В.И. БОДРОВ, Т.Я. ЛАЗАРЕВА, Ю.Ф. МАРТЕМЬЯНОВ

МЕТОДЫ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ ПРИ ПРИНЯТИИ РЕШЕНИЙ

Издательство ТГТУ

Министерство образования и науки Российской Федерации
Тамбовский государственный технический университет

В.И. БОДРОВ, Т.Я. ЛАЗАРЕВА, Ю.Ф. МАРТЕМЬЯНОВ

МЕТОДЫ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ ПРИ ПРИНЯТИИ РЕШЕНИЙ

Утверждено Ученым советом университета
в качестве учебного пособия

Тамбов
Издательство ТГТУ
2004

УДК 519.7(075)
ББК В183я73
Б75

Рецензенты:
Доктор технических наук, профессор
Ю.В. Литовка
Доктор физико-математических наук, профессор
С.М. Дзюба

Бодров В.И., Лазарева Т.Я., Мартемьянов Ю.Ф.
Б75 Методы исследования операций при принятии решений: Учебное пособие. Тамбов: Изд-во Тамб. гос. техн. ун-та, 2004. 160 с.

Изложены такие методы исследования операций, как графы, сети, теория расписания, теория массового обслуживания, управление запасами, используемые при принятии решений экономического плана.

Учебное пособие предназначено для студентов 4 курса дневного отделения специальности 351400 "Прикладная информатика (в экономике)".

УДК 519.7(075)
ББК В183я73

ISBN 5-8265-0259-2

Учебное издание

БОДРОВ Виталий Иванович,
ЛАЗАРЕВА Татьяна Яковлевна,
МАРТЕМЬЯНОВ Юрий Федорович

МЕТОДЫ ИССЛЕДОВАНИЯ ОПЕРАЦИЙ ПРИ ПРИНЯТИИ РЕШЕНИЙ

Учебное пособие

Редактор Т.М. Глинкина
Компьютерное макетирование Е.В. Кораблевой

Подписано в печать 30.12.2004
Формат 60 × 84 / 16. Бумага офсетная. Печать офсетная
Гарнитура Times New Roman. Объем: 9,30 усл. печ. л.; 9,20 уч.-изд. л.
Тираж 100 экз. С. 900

Издательско-полиграфический центр
Тамбовского государственного технического университета,
392000, Тамбов, Советская, 106, к. 14

ВВЕДЕНИЕ

Целью преподавания курса "Теория принятия решений" при подготовке экономистов по специальности 351400 "Прикладная информатика (в экономике)" является формирование у студентов знаний о математических основах принятия решения, умения применять эти знания при решении конкретных задач. Количественная оценка принятого решения может производиться различными методами, среди которых выделяются методы исследования операций. В учебном пособие рассматриваются такие методы, как графы, сети, теория расписаний, теория массового обслуживания, управление запасами, наиболее часто используемые при принятии технико-экономических решений.

Материал курса может быть использован студентами при выполнении курсовых работ по специальности, а также при выполнении квалификационной работы по специальности.

Учебное пособие полностью отражает читаемый курс.

ТЕОРИЯ ГРАФОВ

Для решения многих задач, в частности, задач теории расписаний, сетевых задач, задач поиска решений в пространстве состояний, теории игр и других используется теория графов.

Основные понятия теории графов

Графом называется особого типа схема. Эта схема состоит из кружков (или точек), некоторые из которых соединены линиями и имеют определенный физический смысл.

Кружки называются вершинами графа, соединительные линии – ребрами графа или дугами графа. На рис. 1.1 изображен пример плоского графа.

Вершины графа представляют собой: объект, событие, состояние.

Термин "объект" очень широкий и не требует особых уточнений: это может быть город, станок, человек и т.д. в различных задачах.

Событие – это то, что произошло с некоторыми объектами, например: покупка елки, покупка креста для елки, установка елки и т.д.

Состояние – это некоторый набор признаков (например, список параметров), которые характеризуют объект и позволяют судить о его дальнейшем поведении.

