{k-1} \rho_i; \quad R_k = \sum_{i=1}^{k} \rho_i .$$

Отметим, что полное время ожидания $w_k^{A\Pi}$ складывается из двух составляющих: времени ожидания начала обслуживания (первое слагаемое в выражении (60) и времени нахождения в прерванном состоянии (второе слагаемое), которое зависит от длительности обслуживания b_k.

5.1.4. Законы сохранения

Изменение дисциплины обслуживания (ДО) позволяет уменьшить время ожидания высокоприоритетных заявок за счет увеличения времени ожидания низкоприоритетных заявок. Очевидно, что за счет изменения ДО нельзя добиться того, чтобы уменьшилось время ожидания заявок всех классов. Этот факт сформулирован в виде законов сохранения времени ожидания и времени пребывания.

Формулировка закона сохранения времени ожидания. Для любой ДО

$$\sum_{i=1}^{H} \rho_i w_i = \underset{\text{ДО}}{Const},$$

т.е. сумма произведений загрузок ρ_i на среднее время ожидания w_i заявок всех классов инвариантна относительно ДО $(i=\overline{1,H})$.

Закон сохранения времени ожидания выполняется при следующих условиях:

- система без потерь, т.е. все заявки на обслуживание удовлетворяются;
- система простаивает лишь в том случае, когда в ней нет заявок;

- при наличии прерываний длительность обслуживания прерванных заявок распределена по экспоненциальному закону;
- все поступающие потоки заявок – простейшие, и длительность обслуживания не зависит от потоков заявок.

Значение константы можно определить следующим образом. Поскольку закон сохранения справедлив для любых ДО, удовлетворяющих перечисленным условиям, то он справедлив и для ДО БП, для которой $w_k^{БП} = w^{БП}$ для всех $(k = \overline{1,H})$. Отсюда находим значение константы:

$$Const = w^{БП} \sum_{i=1}^{H} \rho_i = R w^{БП} .$$

Подставив полученное значение константы и формулу (55) вместо $w^{БП}$ в закон сохранения, окончательно получим:

$$\sum_{i=1}^{H} \rho_i w_i = \frac{R}{2(1-R)} \sum_{i=1}^{H} \lambda_i b_i^{\,2} (1 + \nu_{bi}^{\,2}) . \qquad (61)$$

Закон сохранения времени ожидания универсален и справедлив для всех ДО, удовлетворяющих указанным условиям. Его можно использовать для оценки достоверности приближенных результатов, полученных при исследовании сложных ДО и проведении имитационного моделирования, а также при решении задач синтеза.

Модификация закона сохранения. Закон сохранения может быть модифицирован применительно ко времени пребывания заявок в системе с учетом того, что $w_i = u_i - b_i$. Подставив это выражение в закон сохранения времени ожидания, после некоторых преобразований получим закон сохранения времени пребывания:

$$\sum_{i=1}^{H} \rho_i u_i = \frac{R}{2(1-R)} \sum_{i=1}^{H} \lambda_i b_i^{\,2} (1 + \nu_{bi}^{\,2}) + \sum_{i=1}^{H} \rho_i b_i . \qquad (62)$$

Заметим, что изменение ДО приводит только к изменению времени ожидания и, соответственно, времени пребывания, а остальные величины, входящие в выражения (61) и (62), не изменяются.

5.1.5. Критерии эффективности для решения задачи распределения приоритетов

При решении задачи распределения приоритетов необходимо определить, какая из трех рассмотренных выше дисциплин обслуживания (БП, ОП или АП) более эффективна, и, если выбрана приоритетная дисциплина (ОП или АП), распределить приоритеты между классами заявок.

При отсутствии ограничений на *времена пребывания заявок* в системе обычно считается, что чем дольше заявки пребывают в системе, тем ниже качество функционирования последней, т.е. тем в меньшей мере система соответствует своему назначению. Потеря качества функционирования из-за задержки обслуживания заявок характеризуется функцией штрафа

$$F_u = \sum_{i=1}^{H} k_i \lambda_i u_i \,, \qquad (63)$$

где k_i – штраф за задержку в системе на единицу времени одной заявки класса i; λ_i и u_i – интенсивность потока и среднее время пребывания заявок в системе соответственно $(i = \overline{1, H})$.

Произведение $\alpha_i \lambda_i u_i$ определяет штраф за задержку заявок класса i, поступающих в систему за единицу времени, а значение F_u – штраф за задержку заявок всех классов. Критерий (63) – инверсный: более эффективной системе соответствует меньшее значение F_u.

Очевидно, что критерий эффективности в виде функции штрафа за ожидание в очереди: $F_w = \sum_{i=1}^{H} k_i \lambda_i w_i$ где w_i – среднее время ожидания заявок, эквивалентен критерию (63).

С использованием критерия эффективности F_w сравним две дисциплины обслуживания: ДО БП и ДО ОП и определим, при каком распределении приоритетов ДО ОП будет лучше ДО БП, т.е.: $F_w^{\text{ОП}} < F_w^{\text{БП}}$ или в развернутой записи:

$$\sum_{i=1}^{H} k_i \lambda_i w_i^{\text{ОП}} < \sum_{i=1}^{H} k_i \lambda_i w_i^{\text{БП}} \,.$$

Подставляя (55) и (59) в последнее выражение вместо $w_i^{\text{БП}}$ и $w_i^{\text{ОП}}$, после некоторых преобразований можно получить условие, при выполнении которого ДО ОП будет лучше ДО БП:

$$\frac{b_i}{k_i} < \frac{b_{i+1}}{k_{i+1}} \quad (i = \overline{1, H-1}) \,.$$

Если длительности обслуживания заявок распределены по экспоненциальному закону, это же условие справедливо для распределения абсолютных приоритетов, при котором ДО АП будет лучше, чем ДО ОП в соответствии с критерием эффективности (63).

Поскольку формула (59) получена в предположении о том, что относительные приоритеты назначены по правилу «более высокий приоритет – классу заявок с меньшим номером», то из полученного неравенства следует, что приоритеты должны назначаться по правилу: чем меньше отношение длительности обслуживания заявок b_i к величине штрафа k_i, тем более высокий приоритет должен быть назначен данному классу. Другими словами, приоритет должен убывать с возрастанием отношения $\frac{b_i}{k_i} \quad (i = \overline{1, H})$.

В частном случае при $k_i = k = \underset{i=\overline{1,H}}{\text{const}}$ правило назначения приоритетов примет вид:

$$b_i < b_{i+1} \quad (i = \overline{1, H-1}), \qquad (64)$$

т.е. более высокий приоритет должен назначаться заявкам с меньшей длительностью обслуживания. Это правило широко применяется в информационно-управляющих системах реального времени и формулируется как «короткая задача – вперёд». В тех случаях, когда длительность обработки запроса в вычислительной системе заранее неизвестна, принцип «короткая задача – вперёд» реализуется в виде режима разделения времени, когда каждому запросу предоставляется небольшой квант процессорного времени, и если обработка запроса не завершилась за этот квант, то его обработка прерывается и запрос возвращается в очередь ожидающих запросов, а процессор переходит к обработке следующего запроса. Таким образом обеспечивается преимущество (приоритет) коротким запросам, требующим небольшого времени обработки.

Если штрафы k_i одинаковы для всех классов заявок, то критерий (63) принимает вид:

$$F_u = \sum_{i=1}^{H} \lambda_i u_i = \sum_{i=1}^{H} m_i = M, \qquad (65)$$

где $m_i = \lambda_i u_i$ – среднее число заявок класса i, находящихся в системе (в накопителях и на обслуживании).

Таким образом, критерий эффективности (63) вырождается в суммарное число заявок всех классов в системе, а задача распределения приоритетов сводится к минимизации среднего числа заявок, находящихся в системе: $M - \min$.

Представим выражение (65) в следующем виде:

$$F_u = \frac{1}{\Lambda} \sum_{i=1}^{H} \lambda_i u_i = u_\Sigma,$$

где u_Σ - среднее время пребывания в системе заявок суммарного потока, т.е. среднее время пребывания в системе заявки любого класса.

Отсюда следует, что оптимальное распределение приоритетов в соответствии с критерием (65) позволяет минимизировать среднее время пребывания в системе заявок суммарного потока: $u_\Sigma - \min$.

Задержки, т.е. времена пребывания $u_1, ..., u_H$, зависят от двух факторов: производительности устройства и дисциплины обслуживания заявок. Увеличение производительности приводит к уменьшению всех значений $u_1, ..., u_H$. За счет использования приоритетных дисциплин обслуживания можно уменьшить времена ожидания заявок одних классов, но при этом времена ожидания заявок других классов увеличатся.

Рассмотрим вопрос о возможности использования функции (63) в качестве критерия эффективности систем с неограниченным временем пребывания заявок. Допустим, что функция F_u используется в качестве критерия при решении задачи выбора производительности устройства. С увеличением производительности устройства уменьшаются значения $u_1, ..., u_H$ и, следовательно, уменьшается значение F_u. Максимальная эффективность

системы достигается при бесконечной производительности устройства, что свидетельствует о некорректности поставленной задачи. Таким образом, использование критерия (63) предполагает, что производительность устройства задана.

Критерий (63) может быть использован в задаче выбора дисциплины обслуживания заявок. Действительно, различным дисциплинам соответствуют различные значения времени ожидания $u_1,...,u_H$, которые при изменении дисциплины взаимно перераспределяются в соответствии с законом сохранения времени ожидания. Дисциплина обслуживания заявок может считаться оптимальной для определенной системы, если ей соответствует минимальное значение F_u по сравнению с другими дисциплинами обслуживания.

При использовании критерия (63) выбор оптимальной дисциплины обслуживания может проводиться в принципе для любого значения производительности устройства, обеспечивающего отсутствие перегрузок в системе, поскольку это не сказывается на выборе оптимальной дисциплины. Минимальное значение производительности устройства, при котором может быть найдена дисциплина обслуживания, удовлетворяющая некоторым заданным требованиям, представляет собой *нижнюю границу производительности* устройства.

После того как найдена оптимальная дисциплина обслуживания, возникает задача уточнения производительности устройства, заключающаяся в определении оптимального значения в смысле некоторого критерия. Для построения такого критерия будем рассуждать следующим образом. С целью уменьшения времени ожидания заявок в системе необходимо иметь устройство с высокой производительностью. Но с увеличением производительности устройства растет коэффициент его простоя и в пределе приближается к единице:

$$\eta = 1 - R = 1 - \sum_{i=1}^{H} \lambda_i b_i = 1 - \frac{1}{V} \sum_{i=1}^{H} \lambda_i \theta_i , \qquad (66)$$

где λ_i – интенсивность потока заявок класса i; $R = \sum_{i=1}^{H} \lambda_i b_i$ – суммарная загрузка системы; $b_i = \frac{\lambda_i \theta_i}{V}$ – средняя длительность обработки заявок класса $i = \overline{1,H}$.

Очевидно, что для уменьшения простоев системы необходимо выбрать устройство по возможности с меньшей производительностью. Можно предположить о существовании некоторого оптимального решения, позволяющего определить производительность устройства с учетом двух указанных противоречивых факторов. В качестве критерия эффективности при таком подходе может быть использован функционал вида:

$$F_V = k_0 \eta(V) + \sum_{i=1}^{H} k_i \lambda_i u_i(V), \qquad (67)$$

где u_i – время пребывания в системе заявок класса i; k_i – весовые коэффициенты, задаваемые при проектировании системы $(i = 0, 1, \ldots, H)$, например на основе экспертных оценок.

Задание весовых коэффициентов k_i должно осуществляться исходя из назначения системы и требований, предъявляемых к ней. При этом системы, предназначенные для управления объектами или процессами с жесткими требованиями ко времени реакции, должны иметь более высокие значения коэффициентов k_i $(i = \overline{1, H})$, а системы, основное требование к которым – минимум материальных затрат, должны иметь наибольшее значение k_0.

Таким образом, в системах с неограниченным временем пребывания заявок в качестве критерия эффективности может быть использована функция (63), когда выбирается оптимальная дисциплина обслуживания заявок, и функция (67), когда определяется оптимальное значение производительности устройства.

5.2. Системы со смешанными приоритетами

Дисциплины обслуживания (ДО) заявок с одним классом приоритетов (ОП или АП) не всегда позволяют достичь требуемого качества функционирования системы. Поэтому в реальных системах часто используются ДО общего вида, в которых один и тот же класс заявок может иметь АП по отношению к одной группе классов заявок, ОП – к другой группе и не иметь приоритета к остальным классам заявок. Такие ДО называются дисциплинами обслуживания со смешанными приоритетами (ДО СП).

5.2.1. Способы описания дисциплин со смешанными приоритетами

Для представления ДО СП могут использоваться два способа:

1) математическое описание в виде матрицы приоритетов, элементы которой используются в формулах для расчёта характеристик обслуживания заявок;

2) графическое представление в виде схемы ДО, обеспечивающей по сравнению с МП б*о*льшую наглядность.

Матрица приоритетов (МП) представляет собой квадратную матрицу $Q = [q_{ij}\ (i, j = 1, \ldots, H)]$, размерность которой определяется числом классов заявок H, поступающих в систему. Элемент q_{ij} матрицы задает приоритет заявок класса i по отношению к заявкам класса j и может принимать следующие значения: 0 – нет приоритета, 1 – приоритет относительный и 2 – приоритет абсолютный.

С помощью МП можно описать большое множество ДО, в том числе дисциплины с одним классом приоритетов. Например, в случае четырех классов заявок $(H = 4)$, матрицы приоритетов, соответствующие ДО БП, ДО ОП и ДО АП будут иметь вид:

$$Q^{\text{БП}} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \quad Q^{\text{ОП}} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \quad Q^{\text{АП}} = \begin{bmatrix} 0 & 2 & 2 & 2 \\ 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Примерами ДО СП могут служить дисциплины, описываемые следующими матрицами приоритетов:

$$Q^{\text{СП}_1} = \begin{bmatrix} 0 & 1 & 2 & 2 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \quad Q^{\text{СП}_2} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 2 & 2 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

Элементы МП должны удовлетворять следующим требованиям:

1) $q_{ii} = 0$, так как между заявками одного и того же класса не могут быть установлены приоритеты;

2) если $q_{ij} = 1$ или 2, то $q_{ji} = 0$, т.е. если заявки класса i имеют приоритет по отношению к заявкам класса j, то последние не могут иметь приоритет по отношению к заявкам класса i $(i, j = 1, \ldots, H)$.

Схема ДО предназначена для графического представления ДО СП и обладает большей наглядностью по сравнению с МП. При построении схемы ДО используются обозначения, представленные на рисунке 20.

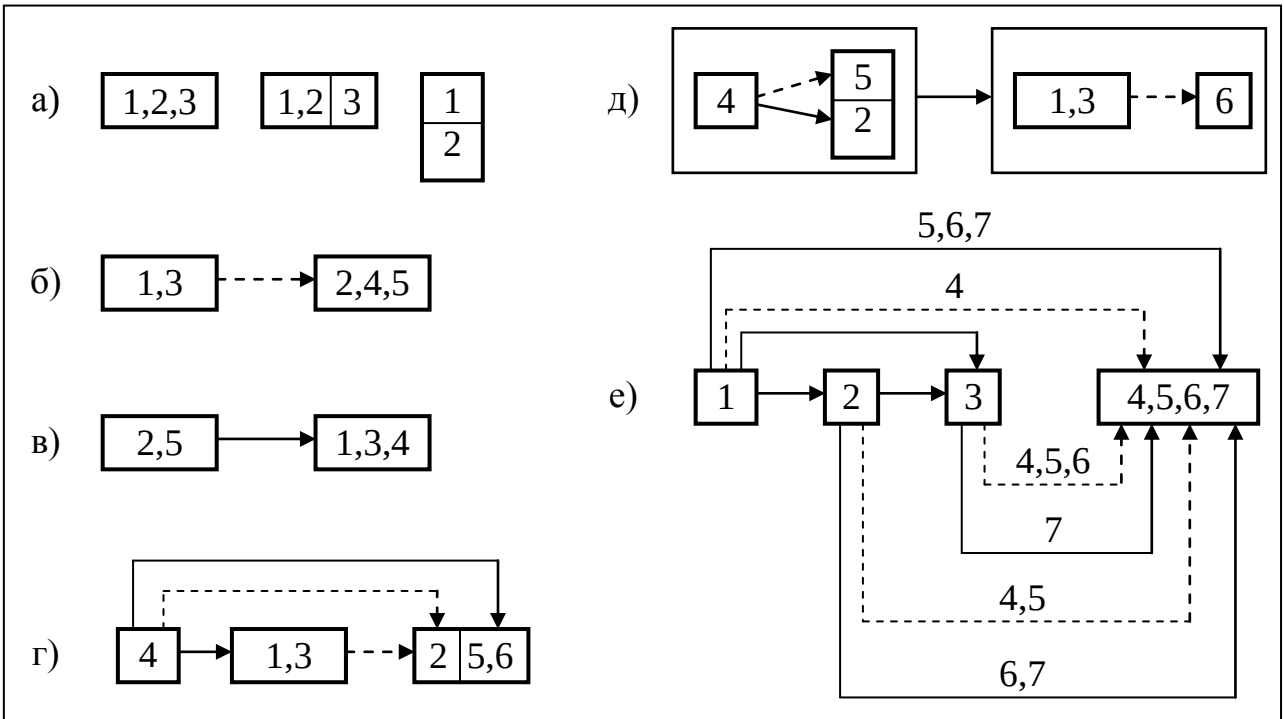

Рисунок 20. Схемы ДО

Множество классов заявок, имеющих одинаковый приоритет и обслуживаемых в соответствии с бесприоритетной ДО в порядке поступления, образуют бесприоритетную группу (БП-группу) и изображаются на схеме в виде прямоугольника, внутри которого указаны номера классов заявок,

входящих в БП-группу (рисунок 20,а). В частном случае БП-группа может содержать только один класс заявок. В случае нескольких классов заявок, входящих в БП-группу, допускаются различные способы изображения БП-группы, как это показано на рисунке 20,а. Два последних способа применяются в тех случаях, когда высокоприоритетные БП-группы имеют разные виды приоритетов (относительные и абсолютные) по отношению к разным классам заявок, образующих между собой БП-группу. При этом классы заявок, по отношению к которым высокоприоритетные БП-группы имеют разные приоритеты, могут отделяться друг от друга чертой (рисунок 20,г и рисунок 20,д) или их номера могут явно указываться на линиях, отображающих вид приоритета между БП-группами (рисунок 20,е).

Относительный приоритет между двумя БП-группами отображается на схеме ДО в виде пунктирной стрелки, направленной от высокоприоритетной БП-группы к низкоприоритетной (рисунок 20,б). Аналогично, абсолютный приоритет между двумя БП-группами отображается на схеме ДО в виде сплошной стрелки (рисунок 20,в).

На основе введенных обозначений могут быть построены более сложные схемы ДО, примеры которых показаны на рисунках 20,г), 20,д) и 20,е).

Между матрицей приоритетов и схемой ДО существует взаимно-однозначное соответствие. Схема ДО, обладая наглядностью, позволяет достаточно просто построить матрицу приоритетов, при этом для исключения ошибок рекомендуется придерживаться определенной последовательности заполнения МП, которую рассмотрим на примере заполнения МП для ДО СП, заданной в виде схемы ДО, изображенной на рисунке 20,д).

Перед заполнением МП имеет вид Q_1, показанный на рисунке 21.

На последующих шагах (матрицы $Q_2,\ldots,Q_6$) последовательно заполняются строки и столбцы МП для классов заявок, имеющих наивысший приоритет среди оставшихся классов. Заметим, что классы заявок, расположенные на схеме ДО левее, имеют приоритет не ниже, чем классы заявок, расположенные правее.

Для рассматриваемой ДО СП (рисунок 20,д) наивысшим приоритетом обладают заявки класса 4, и заполнение МП начинается с этого класса. Сначала заполняется строка с номером 4, в которую заносятся следующие значения:

- 1 – на пересечении со столбцом 5, поскольку, как следует из схемы ДО, заявки класса 4 имеют ОП по отношению к заявкам класса 5;
- 2 – на пересечении со столбцами 2, 1, 3 и 6, поскольку заявки класса 4 имеют АП по отношению к заявкам классов 2, 1, 3 и 6.