Ребро графа может обозначать:

– возможность перехода от одного объекта к другому, возможность того, что за одним объектом может следовать другой;

– возможность поступления одного события после наступления предыдущего, возможность того, что одно событие последует за другим;

– возможность того, что за некоторым состоянием последует другое состояние, что данное состояние перейдет в другое.

Ребро графа может оканчиваться стрелкой односторонней (рис. 1.2, *а*), двусторонней (рис. 1.2, *б*) или вообще быть без стрелок (рис. 1.2, *в*).

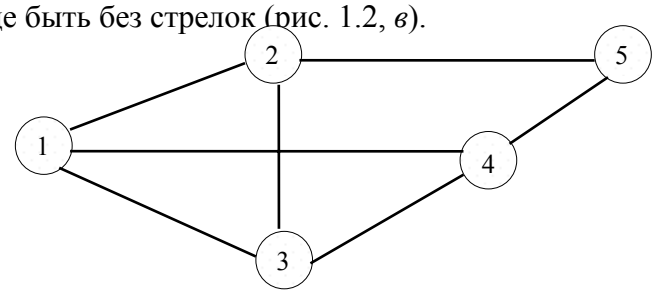
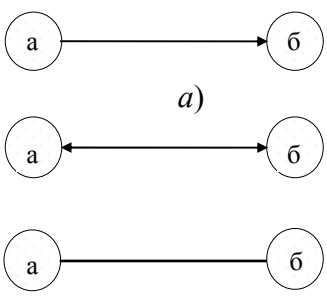


Рис. 1.1 Плоский граф



б)

в)

Рис. 1.2 Ребро графа:
а – с односторонней стрелкой; *б* – с двухсторонней стрелкой; *в* – без стрелок

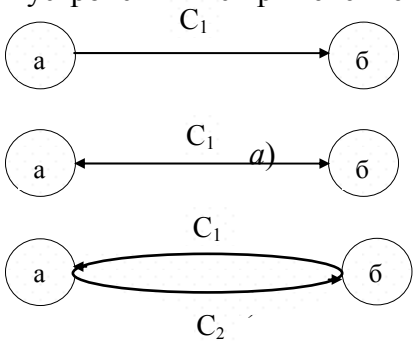
Односторонняя стрелка (рис. 1.2, *а*) обозначает, что из состояния "а" можно перейти в состояние "б", обратный же переход невозможен. Двухсторонняя стрелка (рис. 1.2, *б*) обозначает, что возможен переход из "а" в "б" и из "б" в "а".

Если на ребре стрелок нет (рис. 1.2, *в*), то это обычно означает взаимный переход, если по договоренности или из контекста не следует, что этот переход однонаправленный.

На ребре иногда указывается стоимость перехода из одного состояния в другое (рис. 1.3), причем переход из "а" в "б" может иметь одну цену, а из "б" в "а" другую. Например, изготовление на одном и том же оборудовании черной краски после белой требует небольших затрат на очистку оборудования, переход же от черной краски к изготовлению белой требует больших затрат.

Ситуация, изображенная на рис. 1.4 показывает, что из состояния "а" можно перейти либо в состояние "б", либо в состояние "в", либо в состояние "г".

В этом случае появляется понятие выбора перехода. Сделав выбор, переходят из состояния "а" в соответствующее возможное состояние. Переход может осуществляться автоматически (в том числе управляющим устройством) с применением некоторого алгоритма выбора перехода π.



а)

в)

Рис. 1.3 Стоимость перехода:
а – из "а" в "б" со стоимостью C_1; *б* – из "а" в "б" и из "б" в "а" с одной стоимостью; *в* – из "а" в "б" и из "б" в "а" с разными стоимостями

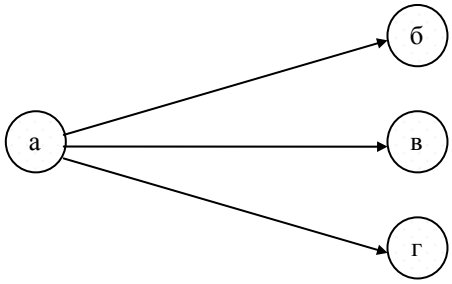


Рис. 1.4 Переходы из состояния "а"

Пусть из некоторого состояния "а" (рис. 1.5) в результате применения π алгоритма система перешла в состояние "б", затем в "в" и наконец в "г".

Последовательность обхода вершин "а-б-в-г" вместе с соответствующими ребрами называется путем "а-б-в-г".

Если существует путь из вершины "а" в вершину "г", то говорят, что "г" достижима из вершины "а". Если такого пути нет, то вершина "г" не достижима из вершина "а".

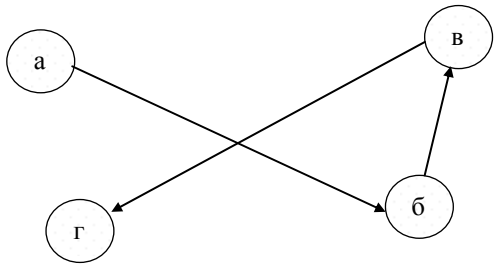


Рис. 1.5 Путь на графе

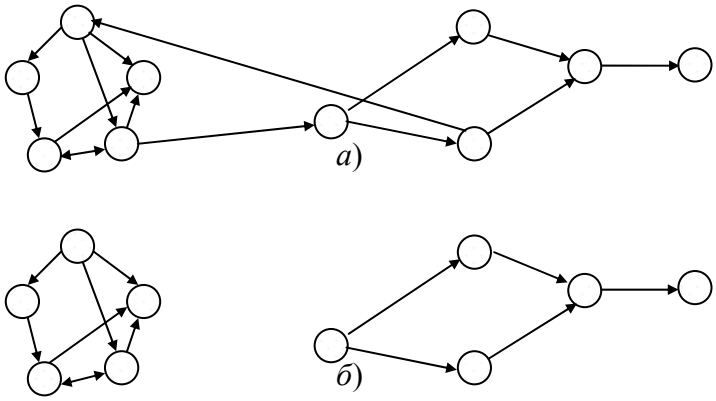


Рис. 1.6 Подграфы:
а – связанные; *б* – изолированные

Если между вершинами "а" и "б" существует путь, говорят, что эти вершины связаны.

Часть графа называется подграфом. Если хотя бы одна вершина подграфа А связана с вершиной подграфа В, то говорят, что эти подграфы связаны (рис. 1.6, *а*). Если же ни одна вершина подграфа А не связана ни с одной вершиной подграфа В, то говорят, что эти подграфы изолированы (рис. 1.6, *б*).

Пути графа классифицируют, на рис. 1.7 представлена классификация этих путей.

Пути делятся на цепи и не цепи.

Цепь (рис. 1.8) отличается тем, что в ней нет общих ребер. Так, "а-б-в-г-д-в-б-е" (рис. 1.8, *б*) не является цепью, так как ребро *s* между "б" и "в" проходится дважды.

Путь "а-б-в-г-д-в-с" (рис. 1.8, *а*) является цепью.