После того как строка заполнена, свободные позиции столбца с тем же номером 4 заполняются нулями. В результате МП примет вид Q_2 (см. рисунок 21). Из дальнейшего рассмотрения класс заявок с номером 4 исключается. Следующим классом, который имеет наивысший приоритет среди оставшихся классов, является класс 5, для которого аналогично заполняются строка и столбец МП. МП примет вид Q_3, и этот класс также исключается из

дальнейшего рассмотрения. Аналогичным образом заполняются строки и столбцы МП для оставшихся классов 2 (Q_4), 1 (Q_5) и 3 (Q_6).

Окончательно получена МП Q_6, однозначно соответствующая исходной схеме ДО (рисунок 20,д).

Q_1	1	2	3	4	5	6
1	0					
2		0				
3			0			
4				0		
5					0	
6						0

Q_2	1	2	3	4	5	6
1	0			0		
2		0		0		
3			0	0		
4	2	2	2	0	1	2
5				0	0	
6				0		0

Q_3	1	2	3	4	5	6
1	0			0	0	
2		0		0	0	
3			0	0	0	
4	2	2	2	0	1	2
5	2	0	2	0	0	2
6				0	0	0

Q_4	1	2	3	4	5	6
1	0	0		0	0	
2	2	0	2	0	0	2
3		0	0	0	0	
4	2	2	2	0	1	2
5	2	0	2	0	0	2
6		0		0	0	0

Q_5	1	2	3	4	5	6
1	0	0	0	0	0	1
2	2	0	2	0	0	2
3	0	0	0	0	0	
4	2	2	2	0	1	2
5	2	0	2	0	0	2
6	0	0		0	0	0

Q_6	1	2	3	4	5	6
1	0	0	0	0	0	1
2	2	0	2	0	0	2
3	0	0	0	0	0	1
4	2	2	2	0	1	2
5	2	0	2	0	0	2
6	0	0	0	0	0	0

Рисунок 21. Последовательность заполнения матрицы приоритетов

Отметим, что предлагаемая последовательность заполнения МП не является обязательной, а лишь упрощает процесс формирования МП, гарантируя однозначность и позволяя избегать путаницы и ошибок.

Матрицу приоритетов будем называть *канонической*, если $q_{ij} = 0$ для всех $i > j$ $(i, j = 1, \ldots, H)$. Канонические МП описывают дисциплины, в которых заявки классов с меньшим номером имеют приоритет не ниже, чем заявки классов с большим номером. Очевидно, что любая МП может быть приведена к каноническому виду путем соответствующей перенумерации классов.

Число вариантов заполнения канонической МП $\xi = 3^{H(H-1)/2}$, где H – число классов заявок, определяющее размерность матрицы. Однако из этого числа должны быть исключены так называемые некорректные матрицы приоритетов.

5.2.2. Корректность матрицы приоритетов

Корректность МП предполагает однозначность и определенность алгоритма, реализующего соответствующую МП. Не всякая матрица приоритетов, удовлетворяющая перечисленным выше требованиям, является

корректной. МП является некорректной, если при ее реализации может возникнуть неоднозначная ситуация, причем любое принятое решение будет противоречить заданной МП.

Поясним понятие корректности на примере МП для трех классов заявок $(H = 3)$ следующего вида:

$$Q = \begin{array}{c|ccc} & 1 & 2 & 3 \\ \hline 1 & 0 & 1 & 1 \\ 2 & 0 & 0 & 2 \\ 3 & 0 & 0 & 0 \end{array}.$$

Рассмотрим следующую ситуацию. Пусть в момент поступления заявки второго класса Z2 на обработке в устройстве находится заявка Z3 класса 3, и в очереди – одна или несколько заявок Z1 класса 1 (рисунок 22), которые не могут прервать обслуживание заявки класса 3, так как в соответствии с заданной МП они имеют только относительный приоритет по отношению к заявкам класса 3 ($q_{13} = 1$). При этом возникает следующая неопределенность. С одной стороны, поступившая заявка класса 2 должна прервать обслуживание заявки класса 3, поскольку по отношению к заявкам класса 3 имеет АП $(q_{23} = 2)$. С другой стороны, поступившая заявка класса 2 не может начать обслуживаться раньше заявок класса 1, находящихся в очереди и имеющих ОП по отношению к заявкам класса 2 ($q_{12} = 1$). Эта неопределенность приводит к неоднозначности при построении алгоритма, реализующего данную ДО. Любое решение из двух возможных (прервать и не прервать обслуживание заявки класса 3) приведет к дисциплине, не соответствующей заданной МП. Такие МП в дальнейшем исключим из рассмотрения, относя их к некорректным. При этом резко уменьшается число возможных вариантов построения МП (таблица 5).

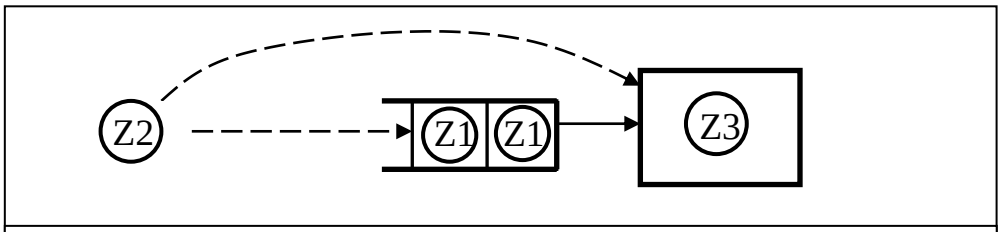

Рисунок 22. К понятию «корректность МП»

Отметим, что с ростом количества классов заявок количество различных ДО резко возрастет.

Для приближенной оценки количества корректных канонических МП при $H > 6$ экспериментальным путем с использованием метода математической индукции получено следующее выражение:

$$\tilde{\xi}_H \approx 1{,}5 \times 1{,}44^{H-2} H! \quad (H \geq 2).$$

Это выражение, как видно из таблицы 5, дает нижнюю оценку количества корректных канонических МП. Заметим, что $\tilde{\xi}_H$ определяет только количество канонических МП. Если допустить произвольные варианты заполнения МП, то

количество допустимых вариантов заполнения МП возрастает примерно ещё в $H!$ раз.

Таблица 5. Количество матриц приоритетов

Размерность МП	2x2	3x3	4x4	5x5	6x6
Количество вариантов ξ заполнения канонических МП	3	27	729	59049	более 14 млн.
Количество корректных канонических МП	3	13	75	541	4683
Оценка количества корректных канонических МП $\tilde{\xi}_H$	3	13	75	537	4644

Поскольку не всякий вариант заполнения МП является корректным, необходимо сформулировать правила, позволяющие формировать только корректные МП.

5.2.3. Правила построения корректных канонических матриц приоритетов

При построении корректных канонических МП необходимо соблюдать следующие правила.

1. ***Правило строки.*** После первого значащего (ненулевого) элемента в строке не должно быть нулевых элементов, т.е. если $q_{ij} = 1$ или 2, то $q_{ik} \neq 0$ для всех $k > j$ $(i, j, k = 1, \ldots, H)$.

2. ***Правило столбца.*** Элементы МП в пределах одного столбца должны образовывать невозрастающую последовательность: $q_{i+1\,j} \le q_{ij}$ $(i = 1, \ldots, H-1;$ $j = 1, \ldots, H)$.

3. ***Правило БП-группы.*** Классы заявок, образующие БП-группу, должны иметь одинаковые приоритеты по отношению к остальным классам заявок, т.е. если $q_{i\,i+1} = 0$, то $q_{i+1\,j} = q_{ij}$ для всех $j = 1, \ldots, H$. Другими словами, если после диагонального элемента в какой-либо строке матрицы приоритетов стоит нулевой элемент, то следующая стока должна совпадать с рассматриваемой.

Если не выполняется хотя бы одно из перечисленных правил, то матрица приоритетов будет некорректной.

В случае неканонической МП для ее проверки на соответствие перечисленным правилам необходимо преобразовать исходную матрицу в канонический вид путем перестановки строк и столбцов в соответствии с уровнем приоритетности, который подсчитывается как сумма всех элементов исходной матрицы в пределах каждой строки. Чем больше полученное значение, тем выше уровень приоритетности у соответствующего класса заявок.

5.2.4. Расчёт характеристик систем со смешанными приоритетами

Рассмотрим характеристики одноканальной СМО с накопителями неограниченной ёмкости, в которую поступают H классов заявок, образующие простейшие потоки с интенсивностями $\lambda_1,\ldots,\lambda_H$. Длительность обслуживания заявок класса k распределена по произвольному закону со средним значением $b_k = \theta_k / V$ и коэффициентом вариации $\nu_{b_k} = \nu_{\theta_k}$, где V - производительность системы (устройства), а θ_k и ν_{θ_k} - соответственно среднее значение и коэффициент вариации ресурсоёмкости обработки запросов класса k $(i,k=\overline{1,H})$. Выбор заявок из очереди на обслуживание осуществляется в соответствии с ДО СП, заданной с помощью МП.

Для дисциплины обслуживания со смешанными приоритетами (ДО СП) среднее время пребывания в системе заявок класса $k=\overline{1,H}$ определяется по формуле [9]:

$$u_k^{СП} = \frac{\sum_{i=1}^{H}(2-q_{ki})(1+q_{ki})\lambda_i b_i^2(1+\nu_{bi}^2)}{[2-\sum_{i=1}^{H}q_{ik}(3-q_{ik})\rho_i][2-\sum_{i=1}^{H}(1-q_{ki})(2-q_{ki})\rho_i]} + \frac{2b_k}{2-\sum_{i=1}^{H}q_{ik}(q_{ik}-1)\rho_i} \quad (68)$$

где q_{ik} $(i,k=\overline{1,H})$ – элементы матрицы приоритетов.

В правой части выражения (68) первое слагаемое определяет время ожидания начала обработки заявки класса k, а второе – время нахождения заявки на обработке, которое включает в себя время ожидания запроса в прерванном состоянии (при наличии АП со стороны запросов других классов) и непосредственно время обработки запроса в устройстве. Заметим, что если запросы всех других классов имеют по отношению к запросам класса k только ОП ($q_{ik}=1$ для всех $i \neq k$) или не имеют приоритета ($q_{ik}=0$ для всех $i \neq k$), то второе слагаемое в (68) станет равным длительности обработки b_k.

5.2.5. Постановка задачи проектирования систем со смешанными приоритетами

В системах с неоднородной нагрузкой циркулирует несколько классов заявок, к которым могут предъявляться различные требования в виде ограничений разных типов. Для рассматриваемых в дальнейшем моделей массового обслуживания с накопителями неограниченной ёмкости основными характеристиками функционирования служат времена пребывания в системе запросов разных классов, к которым предъявляются требования, задаваемые в виде предельных значений.

В простейшем случае ограничения задаются на средние значения времени пребывания запросов в системе:

$$u_k < u_k^* \quad (69)$$

и называются *средними* или относительными, в отличие от *вероятностных* (абсолютных) ограничений, задаваемых в виде

$$\Pr(\tau_{u_k} > u_k^*) < \delta_k^*, \qquad (70)$$

где u_k^* - допустимое значение времени пребывания в системе запросов класса k; $\Pr(\tau_{u_k} > u_k^*)$ – вероятность того, что время пребывания τ_{u_k} запросов класса k превысит допустимое u_k^*; δ_k^* – допустимая вероятность превышения заданного ограничения u_k^* $(k = \overline{1,H})$.

Ограничения (69) и (70) могут быть заданы для всех классов запросов $(k = \overline{1,H})$, поступающих в систему, или только для некоторых из них.

Вероятностные ограничения (70) являются более жесткими по сравнению с (69). При этом задача проектирования решается по аналогии с относительными ограничениями, но сопровождается более громоздкими математическими выкладками и расчётами, связанными с операциями над функциями распределений характеристик обслуживания запросов.

В общем случае, ограничения могут представлять собой комбинацию двух указанных ограничений:

$$\left.\begin{array}{ll} \Pr(\tau_{u_k} > u_k^*) < \delta_k^* & \text{для} \quad k \in \mathcal{H}_1; \\ u_k < u_k^* & \text{для} \quad k \in \mathcal{H}_2, \end{array}\right\},$$

где $\mathcal{H}_1$ и $\mathcal{H}_2$ – непересекающиеся подмножества номеров классов запросов: $\mathcal{H}_1 \bigcap \mathcal{H}_2 = \phi$, причём для подмножества классов $\mathcal{H}_3$ ограничения могут быть не заданы.

Очевидно, что $\mathcal{H}_1 \bigcup \mathcal{H}_2 \bigcup \mathcal{H}_3 = \mathcal{H}$, где $\mathcal{H} = \{1,2,\ldots,H\}$ – множество всех классов запросов, поступающих в систему.

В системах с неоднородной нагрузкой, в качестве моделей которых используются системы с накопителями ограниченной ёмкости, дополнительно могут быть заданы ограничения на вероятность потери запросов для всех H классов запросов, поступающих в систему, или только для некоторых из них:

$$\pi_k < \pi_k^*. \qquad (71)$$

В качестве критерия эффективности при решении задач проектирования реальных систем часто используется стоимостной критерий.

Задача *функционального проектирования* для ограничений любого вида обычно решается с использованием достаточно простых моделей, в частности базовых моделей с неоднородным потоком запросов, и состоит в синтезе дисциплины обслуживания поступающих в систему запросов разных классов.

5.3. Функциональное проектирование систем реального времени

Рассмотрим задачу распределения приоритетов, решаемую в процессе синтеза дисциплины обслуживания запросов в информационно-управляющих системах (ИУС), находящихся в контуре систем автоматического управления и функционирующих в режиме реального времени. Примерами таких систем могут служить системы управления производственными процессами и подвижными объектами, характерной особенностью которых является наличие жестких ограничений на время реакции (ответа), превышение которых

недопустимо или же крайне нежелательно, поскольку это может привести к резкому ухудшению качества функционирования управляемого объекта или вообще к выходу его из строя. Поскольку указанные ограничения во многих случаях могут составлять доли миллисекунд и даже микросекунд, то одна из особенностей таких систем заключается в отсутствии обмена с внешней памятью в процессе управления. Таким образом, внешняя память не влияет на эффективность функционирования системы в целом и, в частности, на время реакции, являющееся основной характеристикой функционирования ИУС.

Задача функционального проектирования ИУС для ограничений любого вида решается с использованием базовой модели в виде СМО с неоднородным потоком запросов и, в простейшем случае, сводится к выбору стратегии управления вычислительным процессом, в частности, стратегии управления потоком поступающих в систему запросов, задаваемой в виде определенной дисциплины обслуживания (ДО) запросов.

В общем случае функциональное проектирование реализуется в три этапа.

1. Определение нижней значения производительности системы, представляющего собой границу, начиная с которой можно искать стратегию управления вычислительным процессом (дисциплину обслуживания), обеспечивающую выполнение заданных ограничений.

2. Синтез стратегии управления вычислительным процессом (метода диспетчеризации поступающих в систему запросов), заключающийся, в простейшем случае, в выборе ДО, наилучшим образом обеспечивающей выполнение заданных ограничений.

3. Определение оптимальной производительности системы, обеспечивающей требуемое качество обслуживания запросов для выбранной ДО при минимальной стоимости системы.

5.3.1. Постановка задачи функционального проектирования

В качестве исходных данных для решения задачи функционального проектирования ИУС с использованием базовой модели в виде одноканальной СМО с неоднородным потоком заявок служит следующая совокупность величин:

1) количество классов запросов H, поступающих в систему;

2) интенсивности $\lambda_1,\ldots,\lambda_H$ потоков запросов, которые полагаются простейшими;

3) средние значения ресурсоёмкостей $\theta_1,\ldots,\theta_H$ обработки запросов, задаваемые в виде среднего числа команд (инструкций), выполняемых в ЦП при обработке одного запроса соответствующего класса;

4) коэффициенты вариации $\nu_1,\ldots,\nu_H$ ресурсоёмкостей обработки запросов;

5) ограничения $u_1^*,\ldots,u_H^*$ на средние значения времени пребывания в системе запросов классов $1,\ldots,H$, заданные в виде (69).

Задержки, т.е. времена пребывания $u_1,\ldots,u_H$ запросов в системе, зависят в данном случае от производительности (быстродействия) V системы и дисциплины обслуживания запросов. Очевидно, что требуемое качество функционирования системы, заданное в виде ограничений $u_1^*,\ldots,u_H^*$ всегда может быть достигнуто только за счет производительности системы при любой ДО, т.е. возможно множество решений, из которых должно быть выбрано оптимальное. В качестве критерия эффективности можно выбрать стоимость системы, отражающую экономическую целесообразность предлагаемого решения. Из всех возможных решений наилучшим является такое, при котором стоимость системы минимальна. В этом случае задача функционального проектирования формулируется следующим образом: выбрать ДО и определить производительность системы, которые обеспечивают выполнение заданных ограничений $u_1^*,\ldots,u_H^*$ при минимальной стоимости системы.

Применительно к рассматриваемой базовой модели стоимость системы определяется как

$$S = S_1 + S_2,$$

где $S_1 = \xi V^{\chi}$ – стоимость процессора, связанная прямопропорциональной зависимостью с его производительностью (быстродействием) V через коэффициенты пропорциональности ξ и нелинейности χ; $S_2 = S_{\text{П}} + s_0 E$ – стоимость памяти, складывающаяся из стоимости памяти $S_{\text{П}}$, отводимой под хранение программ (системных и прикладных), и стоимости памяти, отводимой под хранение данных, поступающих в систему в виде запросов на обработку прикладными программами, и определяемой как произведение стоимости s_0 одной единицы памяти (например, байта) на требуемую ёмкость E памяти для хранения запросов.

Ёмкость памяти E определяется максимальным числом запросов класса k (k-запросов) $\hat{m}_k$, которые могут одновременно находиться в системе:

$$E = \sum_{k=1}^{H} d_k \hat{m}_k,$$

где d_k – объём памяти, занимаемый одним запросом класса k.

Максимальное число запросов $\hat{m}_k$ в системе, по аналогии с пунктом 4.2.5, может быть представлено как $\hat{m}_k = f_k m_k$, где $m_k = \lambda_k u_k$ – среднее число запросов в системе, f_k – коэффициент, зависящий от закона распределения числа запросов в системе $(k = \overline{1,H})$. Таким образом, стоимость системы запишется в следующем виде:

$$S = \xi V^{\chi} + S_{\text{П}} + s_0 \sum_{k=1}^{H} d_k f_k \lambda_k u_k(\text{ДО}, V). \qquad (72)$$

где $u_k(\text{ДО},V)$ – среднее время пребывания в системе k-запросов, являющееся функцией дисциплины обслуживания (ДО) и производительности V.

Выражение (72) используется в качестве критерия эффективности для выбора ДО и оптимальной производительности системы.

5.3.2. Оценка нижней границы производительности процессора

На предварительном этапе функционального проектирования определяется нижняя граница производительности системы, начиная с которой может решаться задача выбора ДО. Другими словами, если производительность системы будет меньше, чем нижняя граница, то можно утверждать, что не может быть найдена ДО, обеспечивающая требуемое качество функционирования ВС, заданное в виде ограничений (69). Кроме того, нижняя граница производительности служит мерой качества решения задачи выбора ДО. Отметим, что нижняя граница производительности определяется безотносительно к какой-либо ДО.

Одним из основных требований, предъявляемых к любой системе, является отсутствие перегрузок, т.е. требование существования стационарного режима функционирования, при котором система справляется с возложенной на нее нагрузкой. Из условия отсутствия перегрузок в системе: $R<1$, где R - суммарная загрузка системы, легко получить следующее неравенство:

$$V > \sum_{i=1}^{H} \lambda_i \theta_i . \qquad (73)$$

Выражение (73) можно рассматривать как условие выбора производительности системы при отсутствии ограничений на время пребывания запросов, где $V_0^{'} = \sum_{i=1}^{H} \lambda_i \theta_i$ определяет нижнюю границу производительности, обеспечивающую отсутствие перегрузок в системе. Заметим, что при $V = V_0^{'}$ загрузка системы $R = 1$.

При наличии ограничений на время пребывания запросов в системе под нижней границей производительности $V_0^{''}$ понимается значение, начиная с которого может искаться ДО, обеспечивающая выполнение заданных ограничений. Другими словами, если производительность системы будет меньше, чем $V_0^{''}$ (но больше, чем $V_0^{'}$), то можно гарантировать, что не найдется ДО, при которой будет обеспечено выполнение заданных ограничений.

Для определения нижнего значения производительности, учитывающего средние ограничения (69) на время пребывания запросов по всем классам, воспользуемся законом сохранения времени пребывания (62).

Заменяя в выражении (62) среднее значение времени пребывания u_i на u_i^* и учитывая неравенство (69), получим

$$\sum_{i=1}^{H} \rho_i u_i^* > \frac{R}{2(1-R)} \sum_{i=1}^{H} \lambda_i b_i^2 (1+\nu_i^2) + \sum_{i=1}^{H} \rho_i b_i . \qquad (74)$$

Выражение (74) представляет собой необходимое условие существования ДО, при которой будут выполняться ограничения (69). Это означает, что если указанное условие, вычисленное при некотором значении производительности

системы V, выполняется, то при этом значении V можно пытаться искать ДО, которая обеспечит выполнение заданных ограничений $u_1^*,\ldots,u_H^*$. В противном случае, если условие (74) не выполняется, то при заданном значении производительности V не может быть найдена ДО, обеспечивающая выполнение ограничений $u_1^*,\ldots,u_H^*$. Отметим, что условие (74) является необходимым, но не достаточным.

Для определения нижнего значения производительности системы заменим в (74) все величины, зависящие от производительности V, на соответствующие выражения:

$$b_i = \frac{\theta_i}{V}; \qquad \rho_i = \frac{\lambda_i \theta_i}{V}; \qquad R = \frac{1}{V}\sum_{i=1}^{H} \lambda_i \theta_i .$$

Подставляя эти выражения в (74) и обозначая:

$$A = \sum_{i=1}^{H} \lambda_i \theta_i; \quad B = \sum_{i=1}^{H} \lambda_i \theta_i^2; \quad C = \sum_{i=1}^{H} \lambda_i \theta_i u_i^*; \quad D = \sum_{i=1}^{H} \lambda_i \theta_i^2 (1+\nu_i^2). \qquad (75)$$

после некоторых преобразований получим квадратное неравенство:

$$V^2 - \left(A + \frac{B}{C}\right)V + \frac{A}{2C}(2B - D) > 0 .$$

Решая это неравенство относительно V, окончательно получим:

$$V > \frac{1}{2}\left(A + \frac{B}{C}\right) + \left[\frac{1}{4}\left(A + \frac{B}{C}\right)^2 - \frac{A(2B-D)}{2C}\right]^{1/2} . \qquad (76)$$

Из двух возможных решений выбрано одно, поскольку второе решение со знаком минус перед квадратным корнем меньше значения производительности $V_0^{'}$, необходимого для обеспечения отсутствия перегрузок.

Обозначим через $V_0^{''}$ выражение, стоящее в правой части неравенства (76). Значение $V_0^{''}$ представляет собой нижнее значение производительности системы, учитывающее ограничения $u_1^*,\ldots,u_H^*$ на время пребывания запросов по всем классам: $V_0^{''} = f(u_i^* \mid i = \overline{1,H})$.

На рисунке 23 показана зависимость нижнего значения производительности системы $V_0^{''}$ от ограничений на время пребывания запросов, представленных в виде вектора $U^{''} = (u_1^*,\ldots,u_H^*)$. Как и следовало ожидать, с уменьшением значений $u_1^*,\ldots,u_H^*$ растет требуемое значение производительности, а при больших значениях $u_1^*,\ldots,u_H^*$ производительность системы $V_0^{''}$

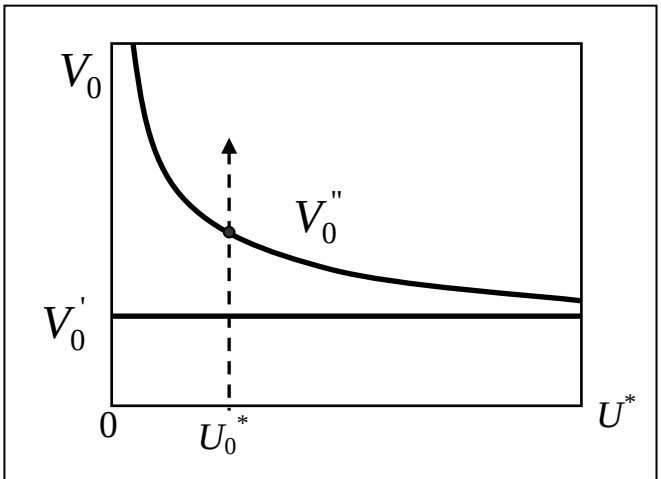

Рисунок 23. Нижняя граница производительности

стремится к значению $V_0^{'}$.

Значение производительности системы, определяемое из выражения (76), было получено из необходимого условия (74) существования ДО. Это значение обеспечивает выполнение ограничений (69), и поэтому может рассматриваться как необходимое значение производительности, но не достаточное для выбора ДО. Кроме того, в выражении (76) ограничения на время пребывания запросов учитываются по всем классам в целом и не учитываются особенности каждого класса. Это проявляется, например, в том, что если для какого-нибудь одного класса заявок задать ограничение $u_k^* = 0$, а для остальных классов $i \neq k$ ограничения $u_i^* > 0$, то на практике это будет означать, что система должна иметь бесконечно большую производительность $V = \infty$ для того, чтобы обеспечить ограничение $u_k^* = 0$. В то же время, из неравенства (76) вытекает, что система может иметь производительность $V < \infty$, что заведомо неверно.

Таким образом, при оценке нижнего значения производительности системы необходимо учитывать особенности каждого класса запросов в отдельности, т.е. необходимо определить некоторые значения производительности $V_1^{'''},\ldots,V_H^{'''}$, учитывающие ограничения на времена пребывания запросов по каждому классу. Указанные значения могут быть получены на основе следующих рассуждений.

Минимальное время пребывания в системе для k-запросов может быть обеспечено только в том случае, если этому классу присвоить самый высокий АП по отношению ко всем другим классам. Тогда это время будет равно

$$u_k = \frac{\lambda_k b_k^2 (1+\nu_k^2)}{2(1-\rho_k)} + b_k,$$

которое согласно (69) должно быть меньше u_k^*:

$$\frac{\lambda_k b_k^2 (1+\nu_k^2)}{2(1-\rho_k)} + b_k < u_k^* \qquad (k = \overline{1,H}). \tag{77}$$

Отсюда можно найти минимальное значение производительности $V_k^{'''}$, при котором будет выполнено заданное ограничение для k-запросов при условии, что этим запросам будет обеспечено минимально возможное время пребывания за счет предоставления им самого высокого АП. Это значение производительности является нижним для k-запросов в том смысле, что если производительность системы окажется меньше, чем $V_k^{'''}$, то не удастся подобрать ДО, при которой будет выполнено ограничение на время пребывания запросов данного класса.

Заменяя в выражении (77) величины, зависящие от производительности, и используя по аналогии с (75) обозначения:

$$A_k = \lambda_k \theta_k; \quad B_k = \lambda_k \theta_k^2; \quad D_k = \lambda_k \theta_k^2 (1+\nu_k^2),$$

после некоторых преобразований получим:

$$V > \frac{1}{2}\left(A_k + \frac{\theta_k}{u_k^*} \right) + \left[\frac{1}{4}\left(A_k + \frac{\theta_k}{u_k^*} \right)^2 - \frac{2B_k - D_k}{2u_k^*} \right]^{1/2}. \qquad (78)$$

Обозначим через V_k''' выражение, стоящее в правой части неравенства (78). Значение V_k''' представляет собой нижнюю границу производительности, учитывающую ограничение $u_k < u_k^*$, причем значение производительности системы должно выбираться из условия $V > V_k'''$ для всех $k = \overline{1, H}$.

Отметим, что выражение (78) совпадает с точностью до обозначений с выражением (76), в котором члены A, B, C, D заменяются соответственно на A_k, B_k, C_k, D_k, где $C_k = \lambda_k \theta_k u_k^*$.

Характер зависимости V_k''' от ограничения u_k^* аналогичен рисунку 23, причем при $u_k^* \to \infty$, что в пределе соответствует случаю отсутствия ограничения, производительность V_k''' принимает значение $V_k''' = \lambda_k \theta_k$, определяющее отсутствие перегрузок системы при условии, что в системе обслуживаются только k-запросы.

Таким образом, нижнее значение производительности системы V_0 находится как $V_0 = \max(V_0', V_0'', V_1''', \ldots, V_H''')$ или с учетом того, что $V_0' < V_0''$, окончательно получим: $V_0 = \max(V_0'', V_1''', \ldots, V_H''')$

Для приближенного расчёта значений V_k''' в некоторых случаях можно воспользоваться более простым выражением: $V_k''' = \theta_k / u_k^*$, которое выводится из условия $b_k < u_k^*$, полученного в предположении, что время ожидания w_k запросов класса k при предоставлении им АП по отношению к запросам других классов пренебрежимо мало по сравнению с длительностью обслуживания b_k, т.е. $w_k << b_k$. С учетом того, что $b_k = \theta_k / V$, окончательно имеем: $V > \theta_k / u_k^* = \tilde{V}_k'''$ для всех $k = \overline{1, H}$.

Приближенное значение $\tilde{V}_k'''$ тем ближе к точному, чем меньше загрузка, создаваемая k-запросами. Степень загрузки системы k-запросами можно оценить величиной произведения $\lambda_k \theta_k$. Для классов с минимальным значением этого произведения допускается использовать выражение для приближенного расчёта $\tilde{V}_k'''$, для классов с максимальным значением $\lambda_k \theta_k$ рекомендуется использовать выражение (78).

5.3.3. Синтез дисциплины обслуживания

Одним из наиболее важных и трудоемких этапов функционального проектирования ВС, в частности систем реального времени, является этап синтеза дисциплины обслуживания (ДО), связанный с назначением приоритетов классам запросов.

Разработка высокоэффективных и хорошо формализованных алгоритмов синтеза ДО, позволяющих получать однозначное решение и гарантировать его оптимальность, представляет собой сложную задачу. При наличии ограничений на время пребывания запросов для синтеза ДО целесообразно использовать эвристические алгоритмы, сводящиеся к целенаправленному перебору некоторого множества ДО.

Ниже излагается алгоритм синтеза ДО, основанный на результатах анализа ДО со смешанными приоритетами (СП) как дисциплин наиболее общего вида. При этом задача заключается в определении таких элементов q_{ij} матрицы приоритетов (МП) **Q**, описывающей ДО СП, при которых выполняются заданные ограничения (69) на время пребывания запросов разных классов, где среднее время $u_k^{\text{СП}}$ пребывания k-запросов рассчитывается по формуле (68). Аналитическое решение системы неравенств (69) не представляется возможным и единственный путь состоит в переборе некоторого множества МП на компьютере, поскольку число различных ДО СП даже при небольшом числе классов запросов оказывается значительным. Последовательный перебор всех возможных МП оказывается нецелесообразным, так как приводит к большим затратам машинного времени. Уменьшение этого времени достигается за счет использования эвристических алгоритмов, основанных на целенаправленном переборе различных ДО.

Проиллюстрируем на графике задачу выбора ДО и выявим эффект, достигаемый за счет перехода от одной ДО к другой.

Положим, что в систему поступают запросы пяти классов, пронумерованных таким образом, что ограничения на времена пребывания запросов разных классов связаны соотношением $u_k^* \le u_{k+1}^*$ для $k = \overline{1, H-1}$ и имеют вид, показанный на рисунке 24.

Пусть при найденном нижнем значении V_0 производительности системы в качестве исходного решения выбрана ДО БП, обозначенная на графике как БП_0 (прямая БП_0 построена в предположении, что ресурсоёмкости обработки запросов всех классов одинаковы). Ясно, что такое решение не может быть признано удовлетворительным, поскольку для запросов классов 1 и 2 времена пребывания превышают допустимые значения u_1^* и u_2^*. Увеличивая производительность системы до значения $V_1 > V_0$ можно добиться того, чтобы ограничения u_k^* выполнялись для всех классов при выбранной ДО БП (прямая БП_1).

Полученное решение, хотя и удовлетворяет ограничениям, очевидно, не может считаться наилучшим, так как для запросов классов $k = \overline{1, H}$ слишком сильно различается запас по времени пребывания, измеряемый как разность $(u_k^* - u_k)$. При этом для наиболее ценных запросов класса 1 этот запас минимальный, в то время как для менее ценных запросов классов 4 и 5 этот запас достаточно большой.

Для увеличения запаса присвоим ОП запросам с меньшим номером, т.е. перейдем к ДО ОП. Согласно закону сохранения времени пребывания получим, что при той же производительности системы V_1 время пребывания высокоприоритетных запросов уменьшится за счет увеличения времени пребывания запросов низкоприоритетных классов (кривая $ОП_1$). Будем теперь уменьшать производительности системы до того минимально возможного значения V_2, при котором ещё выполняются ограничения на времена пребывания. При этом увеличатся времена пребывания запросов всех классов (кривая $ОП_2$). Таким образом, мы получили новое решение в виде ДО ОП. Очевидно, что это решение лучше предыдущего ДО БП, так как позволяет обеспечить заданные ограничения u_k^*, при меньшем значении производительности системы: $V_2 < V_1$.

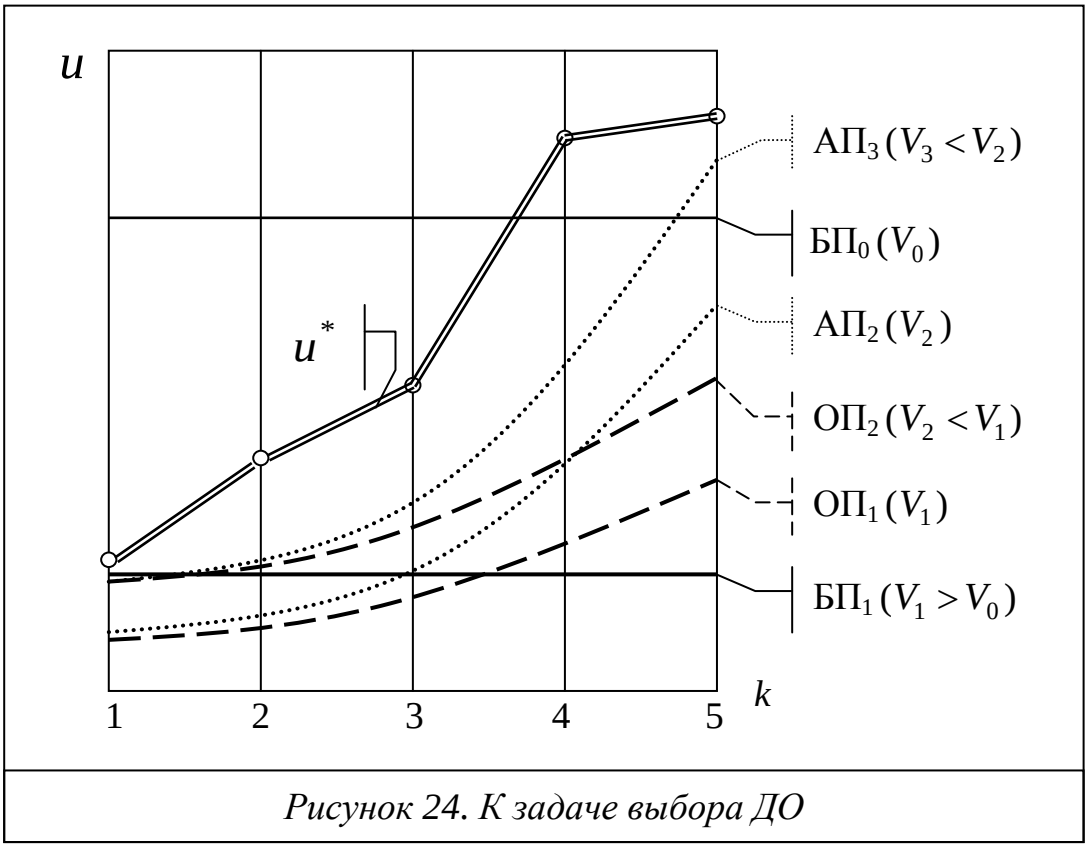

Рисунок 24. К задаче выбора ДО

Перейдем теперь к ДО АП при той же производительности V_2 (кривая $АП_2$). По аналогии с ДО ОП для ДО АП можно, уменьшая производительность системы, обеспечить заданные ограничения при $V_3 < V_2$ (кривая $АП_3$), т.е. получить новое более эффективное решение. Последующие шаги, связанные с улучшением ДО, заключаются в переходе к ДО СП и получении новых решений, обеспечивающих выполнение ограничений u_k^* при меньших значениях производительности системы. Процесс последовательного перебора ДО продолжается до тех пор, пока не будет найдена ДО, обеспечивающая выполнение ограничений на времена пребывания запросов всех классов при минимальном значении производительности системы. Показателем качества

выбранной ДО может служить разность $(V_Д - V_0)$, где $V_Д$ – минимальное значение производительности системы, при котором выполняются ограничения u_k^* для выбранной ДО; V_0 – нижнее значение производительности системы. Чем меньше эта разность, тем лучше решение. Отметим, что для любой ДО: $V_Д \geq V_0$.

При использовании алгоритма распределения приоритетов, основанного на целенаправленном переборе различных ДО, первоначально необходимо задать некоторый исходный вариант назначения приоритетов. Для этого рекомендуется переупорядочить классы запросов в порядке убывания отношения ресурсоёмкости θ_k обслуживания запросов к допустимому времени пребывания u_k^*, т.е. по правилу: $\theta_\alpha / u_\alpha^* > \theta_\beta / u_\beta^* > \ldots > \theta_\omega / u_\omega^*$. В результате упорядочения классов запросов получается некоторая последовательность номеров: $\alpha, \beta, \ldots \omega$, для которой формируется первоначальный вариант назначения приоритетов по правилу: классам запросов, расположенным правее в указанной последовательности назначается приоритет не выше, чем классам, расположенным левее. В нашем случае наиболее высокий приоритет назначается запросам класса α и самый низкий приоритет – запросам класса ω. В простейшем случае в качестве первоначального варианта может быть выбрана ДО ОП, ДО АП или ДО БП.

Необходимость упорядочения классов запросов по указанному правилу иллюстрируется на рисунке 25, из которого видно, что из-за большой ресурсоёмкости обслуживания θ_j $(b_j = \theta_j / V)$ запросов класса j (j-запросов), по сравнению с запросами класса i, целесообразно более высокий приоритет назначить j-запросам, несмотря на то, что $u_j^* > u_i^*$. Это необходимо для обеспечения меньшего времени ожидания j-запросов: $w_j^* < w_i^*$ (см. рисунок 25).

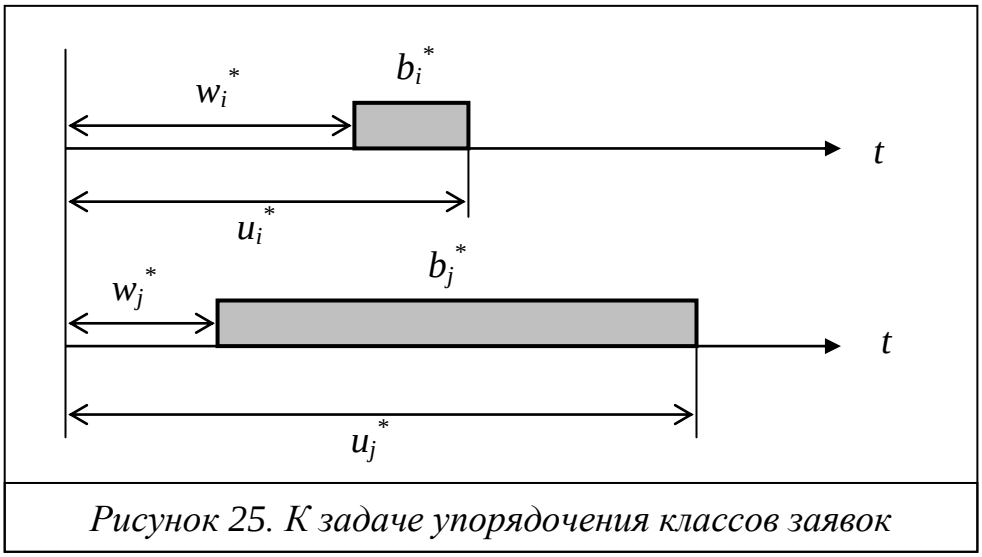

Рисунок 25. К задаче упорядочения классов заявок

Каждый последующий вариант назначения приоритетов формируется по результатам предыдущего варианта. Показателем, определяющим необходимость изменения приоритета запросов некоторого класса, может служить относительное отклонение ζ_{u_k} времени пребывания u_k, полученного

для рассматриваемого варианта назначения приоритетов, от заданного ограничения $\zeta_{u_k} = (u_k^* - u_k)/u_k^* \quad (k = \overline{1,H})$. При этом, в первую очередь, необходимо увеличивать приоритет для того класса запросов, для которого ζ_{u_k} минимально. Изменение приоритета может осуществляться: 1) изменением приоритета данного класса по отношению к другим классам; 2) изменением приоритета других классов по отношению к данному классу. Необходимо стремиться к тому, чтобы значения ζ_{u_k} были приблизительно одинаковы.