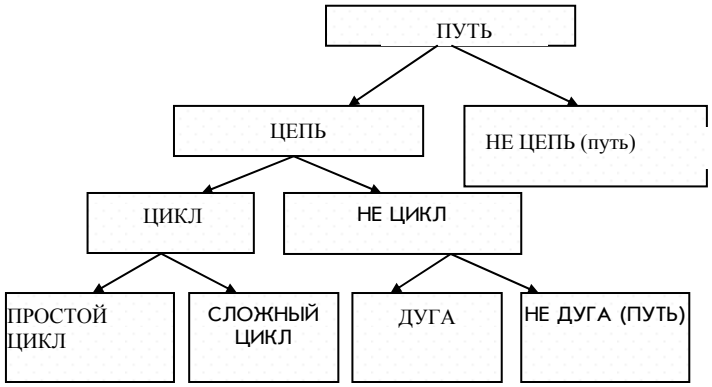
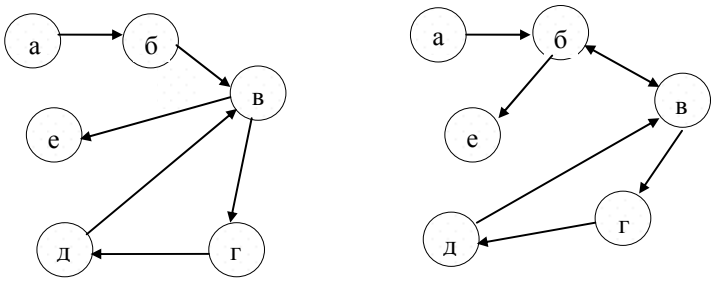
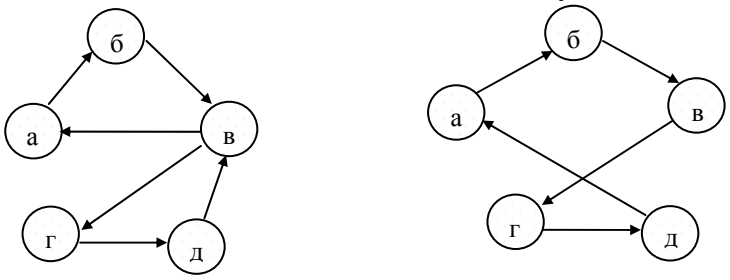


Рис. 1.7 Классификация путей



а) *б*)

Рис. 1.8 Типы путей:
а – "а-б-в-г-д-в-е" – путь; *б* – "а-б-в-г-д-в-б-е" – не цепь



а) *б*)

Рис. 1.9 Циклы:
а – простой; *б* – сложный

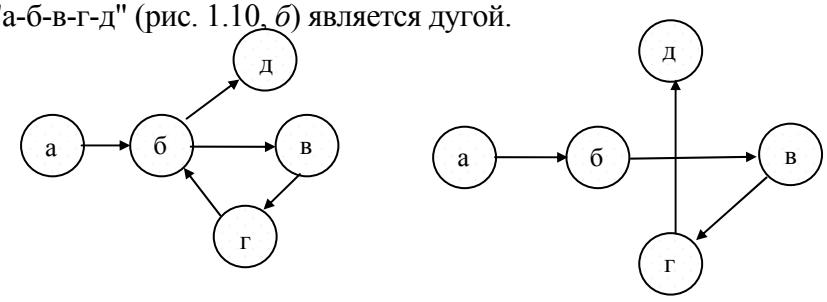
Цепи делятся на циклы и не циклы. Циклы – это замкнутые цепи, т.е. цепи, в которых одна и та же вершина является началом и концом пути. При этом цикл называется простым (рис. 1.9, *а*), если каждая вершина проходит только один раз.

Цикл "а-б-в-г-д-в-а" (рис. 1.9, *а*) является сложным, так как вершина "в" проходится два раза. Цикл "а-б-в-г-д-а" (рис. 1.9, *б*) – простой.

Цепи, не являющиеся циклическими, делятся на дуги и не дуги.

Цепь, не являющаяся циклом, называется дугой, если ни одна вершина в ней не проходится дважды.

Так, на рис. 1.10, *а* цепь "а-б-в-г-б-д" не является дугой, так как вершина "б" проходится дважды, цепь "а-б-в-г-д" (рис. 1.10, *б*) является дугой.



а) *б*)