В тех случаях, когда параметры входных потоков запросов довольно резко меняются во времени, в системах реального времени могут использоваться ДО с динамическими приоритетами. При этом необходимо помнить, что ДО с динамическими приоритетами более сложны в реализации, что выражается в дополнительных аппаратурных и программных затратах, а также в значительных непроизводительных затратах на диспетчеризацию, которые в некоторых случаях превышают эффект, достигаемый за счет использования динамической диспетчеризации. Поэтому целесообразность применения динамической диспетчеризации должна быть оценена в каждом конкретном случае.

5.3.4. Определение оптимальной производительности системы

На последнем этапе определяется оптимальное значение производительности системы, которое, обеспечивая при выбранной ДО выполнение заданных ограничений (69) на время пребывания запросов, позволяет создать систему минимальной стоимости S, определяемой выражением (72).

Задача отыскания минимума функции S сводится к решению относительно производительности V уравнения, полученного путем приравнивания к нулю производной по V от функции S:

$$\frac{\mathrm{d}S}{\mathrm{d}V} = \xi \chi V^{\chi-1} + s_0 \sum_{k=1}^{H} d_k f_k \lambda_k \frac{\mathrm{d}u_k}{\mathrm{d}V} = 0. \qquad (79)$$

где u_k определяется на основе (68) для выбранной на предыдущем этапе ДО.

Уравнение (79) должно решаться с учетом ограничений (69), к которым на данном этапе могут быть добавлены ограничения на производительность V^* системы и ёмкость накопителя E^*.

Ограничения определяют область допустимых значений V, в которой ищется оптимальная производительность V_{opt}. В частности, решая систему неравенств $u_k(V) \le u_k^*$, находим, что V должно быть больше, чем $V_1, \ldots V_H$, т.е. $V > \max\{V_1, \ldots, V_H\}$, где V_k – значение производительности, при котором $u_k(V) = u_k^* \quad (k = \overline{1,H})$. Верхней границей производительности может служить ограничение V^*. Таким образом, V_{opt} должно удовлетворять условию:

$$\max\{V_1, \ldots, V_H\} < V_{opt} \le V^*.$$

Если $V_{opt} < \max\{V_1,\ldots,V_H\}$, то значение производительности должно быть выбрано так, чтобы выполнялись ограничения на время пребывания запросов всех классов, несмотря на то, что стоимость S системы окажется больше, чем минимально возможная при V_{opt}. Если $V_{opt} > V^*$, то это означает, что необходимо строить систему с несколькими параллельными устройствами, так как производительность системы с одним устройством не достаточна для обеспечения заданных ограничений (69). При этом число устройств в системе определяется как ближайшее целое большее V_{opt}/V^*.

На рисунке 26 задача определения оптимального значения производительности иллюстрируется графически.

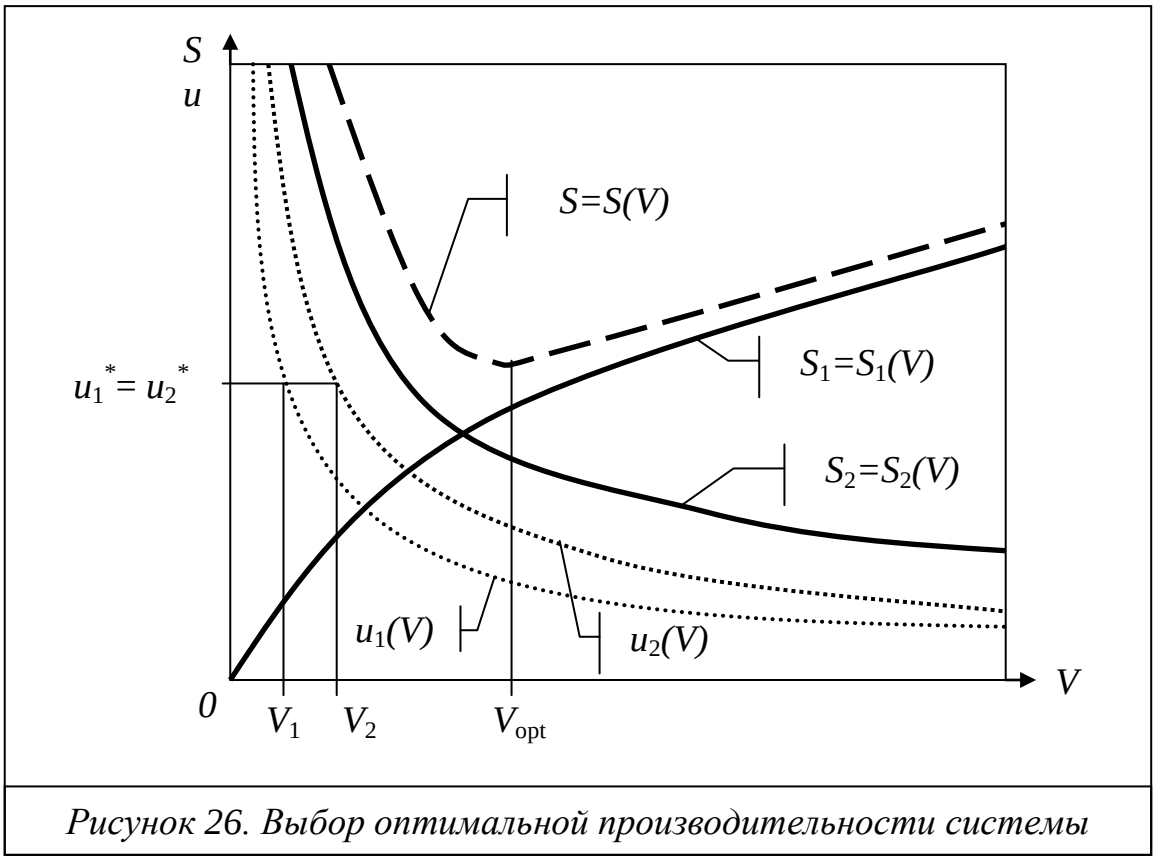

Рисунок 26. Выбор оптимальной производительности системы

С увеличением производительности V стоимость S_1 устройства растет, а стоимость S_2 накопителя, используемого для хранения запросов, уменьшается, причем характер зависимости S_2 от V аналогичен характеру зависимости u_k от V. Значение $V = V_{opt}$, при котором стоимость $S = S_1 + S_2$ принимает минимальное значение, является оптимальным. Здесь же на графике показаны зависимости среднего времени пребывания $u_1(V)$ и $u_2(V)$ запросов двух классов, для которых заданы одинаковые ограничения $u_1^* = u_2^*$. Для этих ограничений найдены значения производительности V_1 и V_2, определяющие нижнюю границу области существования решения, т.е. V_{opt} должно

удовлетворять условию: $V_{opt} > V_2$. Как видно из графика, V_{opt} удовлетворяет данному условию.

5.3.5. Проектирование систем реального времени с вероятностными ограничениями

Ограничения на время реакции в системах реального времени обычно задаются в вероятностном виде (70). При этом значительно усложняется решение задачи синтеза ДО, обусловленное необходимостью проведения математических выкладок на уровне функций распределений для определения вероятности превышения заданного ограничения на время задержки как

$$\Pr(\tau_{u_k} > u_k^*) = 1 - U_k(u_k^*),$$

где $U_k(\tau)$ – функция распределения времени пребывания (задержки) k-запросов в системе $(k = \overline{1, H})$.

Таким образом, необходимо знать закон распределения задержки, который, в свою очередь, зависит от выбранной дисциплины обслуживания запросов.

Задача синтеза ДО для систем с вероятностными ограничениями может быть сведена к задаче синтеза со средними ограничениями путем пересчета вероятностных ограничений в средние.

Для этого воспользуемся функцией распределения $U_k(\tau)$ времени пребывания τ_{u_k} в системе k-запросов и, рассматривая ограничение (70) на границе, запишем его в следующем виде:

$$1 - U_k(u_k^*) = \delta_k^*.$$

Решая полученное уравнение, найдем оценку среднего ограничения как функцию допустимого времени пребывания u_k^* и вероятности его превышения δ_k^*: $\tilde{u}_k = \varphi(u_k^*, \delta_k^*)$.

Например, для экспоненциального закона $1 - U_k(u_k^*) = \mathrm{e}^{-u_k^*/\tilde{u}_k} = \delta_k^*$, откуда получим: $\tilde{u}_k = -u_k^* / \ln \delta_k^*$. Полагая $\delta_k^* = 10^{-n}$, после некоторых преобразований окончательно получим: $\tilde{u}_k = 0{,}435\, u_k^* / n \quad (k = \overline{1, H})$.

Полученная таким образом оценка $\tilde{u}_k$ может использоваться вместо u_k^* в выражении (69), что позволяет свести задачу синтеза ДО для систем с вероятностными ограничениями (70) к задаче синтеза со средними ограничениями:

$$u_k < \tilde{u}_k \quad (k = \overline{1, H}).$$

Основная проблема при таком подходе заключается в необходимости знать закон распределения $U_k(\tau)$ времени пребывания τ_{u_k} в системе k-запросов $(k = \overline{1, H})$, который в общем случае не может быть получен в явном виде.

Задача функционального проектирования систем с вероятностными ограничениями может быть упрощена, если воспользоваться аппроксимацией распределения $U_k(\tau)$ по двум моментам u_k и $u_k^{(2)}$, выражения для которых получены в явном виде [10]. Тогда в зависимости от значения коэффициента вариации, рассчитываемого как $\nu_{u_k} = \sqrt{u_k^{(2)} - u_k^2}\,/u_k$, в качестве аппроксимирующих законов распределений могут использоваться распределение Эрланга, если $0 \le \nu_{u_k} \le 1$, или гиперэкспоненциальное распределение, если $\nu_{u_k} > 1$.

5.4. Системы с несколькими устройствами и неоднородной нагрузкой

Рассмотрим систему (рисунок 27), содержащую K идентичных устройств и накопитель неограниченной ёмкости, в которую поступают H классов заявок, образующие простейшие потоки с интенсивностями $\lambda_1,\ldots,\lambda_H$. Длительность τ_{b_k} обработки заявок класса k распределена по экспоненциальному закону со средним значением b_k. Выбор заявок из очереди на обработку осуществляется в соответствии с некоторой заданной дисциплиной обслуживания (ДО).

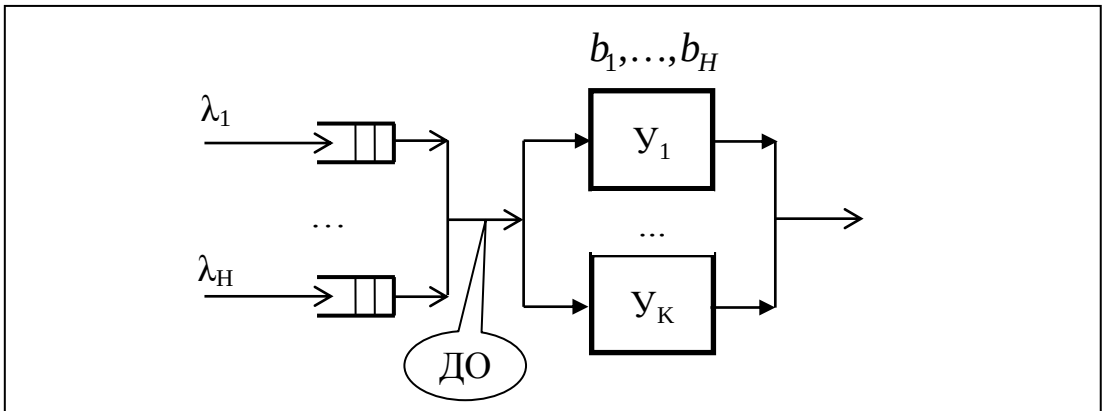

Рисунок 27. Многоканальная система с неоднородной нагрузкой

В качестве основной характеристики, описывающей эффективность функционирования системы, будем рассматривать среднее время ожидания w_k заявок класса $k = 1,\ldots,H$.

Точные методы расчёта характеристик функционирования системы с несколькими устройствами и неоднородным потоком заявок разработаны для ДО БП, ДО ОП, ДО АП при следующих предположениях:

- потоки заявок простейшие;
- длительность обработки заявок каждого класса распределена по экспоненциальному закону с одним и тем же средним значением $b_k = b = \dfrac{\theta}{V} = \text{const}$ для всех $k = 1,\ldots,H$, где θ - средняя ресурсоёмкость обработки; V – производительность (быстродействие) одного устройства;

- все K устройств – идентичны, и любая заявка может быть обработана любым устройством;
- в системе отсутствуют перегрузки, т.е. суммарная загрузка системы меньше 1: $R = \sum_{i=1}^{H} \frac{\lambda_i b}{K} < 1$.

5.4.1. Расчёт характеристик систем с бесприоритетным обслуживанием заявок

Для ДО БП, когда заявки выбираются на обработку в порядке поступления независимо от номера класса, расчёт характеристик системы с неоднородным потоком может быть сведён к расчёту характеристик системы с однородным потоком, который поступает в систему с интенсивностью

$$\Lambda = \sum_{i=1}^{H} \lambda_i$$

и создаёт в системе суммарную нагрузку и загрузку соответственно:

$$Y = \sum_{i=1}^{H} y_i = \sum_{i=1}^{H} \lambda_i b_i ; \qquad R = \sum_{i=1}^{H} \rho_i = \sum_{i=1}^{H} \frac{\lambda_i b}{K} = \frac{\Lambda b}{K}. \tag{80}$$

Для того чтобы в системе не было перегрузок, необходимо выполнение условия: $R < 1$ или, что то же самое, $Y < K$.

Тогда вероятность нахождения в системе k заявок

$$p_k = \begin{cases} \frac{Y^k}{k!} p_0 & (k = \overline{1, K-1}); \\ \frac{K^K}{K!} \left(\frac{Y}{K}\right)^k p_0 & (k = \overline{K, \infty}), \end{cases}$$

где p_0 – вероятность того, что в системе нет ни одной заявки (система простаивает), рассчитывается по формуле:

$$p_0 = \left[\sum_{k=0}^{K-1} \frac{Y^k}{k!} + \frac{Y^K}{(K-1)!(K-Y)} \right]^{-1} \qquad (Y < K).$$

а вероятность того, что все K устройств заняты обслуживанием заявок, определяется как

$$P = \frac{Y^K}{(K-1)!(K-Y)} p_0 \qquad (Y < K)$$

,

Средняя длина очереди заявок всех классов и среднее время ожидания при ДО БП соответственно равны:

$$l^{\text{БП}} = \frac{Y^{K+1}}{(K-1)!(K-Y)^2} p_0, \qquad w^{\text{БП}} = \frac{Pb}{K-Y}.$$

Эти же зависимости могут быть представлены как функции загрузки системы R, причем $R < 1$, путём замены $Y = KR$:

$$p_0 = \left[\sum_{k=0}^{K-1} \frac{(KR)^k}{k!} + \frac{(KR)^K}{K!(1-R)} \right]^{-1};$$

$$P = \frac{(KR)^K}{K!(1-R)} p_0; \qquad (81)$$

$$l^{\text{БП}} = \frac{K^K R^{K+1}}{K!(1-R)^2} p_0; \qquad (82)$$

$$w^{\text{БП}} = \frac{Pb}{K(1-R)}. \qquad (83)$$

Остальные характеристики функционирования системы рассчитываются с использованием фундаментальных соотношений (1) – (3).

5.4.2. Расчёт характеристик систем с относительными приоритетами

Для ДО ОП в предположении, что приоритеты убывают с увеличением номера класса заявок, выражение для расчёта среднего времени ожидания может быть получено методом математической индукции на основе закона сохранения времени ожидания (61).

Пусть $H = 2$. В соответствии с законом сохранения (61):

$\rho_1 w_1^{\text{ОП}} + \rho_2 w_2^{\text{ОП}} = (\rho_1 + \rho_2) w^{\text{БП}}$.

После подстановки (83) в последнее выражение получим:

$$\rho_1 w_1^{\text{ОП}} + \rho_2 w_2^{\text{ОП}} = \frac{(\rho_1 + \rho_2) Pb}{K(1-\rho_1-\rho_2)}.$$

Положим, что $w_1^{\text{ОП}} = \frac{Pb}{K(1-\rho_1)}$ и подставим в предыдущее выражение. После некоторых преобразованиях найдем:

$$w_2^{\text{ОП}} = \frac{Pb}{K(1-\rho_1)(1-\rho_1-\rho_2)}.$$

По аналогии для $H = 3$ получим:

$$w_3^{\text{ОП}} = \frac{Pb}{K(1-\rho_1-\rho_2)(1-\rho_1-\rho_2-\rho_3)}.$$

В соответствии с методом математической индукции, положим, что

$$w_i^{\text{ОП}} = \frac{Pb}{K(1-R_{i-1})(1-R_i)},$$

где $R_{i-1} = \sum_{j=1}^{i-1} \rho_j$ и $R_i = \sum_{j=1}^{i} \rho_j$.

Положим теперь $H = i+1$. Выполнив аналогичные преобразования, получим, что

$$w_{i+1}^{\text{ОП}} = \frac{Pb}{K(1-R_i)(1-R_{i+1})}$$

Таким образом, среднее время ожидания заявок класса k определяется по формуле:

$$w_k^{\text{ОП}} = \frac{Pb}{K(1-R_{k-1})(1-R_k)} \qquad (k=1,\ldots,H), \tag{84}$$

где $R_n = \sum_{i=1}^{n} \rho_i$ – загрузка, создаваемая заявками классов $1,\ldots,n$ $(n=1,\ldots,H)$, причем: $R_0 = 0$; $R_H = R$, а P - вероятность того, что все K устройств заняты обслуживанием заявок, рассчитывается по формуле (81).

5.4.3. Расчёт характеристик систем с абсолютными приоритетами

Для ДО АП среднее время ожидания заявок класса k в системах с K параллельными устройствами было получено с использованием закона сохранения времени ожидания с учетом свойства ДО АП, заключающегося в том, что низкоприоритетные заявки не оказывают влияния на характеристики обслуживания заявок с более высоким АП:

$$w_k^{\text{АП}} = \left(\frac{P_k}{K(1-R_{k-1})(1-R_k)} + \frac{R_{k-1}(P_k - P_{k-1})}{K\rho_k(1-R_{k-1})} \right) b \qquad (k=1,\ldots,H), \tag{85}$$

где P_k – вероятность того, что все K устройств заняты обработкой заявок классов $1,\ldots,k$, т.е. заявок, приоритет которых не ниже k $(k=1,\ldots,H)$.

Вероятность P_k определяется аналогично (41):

$$P_k = \frac{(KR_k)^K}{K!(1-R_k)} P_{0,k},$$

где $P_{0,k}$ – вероятность того, что в системе нет заявок классов $1,\ldots,k$:

$$P_{0,k} = \left[\frac{(KR_k)^K}{K!(1-R_k)} + \sum_{i=0}^{K-1} \frac{(KR_k)^i}{i!} \right]^{-1} \qquad (k=1,\ldots,H).$$

5.4.4. Задачи проектирования систем с несколькими устройствами и неоднородной нагрузкой

Системы с несколькими устройствами и неоднородной нагрузкой представляются наиболее сложными базовыми моделями с приоритетным, в общем случае, обслуживанием заявок разных классов.

Перечень задач, решаемых в процессе проектирования таких систем, включает в себя все рассмотренные выше задачи, связанные как с определением структурных параметров системы (производительности и количества устройств, ёмкостей накопителей для заявок разных классов), так и функциональных параметров, задаваемых в виде дисциплин обслуживания заявок и способа назначения приоритетов. При этом возможны различные постановки задачи проектирования, отличающиеся составом ограничений, налагаемых на характеристики функционирования системы, и, как следствие, разными подходами и методами проектирования.

Проектирование систем с использованием сетевых моделей

В качестве моделей реальных систем с множеством взаимосвязанных устройств, обрабатывающих проходящие через них некоторые объекты, называемые заявками или запросами, используются сетевые модели в виде разомкнутых (РСеМО) и замкнутых (ЗСеМО) сетей массового обслуживания. Тип сетевой модели зависит от характера источника запросов, поступающих в исследуемую систему, и числа циркулирующих в системе запросов [1].

В том случае, когда запросы на обработку поступают независимо от состояния системы, т.е. от количества запросов уже находящихся в системе, обычно используются разомкнутые сетевые модели. Замкнутые сетевые модели используются при наличии в исследуемой системе зависимого источника запросов, например, при построении моделей систем, в которых новый запрос посылается в систему только после завершения обслуживания в системе какого-либо запроса.

Рассматриваемые ниже методы проектирования с использованием сетевых моделей базируются на аналитических методах расчёта линейных однородных разомкнутых и замкнутых сетей.

6.1. Проектирование систем на основе разомкнутых сетевых моделей

Положим, что некоторая система содержит множество обрабатывающих устройств. В систему в случайные моменты времени поступают объекты, которые, передвигаясь между устройствами в некотором порядке, обрабатываются в этих устройствах в течение случайного времени. В качестве модели такой системы с множеством взаимосвязанных устройств, обрабатывающих проходящие через них объекты, называемые запросами, могут использоваться сетевые модели в виде разомкнутых сетей массового обслуживания. Примерами таких систем могут служить информационно-вычислительные системы и сети, предназначенные для обработки поступающих запросов и выдачи требуемой информации, автоматизированные производственные и технологические системы обработки деталей и сборки технических изделий и т.п.

6.1.1. Описание разомкнутых сетевых моделей

Рассмотрим разомкнутую экспоненциальную сетевую модель с однородным потоком заявок при следующих предположениях и допущениях о моделируемой системе:

1) система содержит n групп устройств (узлов в модели) и может иметь произвольную топологию;

2) группы устройств системы могут содержать одно устройство (в модели представляются одноканальными узлами) или несколько однотипных устройств (в модели представляются многоканальными узлами);

3) все однотипные устройства (устройства многоканального узла) являются идентичными, и любая заявка может обрабатываться любым устройством;

4) заявка, поступившая в группу однотипных устройств (в многоканальный узел), когда все или несколько устройств свободны, направляется случайным образом в любое свободное устройство;

5) после завершения обработки в каком-либо узле (устройстве) передача заявки в другой узел происходит мгновенно;

6) длительности обработки заявок во всех устройствах системы представляют собой случайные величины, распределенные по экспоненциальному закону;

7) ёмкости накопителей перед каждой группой устройств (в узлах модели) не ограничены, что означает отсутствие отказов поступающим заявкам при их постановке в очередь, то есть любая поступающая в узел заявка всегда найдет в накопителе место для ожидания независимо от того, сколько заявок уже находится в очереди;

8) никакое устройство не простаивает, если в его накопителе имеется хотя бы одна заявка, причем после завершения обработки очередной заявки мгновенно из накопителя выбирается следующая заявка;

9) заявки поступают в сеть из внешнего независимого источника и образуют простейший поток заявок;

10) заявки из накопителя выбираются в соответствии с бесприоритетной дисциплиной обслуживания в порядке поступления (ОПП) по правилу «первым пришел – первым обслужен» (FIFO – First In First Out).

Для описания линейной разомкнутой однородной экспоненциальной сетевой модели необходимо задать следующую совокупность параметров:

- число узлов в сети: n;
- число однотипных устройств в узлах сети: $K_1,...,K_n$;
- матрицу вероятностей передач: $\mathbf{P} = [p_{ij} \mid i, j = 0,1,\ldots,n]$, где вероятности передач p_{ij} должны удовлетворять условию: сумма элементов каждой строки должна быть равна 1;
- интенсивность λ_0 источника заявок, поступающих в разомкнутую сетевую модель;
- средние длительности обслуживания заявок в узлах сетевой модели: $b_1,\ldots,b_n$.

На рисунке 28 представлена разомкнутая сетевая модель системы, содержащей $n = 4$ группы устройств (узлов в модели). Внешняя среда, откуда заявки поступают в систему и куда они возвращаются после обработки в системе, обозначена в модели как «0». Перемещение заявок в системе (модели) задано с помощью вероятностей передач p_{10}, p_{12}, p_{13}, причем $p_{10} + p_{12} + p_{13} = 1$.

Будем полагать, что в моделируемой системе отсутствуют перегрузки. Для этого необходимо, чтобы перегрузки отсутствовали в каждом из узлов сетевой модели [2], т.е. загрузка во всех узлах была меньше единицы:

$$\rho_j = \frac{\lambda_j b_j}{K_j} < 1 \qquad (j = \overline{1,n}). \tag{86}$$

Тогда на основе перечисленных параметров могут быть рассчитаны узловые и сетевые характеристики, описывающие эффективность функционирования соответственно узлов (устройств) и сети (системы) в целом.

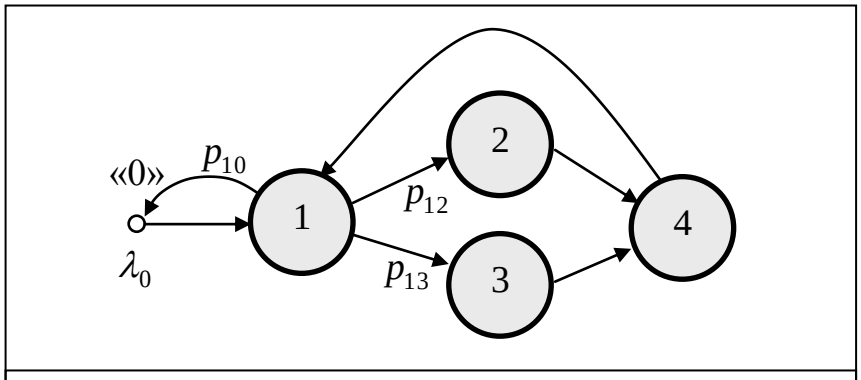

Рисунок 28. Граф разомкнутой сетевой модели

Расчёт характеристик функционирования линейной разомкнутой однородной экспоненциальной сетевой модели базируется на эквивалентном преобразовании сети и проводится в четыре этапа:

- расчёт коэффициентов передач α_j и интенсивностей потоков заявок λ_j в узлах $j=\overline{1,n}$ сети;
- проверка условия отсутствия перегрузок в сети;
- расчёт узловых характеристик;
- расчёт сетевых характеристик.

6.1.2. Расчёт коэффициентов передач и интенсивностей потоков заявок в узлах сетевой модели

Интенсивности $\lambda_0,\ldots,\lambda_n$ потоков заявок, поступающих в узлы $0,...,n$ сетевой модели, однозначно определяются через вероятности передач $p_{ij}\ (i,j=1,\ldots,n)$, задающие маршруты заявок в сети, путем решения системы линейных алгебраических уравнений [2]:

$$\lambda_j=\sum_{i=0}^{n}p_{ij}\lambda_i \quad (i=0,1,\ldots,n). \qquad (87)$$

Интенсивности потоков заявок в узлах разомкнутой сетевой модели могут быть найдены в виде соотношения $\lambda_j=\alpha_j\lambda_0\ (j=\overline{1,n})$, где α_j – коэффициент передачи, показывающий среднее число попаданий заявки в узел j за время ее нахождения в сети, причем $\alpha_0=1$.

Отметим, что коэффициенты передач $\alpha_j\ (j=\overline{1,n})$, так же как и вероятности передач $p_{ij}\ (i,j=1,\ldots,n)$, при заданной топологии сетевой модели полностью описывают последовательность прохождения заявками узлов сети, причем между ними существует взаимно-однозначное соответствие. Это означает, что по значениям вероятностей передач с использованием (87) могут быть рассчитаны коэффициенты передач, и, наоборот, по известным значениям

коэффициентов передач при необходимости можно рассчитать вероятности передач, если задана топология (конфигурация связей) сетевой модели

Во многих случаях при построении сетевой модели реальной системы известно среднее число попаданий заявки в процессе обслуживания в сети в тот или иной узел, т.е. известны коэффициенты передач α_j $(j=\overline{1,n})$.

6.1.3. Расчёт узловых характеристик разомкнутых моделей

Расчёт характеристик функционирования линейных разомкнутых однородных экспоненциальных сетевых моделей базируется на эквивалентном преобразовании, заключающемся в её представлении в виде n независимых экспоненциальных СМО (узлов) типа M/M/K с K устройствами, простейшим потоком заявок и экспоненциально распределенной длительностью их обработки. При этом интенсивность входящего потока заявок в узел j $(j=\overline{1,n})$ сети, определяется из системы алгебраических уравнений (87) через интенсивность входящего в сеть потока и коэффициент передачи узла: $\lambda_j=\alpha_j\lambda_0$, а средняя длительность обработки заявок в узле равна длительности обработки b_j заявок в соответствующем устройстве моделируемой системы: $b_j=\dfrac{\theta_j}{V_j}$, где θ_j – средняя ресурсоёмкость обработки запроса в устройстве j; V_j – производительность (быстродействие) устройства j $(j=\overline{1,n})$.

Характеристики всех n узлов (время ожидания заявок в очереди и пребывания в системе, длина очереди и число заявок в системе, среднее число занятых устройств и т.д.) представляют собой узловые характеристики сетевой модели.

Среднее время ожидания заявок в очереди может быть рассчитано с использованием выражения (40) для многоканальных СМО типа M/M/K или выражения (10**)** для одноканальных СМО типа M/M/1, остальные характеристики узла j $(j=\overline{1,n})$ – с использованием фундаментальных соотношений (1) – (3), а именно:

- нагрузка в узле j, показывающая среднее число занятых устройств: $y_j=\lambda_j b_j$;
- загрузка узла j: $\rho_j=\min(y_j/K_j;1)$, где K_j – число устройств в узле;
- коэффициент простоя узла: $\eta_j=1-\rho_j$;
- время пребывания заявок в узле: $u_j=w_j+b_j$;
- длина очереди заявок: $l_j=\lambda_j w_j$;
- число заявок в узле (в очереди и в устройстве): $m_j=\lambda_j u_j$.

Рассчитанные таким образом характеристики отдельных СМО в точности соответствуют узловым характеристикам сетевой модели, то есть в отношении своих характеристик модель в виде совокупности независимых СМО строго эквивалентна исходной разомкнутой сетевой модели.

6.1.4. Расчёт сетевых характеристик разомкнутых моделей

Сетевые характеристики, описывающие эффективность функционирования разомкнутой сетевой модели в целом, рассчитываются на основе полученных значений узловых характеристик.

В состав сетевых характеристик входят [2]:

- среднее число заявок, ожидающих обслуживания в сети, и среднее число заявок, находящихся в сети:

$$L = \sum_{j=1}^{n} l_j ; \qquad M = \sum_{j=1}^{n} m_j , \tag{88}$$

где l_j – средняя длина очереди и m_j – среднее число заявок в узле j;

- среднее время ожидания и среднее время пребывания заявок в сети:

$$W = \sum_{j=1}^{n} \alpha_j w_j ; \qquad U = \sum_{j=1}^{n} \alpha_j u_j , \tag{89}$$

где w_j и u_j – соответственно среднее время ожидания и среднее время пребывания заявок в узле j; α_j – коэффициент передачи для узла j $(j = \overline{1,n})$.

6.1.5. Способы ликвидации перегрузок в системе и определение допустимой нагрузки в сети

Допустимая нагрузка в сетевой модели, при отсутствии прочих требований, определяется из условия отсутствия в системе перегрузок (86), которое после некоторых преобразований с учетом $\lambda_j = \alpha_j \lambda_0$ может быть записано в следующем виде:

$$\lambda_0 < \frac{K_j}{\alpha_j b_j} \qquad (j = \overline{1,n})$$

или, что то же самое,

$$\lambda_0 < \min\left(\frac{K_1}{\alpha_1 b_1}, \frac{K_2}{\alpha_2 b_2}, ..., \frac{K_n}{\alpha_n b_n} \right). \tag{90}$$

Выражение (90) может использоваться для расчёта допустимой нагрузки в системе, задаваемой в виде интенсивности λ_0 внешнего источника заявок.

Узлы, загрузка которых при заданной интенсивности λ_0 близка к 1, называются высоконагруженными узлами. Для повышения эффективности функционирования системы необходимо разгрузить высоконагруженные узлы. Способы разгрузки узлов в системе, представляемой разомкнутой сетевой моделью, могут быть сформулированы на основе анализа выражения (90). Такими способами являются:

- уменьшение нагрузки системы (интенсивности λ_0) до значения, определяемого выражением (90), при котором это условие будет выполняться;
- увеличение количества устройств K_j в перегруженных узлах;

- уменьшение длительностей b_j обслуживания заявок в перегруженных узлах или, что то же самое, увеличение производительности V_j устройств с учетом того, что $b_j = \frac{\theta_j}{V_j}$;
- уменьшение коэффициентов передач α_j в перегруженных узлах, например путем перемещения файлов из перегруженного накопителя внешней памяти вычислительной системы в менее загруженный накопитель.

Высоконагруженными могут быть сразу несколько узлов, у которых отношения $\frac{K_j}{\alpha_j b_j}$ одинаковы. В последнем случае разгрузка узлов за счет увеличения количества устройств или за счет увеличения производительности (быстродействия) устройств только в одном узле не приведет к ликвидации перегрузки системы. Очевидно, что разгрузка должна быть выполнена одновременно во всех высоконагруженных узлах.

Рассмотрим подробнее перечисленные способы разгрузки узлов.

На рисунке 29 показана зависимость $U' = f'(\lambda_0)$ основной сетевой характеристики разомкнутой сетевой модели – среднего времени пребывания U заявок в сети от интенсивности λ_0 поступления заявок в сеть. Из рисунка видно, что имеется некоторое предельное значение интенсивности $\lambda'_{0\max}$, при котором среднее время пребывания заявок в сети становится бесконечно большим, что свидетельствует о перегрузке в исследуемой системе. Выше показано, что в разомкнутой сетевой модели отсутствуют перегрузки, если они отсутствуют во всех узлах сети, то есть перегрузка наступает только в том случае, если загрузка одного из узлов сети становится равной единице. Такой узел называется «узким местом» и характеризуется тем, что очередь заявок перед ним со временем растёт до бесконечности и, как следствие, становится бесконечным число заявок в разомкнутой сетевой модели.

Для того чтобы избавиться в разомкнутой сетевой модели от перегрузки, необходимо разгрузить «узкое место». Это может быть достигнуто, как сказано выше, несколькими способами, в частности:

- увеличением скорости работы (быстродействия) устройства;
- увеличением числа устройств в узле.

Любой из этих способов позволяет увеличить производительность моделируемой системы в целом и, как следствие, улучшить её характеристики. Зависимость среднего времени пребывания U заявок в сетевой модели от интенсивности λ_0 поступления заявок в сеть принимает вид $U'' = f''(\lambda_0)$, то есть время пребывания заявок при одной и той же интенсивности λ_0 становится меньше (поскольку сеть имеет большую производительность), а предельное значение интенсивности $\lambda''_{0\max}$, при котором наступает перегрузка

системы, становится больше: $\lambda''_{0\,max} > \lambda'_{0\,max}$. При этом появляется новое узкое место в системе, и дальнейшее улучшение системы может быть достигнуто путём разгрузки нового узкого места. Очевидно, что если разомкнутая сетевая модель является моделью реальной технической системы, разгрузка узкого места за счёт увеличения скорости работы устройства или числа устройств означает увеличение стоимости реальной системы.

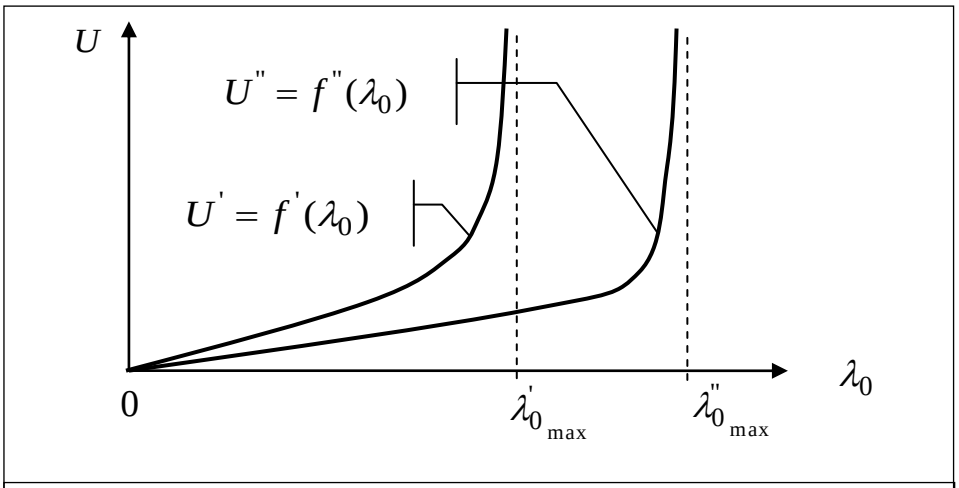

Рисунок 29. Зависимости среднего времени пребывания заявок в РСеМО от интенсивности их поступления

Ещё один способ разгрузки узкого места, заключающийся в уменьшении вероятности передачи заявок к узлу, являющемуся узким местом, часто используется в реальных системах и обычно не связан с увеличением стоимости системы. Например, в вычислительной системе изменение вероятностей передач к накопителям внешней памяти может быть достигнуто за счет перераспределения файлов между накопителями: наиболее часто используемые файлы, расположенные в наиболее загруженном накопителе, переносятся в наименее загруженный накопитель. При этом уменьшается количество обращений к загруженному накопителю (коэффициент передачи соответствующего узла разомкнутой сетевой модели).

Характер зависимостей других сетевых характеристик (времени ожидания, числа заявок в сети и в состоянии ожидания) разомкнутой сетевой модели от интенсивности поступления заявок аналогичен показанному на рисунке 29.

6.1.6. Задачи проектирования систем на основе разомкнутых сетевых моделей

Качество функционирования систем, представляемых разомкнутыми сетевыми моделями, определяется множеством показателей эффективности, в том числе, такими как пропускная способность и производительность системы, время нахождения запроса в системе, стоимость системы, надежность системы и т.д.

Положим, что специфика рассматриваемой системы такова, что в качестве основных показателей эффективности системы используются два показателя: время нахождения запроса в системе и стоимость системы, причем на один из этих показателей наложено ограничение, а второй показатель рассматривается в качестве критерия эффективности для решения задачи оптимального синтеза проектируемой системы.

Очевидно, что оба показателя зависят, прежде всего, от производительности (быстродействия) устройств. С одной стороны, для уменьшения времени нахождения запросов в системе желательно иметь устройства с высокой производительностью. С другой стороны, для уменьшения стоимости системы, производительности устройств должны быть небольшими, так как обычно стоимость отдельного устройства прямо пропорционально связана с его производительностью. Таким образом, можно ожидать, что существует некоторая «золотая середина», т.е. оптимальные производительности устройств, которые обеспечивают наилучшее соотношение рассматриваемых показателей эффективности, обеспечивающих экстремум критерия эффективности. При этом возможны три постановки задачи проектирования систем такого рода.

1. Формируется составной критерий эффективности аддитивного типа в виде:

$$F = k_1 U + k_2 S,$$

где U – среднее время нахождения запроса в системе; S – стоимость системы, складывающаяся из стоимости устройств, входящих в состав системы; k_1, k_2 – весовые коэффициенты, причём $k_1 + k_2 = 1$.

Требуется определить производительности устройств $V_1,...,V_n$ системы, при которых значение критерия эффективности примет минимальное значение: $F \to \min$.