Рис. 1.10 Виды цепей:
а – не дуга; *б* – дуга

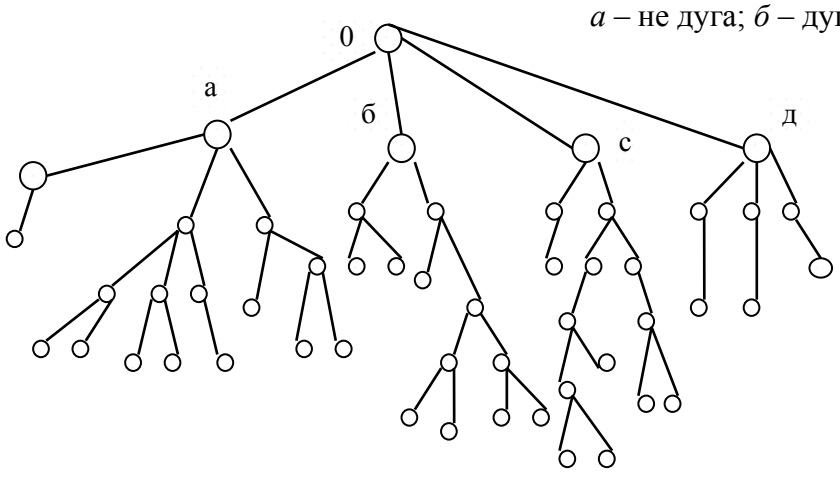


Рис. 1.11 Дерево

Деревом называется связанный граф, такой, что все возможные пути в нем являются дугами, т.е. не содержат циклов и не проходят ни одной вершины более одного раза (рис. 1.11).

Начальная точка "0" (рис. 1.11) называется корнем дерева, отходящие от нее ребра "0а", "0б", "0с", "0д" отделяют изолированные подграфы, похожие на ветви дерева. Именно из-за этого сходства такой граф получил название дерева.

Дерево – это наиболее простой вид графа, легко поддающийся исследованию. Исследование других более сложных графов сводится некоторым усложненным алгоритмом к исследованию соответствующего дерева.

Корень дерева называется вершиной нулевого уровня. Вершины, в которые можно перейти из корня (вершины "a", "b", "c" на рис. 1.12)

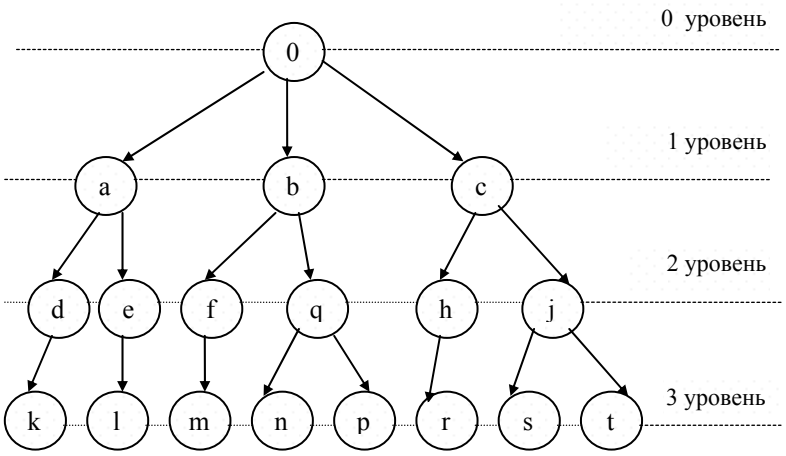


Рис. 1.12 Вершины разных уровней

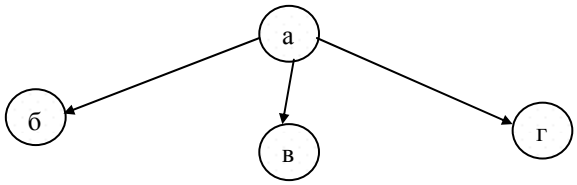


Рис. 1.13 Отношение между вершинами

называются вершинами первого уровня. Вершины, в которые можно попасть за один переход, называются вершинами второго уровня. Вершинами *n*-го уровня называются вершины, в которые можно попасть за один переход из вершины *n* – 1-го уровня.

Если из вершины "а" (*n* – 1)-го уровня следуют вершины "б", "в", "г" следующего *n*-го уровня (рис. 1.13), то вершина "а" называется материнской, вершины "б", "в", "г" называются дочерними. Также вершина "а" называется "предком", а вершины "б", "в", "г" – потомками.

Вершины, из которых не выходят ребра, называются конечными вершинами. Конечных вершин может быть много.

Начальных вершин (корней) также может быть много. В случае дерева каждая начальная вершина (корень) выделяет изолированное дерево, так как у дерева в отличие от произвольного графа отсутствуют циклы.

Поиск решений в пространстве состояний

Первоначально следует рассмотреть состояние системы в виде дерева.