Такую постановку задачи будем называть задачей проектирования системы на основе составного критерия эффективности.

2. Задано ограничение на стоимость системы: $S \le S^*$, а среднее время нахождения запроса в системе используется в качестве критерия эффективности: $F = U$.

Определить производительности устройств $V_1,...,V_n$ системы, при которых среднее время нахождения запроса в системе будет минимально: $U \to \min$.

Такую постановку задачи будем называть задачей проектирования системы заданной стоимости.

3. Задано ограничение на среднее время нахождения запроса в системе: $U \le U^*$, а стоимость системы используется в качестве критерия эффективности: $F = S$.

Определить производительности устройств $V_1,...,V_n$ системы, при которых стоимость системы будет минимальна: $S \to \min$.

Такую постановку задачи будем называть задачей проектирования системы с заданным временем пребывания запроса в системе.

Решения сформулированных задач проектирования базируются на аналитических методах расчёта характеристик разомкнутых сетевых моделей.

6.1.7. Проектирование системы на основе разомкнутой сетевой модели и составного критерия эффективности

Пусть система, состоящая из n устройств, предназначена для обработки однородного потока запросов, поступающих с интенсивностью λ_0. Запросы характеризуются определенными потребностями в ресурсах устройств $1,...,n$ (например, ресурсах оперативной и внешней памяти, каналов и устройств ввода-вывода, времени процессора), задаваемых в виде средних значений ресурсоёмкостей обработки $\theta_1,...,\theta_n$. Предположим, что в процессе обработки запрос в среднем α_i раз попадает в устройство $i=1,\ldots,n$. Стоимость системы складывается из стоимости всех устройств системы, причем стоимость устройства $i=1,\ldots,n$ зависит прямо пропорционально от производительности $V_1,...,V_n$ устройств:

$$S=\sum_{i=1}^{n} s_i=\sum_{i=1}^{n}\xi_i V_i\,, \qquad (91)$$

где ξ_i – стоимостной коэффициент пропорциональности.

Рассмотрим задачу проектирования системы, заключающуюся в определении производительностей устройств $V_1,...,V_n$, обеспечивающих минимум составного критерия эффективности:

$$F=k_1U+k_2S\,, \qquad (92)$$

где U – среднее время пребывания запроса в системе; S – стоимость системы; k_1, k_2 – весовые коэффициенты ($k_1+k_2=1$).

Представим описанную систему в виде разомкнутой сетевой модели, в которую поступает простейший поток заявок с интенсивностью λ_0. Длительность обработки заявки в узле $i=1,\ldots,n$, отображающем функционирование соответствующего устройства, распределена по экспоненциальному закону со средним значением $b_i=\theta_i/V_i \quad (i=1,\ldots,n)$.

Время пребывания заявки в системе может быть рассчитано на основе выражения (89):

$$U=\sum_{i=1}^{n}\alpha_i u_i=\sum_{i=1}^{n}\frac{\alpha_i\theta_i}{V_i-\lambda_i\theta_i}\,, \qquad (93)$$

где $u_i=\dfrac{\theta_i}{V_i-\lambda_i\theta_i}$ – среднее время пребывания заявки в узле $i=1,\ldots,n$, рассчитываемое по формуле (14) для каждого узла сетевой модели, рассматриваемого как СМО типа M/M/1.

Подставляя (91) и (93) в (92) , получим:

$$F = k_1 \sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i - \lambda_i \theta_i} + k_2 \sum_{i=1}^{n} \xi_i V_i \,.$$

Для нахождения минимума функции F возьмем частные производные от F по V_i и приравняем их нулю:

$$\frac{dF}{dV_i} = \frac{k_1 \alpha_i \theta_i}{(V_i - \lambda_i \theta_i)^2} + k_2 \xi_i = 0 \quad (i = 1, \ldots, n).$$

Решая полученную систему независимых уравнений, определим оптимальные значения производительностей устройств системы:

$$V_{i_{\text{opt}}} = \lambda_i \theta_i + \sqrt{\frac{k_1 \alpha_i \theta_i}{k_2 \xi_i}} \quad (i = 1, \ldots, n), \tag{94}$$

Первое слагаемое в полученном выражении представляет собой нижнюю границу производительности устройства $i = 1, \ldots, n$, обеспечивающую отсутствие перегрузок, а второе слагаемое – это некоторая дополнительная производительность, оптимизирующая решение.

Формула (94) с точностью до обозначений совпадает с (18) с единственным отличием, заключающемся в учёте количества попаданий заявки α_i в узел $i = 1, \ldots, n$.

Подставляя (94) в (93) и (91) , определим среднее время задержки запросов в системе и стоимость системы:

$$U = \sqrt{\frac{k_2}{k_1}} \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} \,;$$

$$S = \sum_{i=1}^{n} \xi_i \lambda_i \theta_i + \sqrt{\frac{k_1}{k_2}} \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} \,.$$

В последнем выражении первое слагаемое в правой части представляет собой стоимость системы минимальной конфигурации, которая определяется значениями нижней границы производительности $V_{0_i} = \lambda_i \theta_i$ для всех узлов $i = 1, \ldots, n$ модели (устройств системы), а второе слагаемое – дополнительные затраты, обеспечивающие построение оптимальной системы в соответствии с заданным критерием эффективности (92).

Если показатели эффективности системы U и S – равноценны, то есть весовые коэффициенты $k_1 = k_2$, то выражение (94) примет вид:

$$V_{i_{\text{opt}}} = \lambda_i \theta_i + \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \quad (i = 1, \ldots, n),$$

а среднее время задержки запросов в системе и стоимость системы будут равны:

$$U = \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} \,;$$

$$S = \sum_{i=1}^{n} \xi_i \lambda_i \theta_i + \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} \,.$$

6.1.8. Проектирование системы заданной стоимости на основе разомкнутой сетевой модели

Пусть, как и в предыдущем пункте, система, состоящая из n устройств и обрабатывающая однородные запросы, поступающие с интенсивностью λ_0, представляется в виде разомкнутой сетевой модели. Средние ресурсоёмкости обработки запросов в узлах модели, отображающих функционирование устройств системы, равны $\theta_1,...,\theta_n$. Запросы в среднем α_i раз попадают в устройство $i = 1,\ldots,n$ за время нахождения в системе (сети), следовательно, интенсивность поступления запросов к устройству $\lambda_i = \alpha_i \lambda_0 \quad (i = \overline{1,n})$.

Положим, что на стоимость системы, определяемую выражением (91), наложено ограничение:

$$S \le S^*. \tag{95}$$

Задача проектирования системы формулируется следующим образом: определить производительности $V_1,...,V_n$ устройств, обеспечивающие минимальное время пребывания запросов в системе (в сетевой модели) $U - \min$ при условии, что стоимость системы не превышает заданного значения S^*.

Стоимость системы S и время пребывания запросов в системе U определяются выражениями (91) и (93).

Для решения задачи проектирования системы заданной стоимости воспользуемся методом неопределённых множителей Лагранжа.

Функция Лагранжа будет иметь вид:

$$G = U + \gamma(S - S^*),$$

где γ – неопределённый множитель Лагранжа.

Для того чтобы найти минимум функции Лагранжа G, подставим (91) и (93) в последнее выражение с учетом $\lambda_i = \alpha_i \lambda_0$ и возьмем частные производные от G по V_i:

$$\frac{\mathrm{d}G}{\mathrm{d}V_i} = -\frac{\alpha_i \theta_i}{(V_i - \alpha_i \lambda_0 \theta_i)^2} + \gamma \xi_i \qquad (i = \overline{1,n}).$$

Приравняв последнее выражение к нулю, решим полученную систему уравнений и найдем V_i:

$$V_i = \alpha_i \lambda_0 \theta_i + \frac{1}{\sqrt{\gamma}} \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \quad (i = \overline{1,n}). \tag{96}$$

Полагая, что решение задачи ищется на границе ограничения (95) определим неопределённый множитель Лагранжа из условия:

$$S = S^* \quad \text{или} \quad \sum_{i=1}^{n} \xi_i V_i = S^*.$$

Подставив (96) вместо V_i в последнее выражение, после некоторых преобразований получим:

$$\frac{1}{\sqrt{\gamma}} = \frac{S^* - \sum_{i=1}^{n} \xi_i \alpha_i \lambda_0 \theta_i}{\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}}.$$

Подставляя последнее выражение в (96), получим окончательное решение в виде оптимальных значений производительности устройств проектируемой системы:

$$V_{i_{\text{opt}}} = \alpha_i \lambda_0 \theta_i + \frac{S^* - \sum_{i=1}^{n} \xi_i \alpha_i \lambda_0 \theta_i}{\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}} \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \qquad (i = \overline{1,n}). \tag{97}$$

Выполним анализ полученного результата.

Оптимальная производительность i-го устройства складывается из двух составляющих:

- минимально необходимой производительности $V_{0_i} = \alpha_i \lambda_0 \theta_i$, представляющей собой нижнюю границу производительности, начиная с которой обеспечивается отсутствие перегрузок в устройстве, а, следовательно, и в системе;
- дополнительной производительности, распределяемой между всеми устройствами.

Дополнительная производительность формируется исходя из остаточной стоимости системы, представляющей собой разность между заданной стоимостью S^* и затратами $S_{\min} = \sum_{i=1}^{n} \xi_i \alpha_i \lambda_0 \theta_i$, необходимыми для построения системы минимальной конфигурации, в которой отсутствуют перегрузки: $\Delta S = S^* - S_{\min}$. Эта остаточная стоимость распределяется между всеми устройствами системы пропорционально $\sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \quad (i = \overline{1,n})$.

Таким образом, минимум среднего времени пребывания запросов в системе стоимостью S^* достигается при распределении производительностей $V_1,...,V_n$ устройств в соответствии с (97). Такое распределение называется оптимальным для системы заданной стоимости.

Среднее время ответа системы стоимостью S^* будет равно

$$U = \frac{\left(\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \right)^2}{S^* - \lambda_0 \sum_{i=1}^{n} \xi_i \alpha_i \theta_i}.$$

6.1.9. Проектирование системы с заданным временем пребывания на основе разомкнутой сетевой модели

Положим теперь, что ограничение налагается на среднее время пребывания запросов в системе:

$$U \le U^*. \qquad (98)$$

Задача проектирования системы формулируется следующим образом: определить производительности $V_1,...,V_n$ устройств, обеспечивающие минимальную стоимость системы $S - \min$ при условии, что среднее время пребывания запросов в системе (в сетевой модели) не превышает заданного значения U^*.

Стоимость системы S и время пребывания запросов в системе U определяются выражениями (91) и (93).

Для решения задачи проектирования системы заданной стоимости воспользуемся методом неопределённых множителей Лагранжа.

Функция Лагранжа будет иметь вид:

$$G = S + \gamma(U - U^*),$$

где γ – неопределённый множитель Лагранжа.

Для того чтобы найти минимум функции Лагранжа G, подставим (91) и (93) в последнее выражение с учетом $\lambda_i = \alpha_i \lambda_0 \quad (i = \overline{1, n})$ и возьмем частные производные от G по V_i $(i = \overline{1,n})$:

$$\frac{\mathrm{d}G}{\mathrm{d}V_i} = \xi_i - \frac{\gamma \alpha_i \theta_i}{(V_i - \alpha_i \lambda_0 \theta_i)^2} \qquad (i = \overline{1,n}).$$

Приравняв последнее выражение нулю, решим полученное уравнение и найдем V_i $(i = \overline{1,n})$:

$$V_i = \alpha_i \lambda_0 \theta_i + \sqrt{\gamma}\sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \qquad (i = \overline{1,n}). \qquad (99)$$

Полагая, что решение задачи ищется на границе ограничения (95) определим неопределённый множитель Лагранжа из условия:

$$U = U^* \quad \text{или} \quad \sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i - \alpha_i \lambda_0 \theta_i} = U^*$$

Подставив (96) вместо V_i в последнее выражение, после некоторых преобразований получим:

$$\sqrt{\gamma} = \frac{\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}}{U^*}.$$

Подставляя последнее выражение в (96), получим окончательное решение в виде оптимальных значений производительности устройств проектируемой системы:

$$V_{i_{\text{opt}}} = \alpha_i \lambda_0 \theta_i + \frac{1}{U^*} \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \qquad (i = \overline{1,n}). \tag{100}$$

Как и в предыдущем случае оптимальная производительность i-го устройства складывается из двух составляющих: минимально необходимой производительности $V_{0_i} = \alpha_i \lambda_0 \theta_i$, представляющей собой нижнюю границу производительности, начиная с которой обеспечивается отсутствие перегрузок в системе, и дополнительной производительности, при которой выполняется ограничение на среднее время пребывания запросов в системе.

Дополнительная производительность тем больше, чем меньше ограничение U^*, причем она распределяется между всеми устройствами системы пропорционально $\sqrt{\frac{\alpha_i \theta_i}{\xi_i}}$ $(i = \overline{1,n})$.

Таким образом, минимум стоимости системы при ограничении на среднее время пребывания запросов в системе U^* достигается при распределении производительностей $V_1,...,V_n$ устройств в соответствии с формулой (100). Такое распределение является оптимальным для системы с заданным временем пребывания запросов в системе, при этом стоимость системы с заданным временем пребывания U^* будет равна

$$S = \sum_{i=1}^{n} \xi_i \alpha_i \lambda_0 \theta_i + \frac{1}{U^*} \left(\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \right)^2 .$$

Стоимость системы, определяемая этим выражением, минимальна при рассматриваемой постановке задачи синтеза.

6.2. Проектирование систем на основе замкнутых сетевых моделей

Положим, что проектируемая система содержит множество взаимосвязанных обрабатывающих устройств, между которыми перемещаются объекты, называемые запросами и обрабатываемые в этих устройствах в течение случайного времени. Перемещение объектов в общем случае представляет собой случайный процесс, описываемый в виде вероятностей передач между устройствами. Если число запросов, циркулирующих в системе, постоянно и не меняется в течение длительного времени, в качестве модели такой системы используются замкнутые сети массового обслуживания. Эти же модели могут использоваться, например, в том случае, если в момент завершения обработки в системе некоторого запроса в систему сразу же посылается новый запрос. Таким образом, количество обрабатываемых в системе запросов (запросов, находящихся одновременно в системе) остается постоянным.

Примерами таких систем могут служить многотерминальные информационно-вычислительные системы, такие как системы резервирования билетов, справочные и библиотечные системы. Отличительная особенность таких систем состоит в том, что с каждого терминала в систему может быть послан новый запрос только после того, как получен ответ на предыдущий

запрос. С учетом этого и включая терминалы в состав модели, можно констатировать, что в системе (в модели) циркулирует одно и то же число запросов, равное количеству терминалов в исследуемой системе [1].

6.2.1. Описание замкнутых сетевых моделей

Рассмотрим замкнутую экспоненциальную сетевую модель с однородным потоком заявок при следующих предположениях и допущениях о моделируемой системе:

1) система содержит n групп устройств (узлов в модели) и может иметь произвольную топологию;

2) все группы содержат только по одному устройству (все узлы замкнутой модели одноканальные);

3) после завершения обработки в каком-либо устройстве (узле) передача заявки в другой узел происходит мгновенно;

4) в системе циркулирует постоянное число заявок;

5) длительности обработки заявок во всех узлах сети представляют собой случайные величины, распределенные по экспоненциальному закону;

6) ёмкость накопителя перед каждым устройством (узлом модели) достаточна для хранения всех заявок, циркулирующих в системе, что означает отсутствие отказов поступающим заявкам при их занесении в накопитель любого узла (в частности, можно считать, что ёмкость накопителя в каждом узле равна числу заявок, циркулирующих в сети);

7) никакое устройство не простаивает, если в его накопителе имеется хотя бы одна заявка, причем после завершения обработки очередной заявки мгновенно из накопителя выбирается следующая заявка;

8) заявки из накопителей выбираются в соответствии с бесприоритетной дисциплиной обслуживания в порядке поступления (ОПП) по правилу «первым пришел – первым обслужен» (FIFO – First In First Out).

Для описания линейных замкнутых однородных экспоненциальных моделей необходимо задать такую же совокупность параметров, как и для разомкнутых моделей, с единственным отличием, заключающимся в том, что вместо интенсивности источника заявок следует задать число заявок, циркулирующих в замкнутой модели. Таким образом, совокупность параметров для замкнутых моделей будет иметь следующий вид:

- число узлов в сети: n;
- число устройств в узлах сети: $K_1 = \ldots = K_n = 1$;
- матрица вероятностей передач: $\mathbf{P} = [p_{ij} \mid i, j = 0, 1, \ldots, n]$, где p_{ij} – вероятность передачи заявки из узла i в узел j;
- число заявок M, циркулирующих в замкнутой модели;
- средние длительности обслуживания заявок в узлах сети: $b_1, \ldots, b_n$.

На основе перечисленных параметров могут быть рассчитаны узловые и сетевые характеристики, описывающие эффективность функционирования соответственно узлов и замкнутой модели в целом.

Для замкнутых сетевых моделей, как и для разомкнутых, наибольший интерес представляют характеристики, описывающие эффективность функционирования проектируемой системы в целом, основными среди которых являются:

- производительность λ_0 системы (замкнутой сетевой модели), которую будем называть также *системной (сетевой) производительностью* в отличие от производительностей устройств (узлов) $V_1,...,V_n$;
- среднее время пребывания U запросов в системе, измеряемое между двумя последовательными моментами прохождения заявки через выбранную на одной из дуг сетевой модели нулевой точки «0».

На рисунке 30 представлен граф замкнутой сетевой модели системы, содержащей $n = 4$ группы однотипных устройств (узлов в модели), аналогичный разомкнутой модели (рисунок 28). Особенность этой модели заключается в отсутствии внешней среды. В представленной модели дуга, выходящая из узла 1 и входящая обратно в этот же узел, в реальной системе может соответствовать ситуации, когда новая заявка поступает в систему только в момент выхода из сети обработанной заявки.

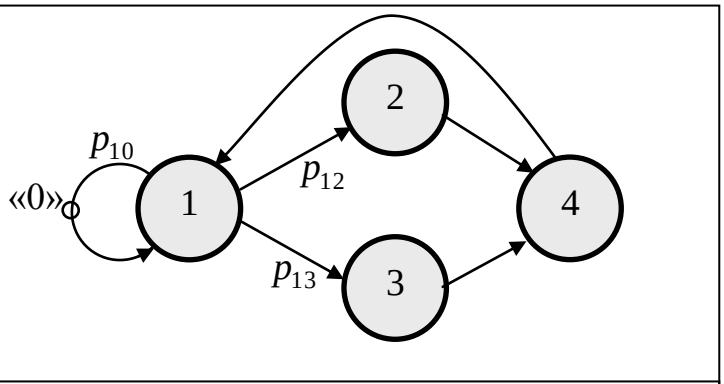

Рисунок 30. Граф замкнутой сетевой модели

Расчёт характеристик функционирования линейных замкнутых однородных экспоненциальных сетевых моделей с одноканальными узлами базируется на так называемой «теореме о прибытии» и проводится с использованием метода средних значений в два этапа:

- расчёт коэффициентов передач в узлах замкнутой сетевой модели;
- расчёт характеристик замкнутой сетевой модели.

Заметим, что в отличие от разомкнутых сетевых моделей при расчёте характеристик функционирования замкнутых моделей отсутствует этап проверки условия отсутствия перегрузок в сети. Это обусловлено тем, что в замкнутых моделях (системах) по определению невозможны перегрузки, поскольку в сети циркулирует постоянное число заявок.

6.2.2. Расчёт коэффициентов передач в узлах замкнутой модели

Для замкнутой модели на первом этапе рассчитываются только коэффициенты передач. Интенсивности потоков заявок в узлах замкнутой модели не могут быть рассчитаны, как в разомкнутой модели, поскольку для

замкнутой модели изначально не известна интенсивность λ_0, которая является не параметром, задаваемым в составе исходных данных, а характеристикой, представляющей собой производительность замкнутой сетевой модели и определяемой в процессе анализа эффективности функционирования замкнутой модели. Производительность замкнутой сетевой модели λ_0 определяет производительность проектируемой системы (системную производительность) и измеряется количеством запросов, обработанных системой за единицу времени, например, количеством задач, выполненных вычислительной системой за единицу времени, или количеством пакетов, переданных в компьютерной сети за единицу времени. Системную производительность не следует путать с производительностью узлов сетевой модели (устройств системы) $V_1,...,V_n$. Ясно, что производительность системы, представляемой замкнутой сетевой моделью, зависит от производительности устройств: $\lambda_0 = f(V_1,...,V_n)$, причем с увеличением производительности устройств будет расти системная производительность.

Для расчёта коэффициентов передач $\alpha_1,\ldots,\alpha_n$ после некоторых преобразований можно воспользоваться той же системой линейных алгебраических уравнений (87). Для этого в левой и правой части выражения (87) представим интенсивности в виде $\lambda_j = \alpha_j \lambda_0$. Разделив левую и правую часть выражения (87) на λ_0, окончательно получим систему линейных алгебраических уравнений относительно $\alpha_1,\ldots,\alpha_n$:

$$\alpha_j = \sum_{i=0}^{n} p_{ij}\alpha_i \quad (i = 0,1,\ldots,n). \qquad (101)$$

Полагая $\alpha_0 = 1$, можно найти корни системы уравнений, численно определяющие значения. $\alpha_1,\ldots,\alpha_n$

6.2.3. Расчёт характеристик замкнутой сетевой модели

Характеристики замкнутой сетевой модели могут быть рассчитаны с использованием *метода средних значений*, позволяющего вычислять средние характеристики функционирования экспоненциальных замкнутых сетевых моделей на основе сравнительно простых рекуррентных соотношений.

Положим, что однородная замкнутая сетевая модель содержит n одноканальных узлов, длительности обслуживания заявок в которых распределены по экспоненциальному закону со средними значениями $b_1,\ldots,b_n$ соответственно. Пусть для каждого узла i сети известно среднее число попаданий заявки в данный узел за время ее нахождения в сети, то есть коэффициент передачи α_i, который, если конфигурация сети задана матрицей вероятностей передач $P = [p_{ij} \,|\, i, j = 0,1,\ldots,n]$, определяется в результате решения системы линейных алгебраических уравнений (101).

Обозначим: u_i – среднее время пребывания заявки в узле i за время пребывания в сети; m_i – среднее число заявок в узле $i\,(i = 1,\ldots,n)$; λ_0 –

производительность замкнутой сети. Очевидно, что эти величины зависят от числа заявок M, циркулирующих в замкнутой сети, то есть $u_i = u_i(M)$; $m_i = m_i(M)$; $\lambda_0 = \lambda_0(M)$.

Можно показать, что имеют место следующие соотношения:

$$u_i(M) = b_i[1 + m_i(M-1)]; \quad (102)$$

$$U(M) = \sum_{i=1}^{n} \alpha_i u_i(M); \quad (103)$$

$$\lambda_0(M) = \frac{M}{U(M)}; \quad (104)$$

$$m_i(M) = \alpha_i \lambda_0(M) u_i(M), \quad (105)$$

где $U(M)$ – среднее время пребывания заявок в сети при условии нахождения в ней M заявок; $m_i(0) = 0$.

Выражение (102) получено на основе *теоремы о прибытии*, утверждающей, что в замкнутой экспоненциальной сети с одноканальными узлами, в которой циркулируют M заявок, стационарная вероятность состояния любого узла в момент поступления в него новой заявки совпадает со стационарной вероятностью того же состояния рассматриваемого узла в сети, в которой циркулирует на одну заявку меньше, то есть $(M-1)$ заявок. Это означает, что в сети с M заявками среднее число заявок $m_i(M)$, находящихся в узле i в момент поступления в этот узел новой заявки, равно $m_i(M-1)$. Тогда среднее время пребывания в узле i поступившей заявки будет складываться из среднего времени обслуживания всех $m_i(M-1)$ ранее поступивших и находящихся в узле i заявок и средней длительности обслуживания рассматриваемой заявки:

$$u_i(M) = b_i m_i(M-1) + b_i = b_i[1 + m_i(M-1)].$$

В этом выражении учтено, что среднее время дообслуживания заявки, находящейся в устройстве на момент поступления рассматриваемой заявки, равно средней длительности обслуживания b_i в силу свойства отсутствия последействия, присущего экспоненциальному закону. Среднее время пребывания заявки в узле i за время ее нахождения в сети, учитывающее число попаданий α_i заявки в данный узел, равно $U_i(M) = \alpha_i u_i(M)$.

Выражения (103) и (104) представляют собой формулы Литтла для сети, а выражение (105) – для узла i, где $\lambda_i(M) = \alpha_i \lambda_0(M)$ – интенсивность потока заявок в узел i $(i = 1, \ldots, n)$.

На основе рекуррентных соотношений (102) – (105) последовательно для $M = 1, 2, \ldots, M^*$, где M^* – заданное число заявок в замкнутой сети, могут быть рассчитаны средние значения характеристик замкнутой экспоненциальной сетевой модели.

Отметим ещё раз, что приведенный метод расчёта является точным для замкнутых экспоненциальных сетевых моделей с *одноканальными* узлами.

6.2.4. Рекомендации по проектированию систем на основе замкнутых сетевых моделей

Для качественного проектирования систем, моделями которых служат замкнутые сетевые модели, и формирования рекомендаций по построению эффективных систем важным этапом является выявление и изучение их свойств на основе этих моделей. При этом наибольший интерес представляют свойства системы, характеризующие влияние циркулирующих в замкнутой сетевой модели числа заявок, на такие сетевые характеристики как системная производительность λ_0 и среднее время пребывания U заявок в сети.

На рисунке 31 представлены зависимости производительности $\lambda_0 = f(M)$ и времени пребывания заявок в сети $U = f(M)$ от числа заявок M, циркулирующих в замкнутой сетевой модели.

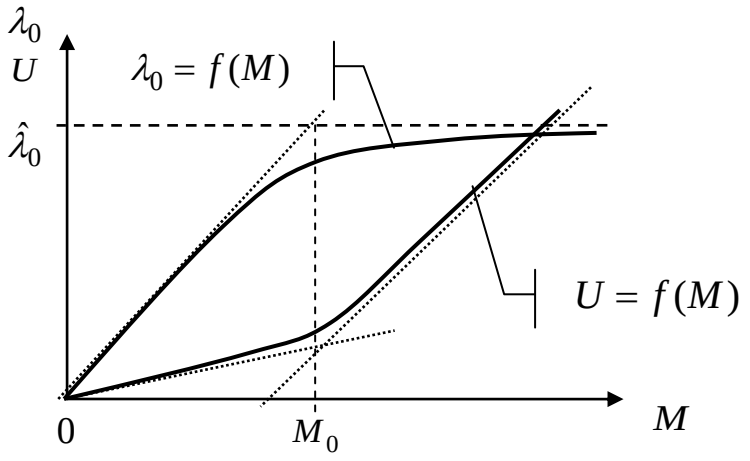

Рисунок 31. Зависимости производительности и времени пребывания заявок от их числа в ЗСеМО

Анализ представленных зависимостей позволяет сформулировать следующие выводы.

1. При увеличении числа M циркулирующих в сети заявок системная производительность λ_0 вначале растёт до некоторого значения M_0, после которого рост производительности замедляется, а с дальнейшим увеличением M производительность сети асимптотически стремится к некоторому предельному значению $\hat{\lambda}_0$, представляющему собой пропускную способность замкнутой сети. Для объяснения этой зависимости вспомним, что производительность замкнутой сети измеряется как интенсивность потока заявок, проходящих через некоторую условную точку, обозначаемую как «0» и расположенную на одной из дуг замкнутой сетевой модели (рисунок 30), отображающей завершение обслуживания заявок в сети и мгновенное формирование новой заявки, поступающей в сеть. Очевидно, что увеличение числа заявок в замкнутой сети приводит к увеличению значений всех сетевых характеристик, включая производительность λ_0. В свою очередь, увеличение

производительности приводит к увеличению загрузок узлов сети, связанных с интенсивностью λ_0 зависимостью:

$$\rho_j = \frac{\alpha_j \lambda_0 b_j}{K_j},$$

где α_j, b_j и K_j – соответственно коэффициент передачи, средняя длительность обслуживания и количество устройств в узле $j = \overline{1,n}$.

Когда число заявок в сети достигает некоторого значения M_0, загрузка одного из узлов становится близкой к 1, при этом практически прекращается рост производительности системы, которая при $M \to \infty$ достигает своего предельного значения – пропускной способности $\hat{\lambda}_0$. Такой узел представляет собой «узкое место» сети, и значение пропускной способности $\hat{\lambda}_0$ определяется пропускной способностью узкого места из условия, что загрузка ρ_y этого узла равна 1:

$$\rho_y = \frac{\alpha_y \lambda_0 b_y}{K_y} = 1.$$

Отсюда пропускная способность замкнутой сети:

$$\hat{\lambda}_0 = \frac{K_y}{\alpha_y b_y},$$

где α_y, b_y и K_y – соответственно коэффициент передачи, средняя длительность обслуживания и количество устройств в узле, являющимся узким местом.

Правая часть последнего выражения представляет собой пропускную способность узла, являющегося узким местом сети: $\mu_y = \frac{K_y}{\alpha_y b_y}$. Действительно, $\alpha_y b_y$ представляет собой полное время обслуживания одной заявки в данном узле с учётом того, что заявка за время нахождения в сети в среднем α_y раз побывает в данном узле. Тогда величина, обратная $\alpha_y b_y$, представляет собой интенсивность обслуживания заявок одним устройством в данном узле: $\mu_1 = 1/\alpha_y b_y$, а $\mu_y = K_y \mu_1$ – интенсивность обслуживания заявок узлом, то есть всеми устройствами.

Этот же результат можно получить следующими рассуждениями. Если загрузка некоторого узла, являющегося узким местом сети, становится равной 1, то это означает, что все устройства данного узла постоянно обслуживают заявки, то есть не простаивают. Тогда интенсивность выходящего из этого узла потока заявок будет равна интенсивности обслуживания: $\lambda_y = \mu_y = K_y \mu_1$. Напомним, что интенсивность потока заявок в узле λ_y связана с производительностью замкнутой сети λ_0 зависимостью $\lambda_y = \alpha_y \lambda_0$. Отсюда вытекает, что производительность замкнутой сети равна:

$$\lambda_0 = \frac{\lambda_y}{\alpha_y} = \frac{K_y \mu_1}{\alpha_y} = \frac{K_y}{\alpha_y b_y}.$$

2. Среднее время пребывания заявок (рисунок 31) в замкнутой сетевой модели, как и производительность, растёт с увеличением числа циркулирующих в сети заявок M, причём вначале наблюдается незначительный рост, а затем, после значения $M = M_0$, наблюдается линейный рост времени пребывания.

Действительно, если в сети циркулирует только одна заявка, то в такой сети не может быть очередей, и время пребывания заявок в сети складывается только из времён обслуживания заявок в узлах с учётом коэффициентов передач:

$$U = \sum_{i=1}^{n} \alpha_i b_i .$$

С увеличением числа заявок M в узлах замкнутой сети появляются очереди, причём очевидно, что чем больше заявок в сети, тем более длинные очереди образуются в узлах и тем больше время ожидания, а, следовательно, и время пребывания заявок в замкнутой сети.

Сопоставляя зависимости производительности и среднего времени пребывания заявок от их числа в замкнутой сети, можно сделать следующий вывод: увеличение числа заявок в сети, с одной стороны, приводит к увеличению производительности, что может рассматриваться как положительный фактор, а, с другой стороны, – к увеличению времени пребывания заявок в сети, что является нежелательным фактором.

Точка $M = M_0$ характеризует некоторое граничное значение числа заявок в замкнутой сети. Дальнейшее увеличение числа заявок в сети оказывается нецелесообразным, поскольку приводит к резкому увеличению времени пребывания заявок в замкнутой сети при незначительном увеличении производительности сети.

3. Когда загрузка узкого места становится равной единице, дальнейший рост производительности за счёт увеличения числа заявок в замкнутой сети невозможен. Для увеличения производительности замкнутой сети, как и в разомкнутой сети, необходимо разгрузить узкое место, то есть уменьшить загрузку: $\rho_y = \frac{\alpha_y \lambda_0 b_y}{K_y} = 1$, что может быть достигнуто:

- уменьшением длительности обработки заявок b_y, например за счет увеличения производительности (быстродействия) устройства;
- увеличением числа устройств K_y в узле;
- уменьшением коэффициента передачи α_y или, что то же самое, вероятности передачи заявок к узлу, являющемуся узким местом.

Если до разгрузки узкого места зависимость производительности замкнутой сети от числа заявок в сети имела вид $\lambda_0^{'} = f(M)$ (рисунок 32), а

пропускная способность была равна $\hat{\lambda}_0'$, то после разгрузки – зависимость производительности от числа заявок будет иметь вид $\lambda_0'' = f(M)$, а пропускная способность станет равной $\hat{\lambda}_0'' > \hat{\lambda}_0'$. При этом граничное значение числа заявок в замкнутой сети увеличится: $M_0'' > M_0'$.

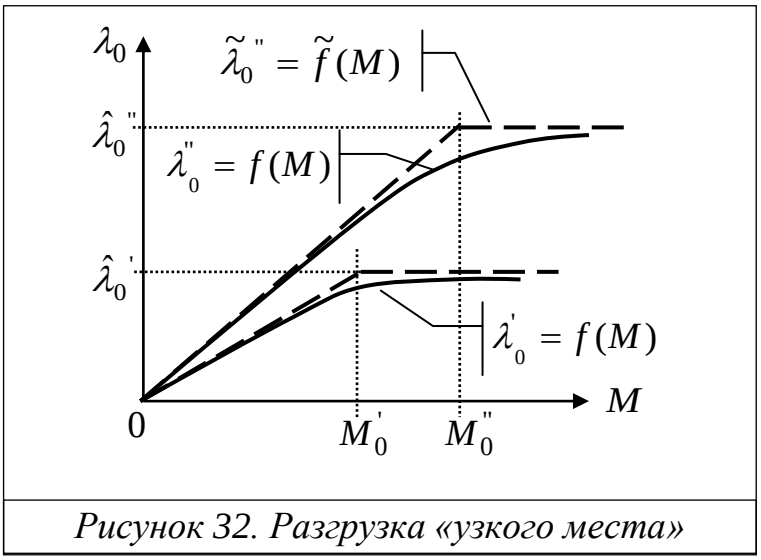

Рисунок 32. Разгрузка «узкого места»

Следует отметить, что к рассматриваемой зависимости производительности замкнутой сети λ_0 от числа M циркулирующих в сети заявок может быть применена линейная аппроксимация $\tilde{\lambda}_0'' = \tilde{f}(M)$, показанная на рисунке 32 в виде пунктирных линий и представляющая собой верхнюю границу производительности замкнутой сети. Последнее означает, что производительность замкнутой сети будет не больше, чем рассчитанное верхнее значение.

Нетрудно представить себе и изобразить на графике, как изменится зависимость среднего времени пребывания заявок в замкнутой сети от числа циркулирующих в сети заявок после разгрузки узкого места.

Отметим, что в некоторых случаях разгрузка узкого места не приводит к улучшению характеристик системы, в частности, к увеличению производительности. Обычно это связано с тем, что в системе может существовать несколько узлов, являющихся «узкими местами». Условием этого является равенство загрузок узлов: $\rho_i = \rho_j$ или $\frac{\alpha_i \lambda_0 b_i}{K_i} = \frac{\alpha_j \lambda_0 b_j}{K_j}$, откуда окончательно получим: $\frac{\alpha_i b_i}{K_i} = \frac{\alpha_j b_j}{K_j}$ $(i \neq j)$. В этом случае для улучшения характеристик системы необходимо одновременно разгрузить все «узкие места».

Последовательно разгружая узкие места проектируемой системы, мы можем прийти к некоторому «идеальному» варианту системы, в которой загрузки всех устройств одинаковы.

Система, в которой загрузки всех узлов равны, называется *сбалансированной*. Сбалансированная система обладает наилучшими характеристиками по сравнению с несбалансированной.

При построении реальных систем, моделями которых служат разомкнутые и замкнутые сети, необходимо, по-возможности, строить сбалансированные системы, хотя на практике по многим причинам достичь этого не удаётся.

6.2.5. Задачи проектирования систем на основе замкнутых сетевых моделей

Проектирование на основе замкнутых сетевых моделей при заданном количестве устройств n может быть направлено на решение следующих задач:

- определение производительностей (быстродействий) устройств $V_1,...,V_n$ системы;
- определение числа запросов M, одновременно находящихся в системе.

Основными показателями эффективности функционирования систем, представляемых замкнутыми сетевыми моделями, являются:

- системная производительность – количество запросов, обрабатываемых системой за единицу времени λ_0;
- время нахождения запроса в системе – от момента поступления в систему до момента завершения обработки в системе U;
- стоимость системы S.

Объединяя эти показатели в единый обобщенный критерий эффективности или выбирая один из этих показателей в качестве критерия эффективности и налагая ограничение на остальные показатели, можно сформулировать несколько постановок задач оптимального синтеза проектируемой системы.

Следует помнить, что в системах, представляемых замкнутыми сетевыми моделями, среднее время пребывания запроса в системе U и системная производительность λ_0 связаны зависимостью (104):

$$\lambda_0 = \frac{M}{U},$$

где M - число запросов, одновременно находящихся в системе.

Отсюда следует, что при заданном значении M уменьшение времени пребывания запроса в системе U равносильно увеличению системная производительности λ_0, следовательно, при решении задачи определения производительности устройств можно использовать только один из этих показателей. Но при определении числа запросов M следует учитывать оба показателя U и λ_0, так как с увеличением M увеличиваются U и λ_0. (см. рисунок 31)

Как и в случае разомкнутых сетевых моделей можно сформулировать, по крайней мере, три постановки задачи проектирования, связанные с определением производительностей устройств системы.

1. Определить производительности устройств $V_1,...,V_n$ системы, при которых значение критерия эффективности:

$$F_1 = k_1 U + k_2 S \quad \text{или} \quad F_2 = k_1 \frac{M}{\lambda_0} + k_2 S,$$

где k_1, k_2 – весовые коэффициенты, примет минимальное значение: $F - \min$.

С учетом (104) можно констатировать, что критерии эффективности F_1 и F_2 эквивалентны.

Вместо аддитивного критерия эффективности может использоваться мультипликативный критерий вида: $F_3 = \frac{S}{\lambda_0}$, определяющий стоимость единицы производительности системы. Очевидно, что оптимальной системе в этом случае, как и выше, соответствует система, у которой $F_3 - \min$.

2. Задано ограничение на системную производительность: $\lambda_0 \geq \lambda_0^*$, а стоимость системы используется в качестве критерия эффективности: $F = S$.

Определить производительности устройств $V_1,...,V_n$, при которых стоимость системы будет минимальна: $S - \min$.

Такая постановка задачи называется задачей проектирования системы минимальной стоимости.

3. Задано ограничение на стоимость системы: $S \leq S^*$, а системная производительность λ_0 используется в качестве критерия эффективности: $F = \lambda_0$.

Определить производительности устройств $V_1,...,V_n$, при которых системная производительность будет максимальна: $\lambda_0 - \max$.

Такая постановка задачи называется задачей проектирования системы максимальной производительности.

Решения сформулированных задач проектирования базируются на аналитических методах расчёта характеристик замкнутых сетевых моделей и реализуются по аналогии с задачами проектирования на базе разомкнутых сетевых моделей. Для замкнутой сетевой модели, в которой циркулирует произвольное количество заявок M, отсутствуют в явном виде аналитические зависимости характеристик функционирования от параметров системы. Поэтому ниже при решении задач проектирования на замкнутых моделях для простоты изложения рассматриваются замкнутые сетевые модели, в которых циркулирует только одна заявка: $M = 1$.

6.2.6. Проектирование системы на основе замкнутой модели и составного критерия эффективности

Положим, что в системе, состоящей из n устройств постоянно циркулирует только один запрос ($M = 1$), средние ресурсоёмкости обработки которых в устройствах $1,...,n$ равны $\theta_1,...,\theta_n$. В процессе обработки запрос в

среднем α_i раз попадает в устройство $i = 1,\ldots,n$. Пусть, как и ранее, стоимость системы зависит от производительности $V_1,...,V_n$ устройств и определяется как

$$S = \sum_{i=1}^{n} s_i = \sum_{i=1}^{n} \xi_i V_i , \qquad (106)$$

где ξ_i – стоимостной коэффициент пропорциональности.

Задача проектирования заключается в определении производительностей устройств $V_1,...,V_n$, обеспечивающих минимум критерия эффективности:

$$F = k_1 U + k_2 S , \qquad (107)$$

где U – среднее время пребывания запроса в системе; S – стоимость системы; k_1, k_2 – весовые коэффициенты ($k_1 + k_2 = 1$).

Представим описанную систему в виде замкнутой сетевой модели, в которой циркулирует одна заявка: $M = 1$. Очевидно, что в этом случае время пребывания заявок в узле $i = 1,\ldots,n$ будет равно длительности обработки заявки в соответствующем устройстве системы: $u_i = b_i = \theta_i / V_i \quad (i = 1,\ldots,n)$.

Тогда время пребывания заявки в системе:

$$U = \sum_{i=1}^{n} \alpha_i u_i = \sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i} . \qquad (108)$$

Подставляя (106) и (108) в (107) , получим:

$$F = k_1 \sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i} + k_2 \sum_{i=1}^{n} \xi_i V_i .$$

Для нахождения минимума функции F возьмем частные производные от F по V_i и приравняем их нулю:

$$\frac{dF}{dV_i} = -\frac{k_1 \alpha_i \theta_i}{V_i^{\,2}} + k_2 \xi_i = 0 \quad (i = 1,\ldots,n).$$

Решая полученную систему независимых уравнений, определим оптимальные значения производительностей устройств системы:

$$V_{i_{\text{opt}}} = \sqrt{\frac{k_1 \alpha_i \theta_i}{k_2 \xi_i}} \quad (i = 1,\ldots,n), \qquad (109)$$

Заметим, что в последнем выражении, в отличие от формулы (94), полученной для разомкнутой сетевой модели, отсутствует слагаемое, определяющее нижнюю границу производительности устройства, поскольку в замкнутых сетевых моделях невозможны перегрузки.

Подставляя (109) в (106) и (108) , определим стоимость системы и среднее время задержки запросов в системе:

$$S = \sqrt{\frac{k_1}{k_2}} \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} ;$$

$$U = \sqrt{\frac{k_2}{k_1}} \sum_{i=1}^{n} \sqrt{\alpha_i \xi_i \theta_i} .$$

При этом системная производительность будет равна

$$\lambda_0 = \sqrt{\frac{k_1}{k_2}}\left(\sum_{i=1}^{n}\sqrt{\alpha_i \xi_i \theta_i}\right)^{-1}.$$

6.2.7. Проектирование системы заданной стоимости на основе замкнутой модели

Пусть, как и в предыдущем пункте, система, состоящая из n устройств, в которой циркулирует один запрос ($M=1$), представляется в виде замкнутой сетевой модели. Средние ресурсоёмкости обработки запроса в узлах модели, отображающих функционирование устройств системы, равны $\theta_1,...,\theta_n$. Запросы в среднем α_i раз попадают в устройство $i=1,\ldots,n$ за время нахождения в системе (сети).

Положим, что на стоимость системы, определяемую выражением (106), наложено ограничение:

$$S \le S^*. \qquad (110)$$

Задача проектирования системы формулируется следующим образом: определить производительности $V_1,...,V_n$ устройств, обеспечивающие максимальную системную производительность $\lambda_0 - \max$ при условии, что стоимость системы не превышает заданного значения S^*.

Из (104) при $M=1$ получим $\lambda_0 = \frac{1}{U}$, откуда следует, что требование обеспечить максимальную системную производительность λ_0 равносильно требованию минимизировать время пребывания запросов в системе U.

Стоимость системы S и время пребывания запросов в системе U определяются выражениями (106) и (108).

Для решения поставленной задачи воспользуемся методом неопределённых множителей Лагранжа.

Функция Лагранжа будет иметь вид:

$$G = U + \gamma(S - S^*),$$

где γ – неопределённый множитель Лагранжа.

Для того чтобы найти минимум функции Лагранжа G, подставим (106) и (108) в последнее выражение и возьмем частные производные от G по V_i:

$$\frac{\mathrm{d}G}{\mathrm{d}V_i} = -\frac{\alpha_i \theta_i}{V_i^2} + \gamma \xi_i \qquad (i=\overline{1,n}).$$

Приравняв последнее выражение к нулю, решим полученную систему уравнений и найдем V_i:

$$V_i = \frac{1}{\sqrt{\gamma}}\sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \quad (i=\overline{1,n}). \qquad (111)$$

Полагая, что решение задачи ищется на границе ограничения (110), определим неопределённый множитель Лагранжа из условия:

$$S = S^* \quad \text{или} \quad \sum_{i=1}^{n} \xi_i V_i = S^*.$$

Подставив (111) вместо V_i в последнее выражение, после некоторых преобразований получим:

$$\frac{1}{\sqrt{\gamma}} = \frac{S^*}{\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}}.$$

Подставляя последнее выражение в (111), получим окончательное решение в виде оптимальных значений производительности устройств проектируемой системы:

$$V_{i_{\text{opt}}} = \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \cdot \frac{S^*}{\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}} \qquad (i = \overline{1,n}). \tag{112}$$

Производительность устройств системы должна быть распределена пропорционально $\sqrt{\frac{\alpha_i \theta_i}{\xi_i}}$ $(i = \overline{1,n})$, т.е. чем чаще попадает запрос в устройство и чем больше ресурсоёмкость обработки запроса в этом устройстве, тем большей должна быть его производительность.

Таким образом, минимум среднего времени пребывания запросов в системе и, следовательно, максимум производительности системы стоимостью S^* достигается при распределении производительностей $V_1,\ldots,V_n$ устройств в соответствии с выражением (112). Такое распределение называется оптимальным для системы заданной стоимости, представляемой замкнутой сетевой моделью.

Максимальная производительность системы стоимостью S^* будет равна

$$\lambda_0 = \frac{S^*}{\left(\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \right)^2}.$$

6.2.8. Проектирование системы заданной производительности на основе замкнутой модели

Положим, что ограничение налагается на системную производительность:

$$\lambda_0 \geq \lambda_0^*. \tag{113}$$

Задача проектирования системы формулируется следующим образом: определить производительности $V_1,\ldots,V_n$ устройств, обеспечивающие минимальную стоимость системы $S - \min$ при условии, что системная производительность не менее заданного значения λ_0^*.

Заметим, что системная производительность и среднее время пребывания запроса в системе с учетом (104) связаны зависимостью: $\lambda_0 = \frac{1}{U}$, откуда следует, что ограничение (113) эквивалентно ограничению:

$$U \le \frac{1}{\lambda_0^*}, \qquad (114)$$

которое будем использовать в дальнейших наших выкладках.

Стоимость системы S определяется выражением (106), а системная производительность как величина, обратная среднему времени пребывания (108) запроса в системе:

$$\lambda_0 = \left(\sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i} \right)^{-1}. \qquad (115)$$

Для решения задачи проектирования системы заданной производительности воспользуемся методом неопределённых множителей Лагранжа.

С учетом (114) запишем функцию Лагранжа в следующем виде:

$$G = S + \gamma (U - \frac{1}{\lambda_0^*}),$$

где γ – неопределённый множитель Лагранжа.

Подставим (106) и (115) в последнее выражение и возьмем частные производные от G по V_i:

$$\frac{\mathrm{d}G}{\mathrm{d}V_i} = \xi_i - \gamma \frac{\alpha_i \theta_i}{V_i^2} \qquad (i = \overline{1,n}).$$

Приравняв последнее выражение нулю, решим полученное уравнение и найдем V_i:

$$V_i = \sqrt{\gamma} \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \qquad (i = \overline{1,n}). \qquad (116)$$

Полагая, что решение задачи ищется на границе ограничения (114) найдем неопределённый множитель Лагранжа из условия:

$$U = 1/\lambda_0^* \quad \text{или} \quad \sum_{i=1}^{n} \frac{\alpha_i \theta_i}{V_i} = \frac{1}{\lambda_0^*}.$$

Подставив (116) вместо V_i в последнее выражение, получим:

$$\sqrt{\gamma} = \lambda_0^* \sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i}.$$

Подставляя последнее выражение в (116), получим окончательное решение в виде оптимальных значений производительности устройств проектируемой системы:

$$V_{i_{\text{opt}}} = \lambda_0^* \sqrt{\frac{\alpha_i \theta_i}{\xi_i}} \sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \qquad (i = \overline{1,n}). \qquad (117)$$

Производительность устройства распределяется пропорционально $\sqrt{\frac{\alpha_i \theta_i}{\xi_i}}$ $(i = \overline{1,n})$, причем она тем больше, чем больше ограничение λ_0^*.

Таким образом, минимум стоимости системы при ограничении на системную производительность λ_0^* достигается при распределении производительностей $V_1,...,V_n$ устройств в соответствии с (117). Такое распределение называется оптимальным для системы заданной производительности, при этом стоимость системы будет равна

$$S = \lambda_0^* \left(\sum_{i=1}^{n} \sqrt{\xi_i \alpha_i \theta_i} \right)^2 .$$

Стоимость системы, определяемая этим выражением, минимальна при рассматриваемой постановке задачи проектирования.

Задачи проектирования на основе замкнутых сетевых моделей рассмотрены для частного случая, когда в системе циркулирует только одна заявка. Для произвольного числа заявок в системе невозможно получить аналитическое решение задачи оптимизации. В этом случае применяются приближенные методы, либо разрабатываются алгоритмы, которые реализуют целенаправленный перебор различных вариантов построения проектируемой системы. При этом наряду с аналитическими методами могут применяться численные методы и имитационное моделирование.

Вопросы для самопроверки

- Привести примеры дискретных систем.
- В чем отличие базовых моделей дискретных систем от сетевых?
- В чем проявляется неоднородность нагрузки?
- В чем отличие замкнутых сетевых моделей от разомкнутых?
- Перечислить основные показатели эффективности дискретных систем.
- Понятие производительности системы.
- В чем отличие пропускной способности от производительности системы?
- В чем суть структурно-функционального проектирования?
- В чем суть нагрузочного проектирования?
- Какие исходные данные используются в процессе структурно-функционального проектирования на системотехническом уровне?
- Перечислить типовые подходы к проектированию дискретных систем.
- Что понимается под минимальной конфигурацией системы.
- В чем суть проектирования системы минимальной конфигурации?
- В чем суть оптимального проектирования системы?
- Краткая характеристика аналитических методов проектирования.
- Краткая характеристика имитационных методов проектирования.
- В чем суть комбинированного метода проектирования реальных систем?
- В чем суть метода средних значений?
- Что представляют собой формулы Литтла?
- В чем суть метода марковских случайных процессов?
- Что такое марковский случайный процесс?
- При каком условии случайный процесс *с непрерывным временем* является марковским?
- Какие параметры используются для описания марковского случайного процесса с дискретными состояниями и непрерывным временем?
- Что представляет собой матрица интенсивностей переходов?
- Эргодическое свойство случайных процессов.
- Что такое стационарные вероятности случайного процесса?.
- Перечислить этапы разработки марковской модели исследуемой системы.
- В каких случаях применяется имитационное моделирование в процессе системотехнического проектирования.
- Какой подход является наиболее универсальным при проектировании дискретных систем со стохастическим характером функционирования?
- При решении каких задач аналитические методы оказываются наиболее эффективными?
- Задача определения нижней границы производительности устройства.
- Что такое нижняя граница производительности устройства?

- Что произойдет, если производительность устройства меньше нижней границы?
- Задача определения минимальной производительности устройства с учётом ограничения на среднее время пребывания заявок.
- Что произойдет, если производительность устройства будет находиться в интервале между нижней границей и минимальной производительностью?
- Задача определения оптимальной производительности устройства.
- Какой зависимостью связаны стоимость системы и производительности устройства?
- Записать критерий эффективности, связывающий время пребывания заявок в системе и стоимость системы.
- Проиллюстрировать на графике задачу оптимального проектирования на основе критерия эффективности, связывающего время пребывания заявок в системе и стоимость системы.
- Задача определения оптимальной производительности устройства с учетом ёмкости накопителя.
- Задача определения оптимальной производительности устройства с учётом ограничения на время пребывания.
- Задача оценки ёмкости накопителя.
- Понятие допустимой нагрузки.
- Задача определения допустимой нагрузки при заданной производительности устройства и ограничении на время пребывания заявок в системе.
- Показать на графике характер зависимости вероятности потери заявок от ёмкости накопителя.
- Показать на графике характер зависимости производительности устройства от ёмкости накопителя.
- Сформулировать постановки задач проектирования системы с одним устройством и накопителем ограниченной ёмкости.
- Перечислить основные характеристики эффективности системы с накопителем ограниченной ёмкости.
- Сформировать обобщённый критерий эффективности для системы с накопителем ограниченной ёмкости.
- Сформулировать задачу проектирования системы с накопителем ограниченной ёмкости.
- Проиллюстрировать на графике задачу определения оптимальных значений ёмкости накопителя и производительности устройства системы с одним устройством и накопителем ограниченной ёмкости.
- Выполнить **с**равнительный анализ систем с накопителями ограниченной и неограниченной ёмкости.
- При каких условиях замена системы с накопителем ограниченной ёмкости системой с накопителем неограниченной ёмкости позволяет получить результаты, погрешность которых не превышает 5%?

- Принцип расчёта характеристик систем с несколькими устройствами и индивидуальными накопителями неограниченной ёмкости.
- Задача определения минимального количества устройств в системе.
- Проблема проектирования систем с несколькими устройствами
- Проиллюстрировать на графике зависимость времени пребывания в системе от количества устройств при условии, что их суммарная производительность (быстродействие) остается постоянной.
- Задача проектирования системы с несколькими устройствами и накопителями ограниченной ёмкости.
- Сформулировать постановки задачи проектирования системы с несколькими устройствами и общим накопителем ограниченной ёмкости.
- При выполнении каких условий неоднородный поток заявок можно свести к однородному?
- По каким формулам рассчитываются параметры однородного потока при сведении неоднородного потока заявок к однородному?
- В каких случаях неоднородный поток невозможно свести к однородному?
- Чем абсолютный приоритеты отличаются от относительных?
- Какие варианты продолжения обслуживания прерванной заявки возможны в случае абсолютных приоритетов?
- Из каких составляющих складывается время ожидания заявок в случае абсолютных приоритетов?
- Сформулировать закон сохранения времени ожидания.
- При каких условиях выполняется закон сохранения времени ожидания?
- Записать критерий эффективности в виде функции штрафа для решения задачи распределения приоритетов в системах с неоднородным потоком заявок при отсутствии ограничений на времена пребывания заявок в системе.
- При каком условии дисциплина обслуживания с относительным приоритетом будет лучше бесприоритетной дисциплины?
- По какому правилу должны назначаться приоритеты заявкам в случае их равноценности?
- Способы описания дисциплин со смешанными приоритетами.
- Что представляет собой матрица приоритетов?
- Какие требования предъявляются к элементам матрицы приоритетов?
- Привести примеры матрицы приоритетов.
- Что представляет собой схема ДО?
- Понятие БП-группы.
- Понятие канонической матрицы приоритетов.
- Чему равно количество вариантов заполнения канонической МП в общем случае?

- Чему равно количество вариантов заполнения канонической МП в случае 4-х классов заявок?
- Понятие корректности матрицы приоритетов.
- Привести пример некорректной матрицы приоритетов.
- Пояснить понятие некорректности матрицы приоритетов на примере.
- Правила построения корректных канонических матриц приоритетов.
- Постановка задачи проектирования систем со смешанными приоритетами.
- Понятие средних ограничений.
- Понятие вероятностных ограничений.
- Отличие вероятностных ограничений от средних.
- Этапы функционального проектирования систем реального времени.
- Постановка задачи функционального проектирования систем реального времени.
- Перечень исходных данных для решения задачи функционального проектирования систем реального времени.
- Оценка нижней границы производительности устройства при решении задачи синтеза дисциплины обслуживания в системах реального времени.
- Проиллюстрировать на графике задачу выбора ДО.
- Проиллюстрировать на графике задачу определения оптимальной производительности системы для найденной ДО.
- Основная проблема проектирования систем реального времени с вероятностными ограничениями.
- Перечень параметров, необходимых для описания линейной разомкнутой однородной экспоненциальной сетевой модели.
- Условие отсутствия перегрузок в системах, представляемых разомкнутыми сетевыми моделями.
- Этапы расчёта характеристик функционирования линейной разомкнутой однородной экспоненциальной сетевой модели.
- На основе какого преобразования сети базируется расчёт характеристик линейной разомкнутой однородной экспоненциальной сетевой модели?
- Что такое коэффициент передачи в сетевой модели и что он характеризует?
- Записать формулы для расчёта сетевых характеристик разомкнутых моделей на основе известных значений узловых характеристик.
- Перечислить способы ликвидации перегрузок в системе.
- Задача определения допустимой нагрузки в разомкнутых сетевых моделях.
- Сформулировать постановки задач проектирования систем на основе разомкнутых сетевых моделей.
- Задача проектирования системы, представляемой разомкнутой сетевой моделью, на основе составного критерия эффективности.

- Задача проектирования системы заданной стоимости на основе разомкнутой сетевой модели.
- В чем суть метода неопределенных множителей Лагранжа?
- Задача проектирования системы с заданным временем пребывания на основе разомкнутой сетевой модели.
- Перечислить параметры, используемые для описания линейных замкнутых однородных экспоненциальных моделей.
- Этапы расчёта характеристик функционирования линейных замкнутых однородных экспоненциальных сетевых моделей.
- Почему для замкнутой сетевой модели на первом этапе рассчитываются только коэффициенты передач?
- Рекомендации по проектированию систем на основе замкнутых сетевых моделей.
- Нарисовать и пояснить зависимости производительности и времени пребывания заявок в замкнутой сетевой модели от числа циркулирующих заявок.
- Какая система называется сбалансированной?
- Основные показатели эффективности функционирования систем, представляемых замкнутыми сетевыми моделями.
- Какие задачи проектирования могут решаться на базе замкнутых сетевых моделей?
- Сформулировать постановки задач проектирования на основе замкнутых сетевых моделей.
- Задача проектирования системы, представляемой замкнутой сетевой моделью, на основе составного критерия эффективности.
- Задача проектирования системы заданной стоимости на основе замкнутой сетевой модели.
- Задача проектирования системы с заданным временем пребывания на основе замкнутой сетевой модели.

Список литературы

1. Алиев Т.И. Основы проектирования систем. Учебное пособие. – СПб: Университет ИТМО, 2015. – 120 с.

2. Алиев Т.И. Основы моделирования дискретных систем. Учебное пособие. – СПб: СПбГУ ИТМО, 2009. – 363 с. ISBN 978-5-7577-0336-7.

3. Жожикашвили В.А., Вишневский В.М. Сети массового обслуживания. Теория и применение к сетям ЭВМ. – М.: Радио и связь, 1988. – 192 с.: ил.

4. Клейнрок Л. Вычислительные системы с очередями. – М.: Мир, 1979. – 600 с.

5. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд. / Олифер В.Г., Олифер Н.А. – СПб: Питер, 2006. – 958 с.: ил. ISBN 978-5-469-00504-9. (Глава 7. Методы обеспечения качества обслуживания).

6 Основы теории вычислительных систем / С.А.Майоров, Г.И.Новиков, Т.И.Алиев, Э.И.Махарев, Б.Д.Тимченко. – М.: Высшая школа, 1978. – 408 с.

7. Вишневский В.М., Семенова О.В. Системы поллинга: теория и применение в широкополосных беспроводных сетях. – М.: Техносфера, 2007. – 312 с. ISBN 978-5-94836-166-6.

8. Столингс В. Современные компьютерные сети. – СПб.: Питер, 2003. – 783 с.: ил. ISBN 978-5-94723-327-4. (Часть 3 – Моделирование и оценка производительности).

9. Алиев Т.И., Махаревс Э. Дисциплины обслуживания на основе матрицы приоритетов // Научно-технический вестник информационных технологий, механики и оптики. – 2014. – Т. 88. – № 6. – С. 91-97.

10. Алиев Т.И. Аппроксимация вероятностных распределений в моделях массового обслуживания // Научно-технический вестник информационных технологий, механики и оптики, 2013, № 2 (84). – С. 88-93.