Деревья в пространстве состояния могут быть заданы в эксплицидной и имплицидной форме.

Дерево, заданное в эксплицидной форме, представляет собой граф с полным обозначением всех вершин (состояний) и всех ребер перехода от одного состояния к другому. Если такой граф задан, любое решение очевидно – его можно показать на графе. Но к сожалению такой граф можно построить лишь для очень простых задач, т.е. задач с очень малым числом состояний.

Если представить, что состояние – это список букв и знаков русского алфавита, то корневой вершиной будет вершина, состояние которой "а". Эта вершина в качестве дочерних имеет вершины, в которых состояниями будут две буквы аа, аб, ав, ..., ая, а также буква а и знаки – $а_\vee$, а,, а!, а? и т.п.

Каждая из этих вершин имеет столько же дочерних. Этот гигантский бесконечный граф в качестве вершин где-то имеет набор – "Анна Каренина", "Преступление и наказание", другая вершина – это почти "Анна Каренина", отличающаяся, может быть, строкой, или буквой, или ошибкой. Все в этом графе гениально, в нем еще не написанные рассказы, все мудрые мысли прошлых и будущих мыслителей, теоремы и их доказательства, изобретения будущих тысячелетий.

Однако, такой граф нельзя построить в явном виде. Обычно дерево состояния строят в имплицидной форме.

Дерево в имплицидной форме считается заданным, если а) определено понятие состояния (формализованы обозначение каждого состояния); б) задано начальное состояние; в) задан оператор, переводящий одно состояние в дочернее.

Этот оператор обозначается $F(a_{n-1} \to a_n)$, а $R(a_n)$ – оператор раскрытия вершин, под которым понимают построение всех дочерних вершин для вершины a_n.

Имея оператор $R(a_n)$, можно, последовательно применяя его к вершинам, из имплицидного дерева получить эксплицидное на любую глубину. Применяя последовательно оператор $R(a_n)$ к оператору $F(a_{n-1} \to a_n)$, формируют одну за другой дочерние вершины для материнской вершины a_n. При этом оператор $F(a_{n-1} \to a_n)$ формирует новое состояние a_{n+1} следующим образом:

а) либо применяя правила переписывания строк, определяющие состояние, например, последовательную запись расположения фигур на шахматной доске $К_р$q1, Фf3, ЛФ1 и т.д. Эти правила иногда называются продукциями;

б) в виде таблицы, в которой заданы вершины, дочерние вершины, стоимость всех связывающих их ребер;

в) с применением оператора (вычислительной процедуры), который определяет (вычисляет) все параметры дочерних вершин, описывающие их состояние, рассчитывает стоимости соединяющих их ребер. В этом случае говорят, что граф задан алгоритмически (неявно), причем вычисления могут быть очень сложными.

При формулировке задачи принятия решения необходимо формализовать конечное (целевое) состояние. Например, при игре в шахматы конечным желаемым состоянием является состояние мата королю соперника.

В этом случае простейшая задача оптимизации (принятия решения) на графе состояний заключается в нахождении пути (последовательности состояний) из начального состояния в целевое, при котором целевая функция принимает минимальное (максимальное) значение.
В качестве целевой функции может выступать число пройденных вершин (это часто тождественно времени решения) или затраты (прибыль), которые на каждом этапе характеризуются стоимостью ребер, соединяющих вершины на выбранном пути.

Если конечных состояний много, то необходимо найти такое конечное состояние и такой путь из начального состояния в конечное, при котором целевая функция будет минимальна.

Если и начальных состояний много, то необходимо найти такое начальное и такое конечное состояние и такой соединяющий их путь, при котором целевая функция примет минимальное значение.

Методы поиска оптимальных решений в пространстве состояний

Пусть оператор $R(a)$ задан, этот оператор строит все вершины a_i, дочерние вершине a, и устанавливает у каждой из них указатель (стрелку) на ту вершину, которая является материнской. Это позволяет

найти путь от любой раскрытой вершины к корню дерева, переходя по стрелке последовательно к материнским вершинам.

Ниже рассматриваются различные методы поиска оптимального пути на графе. При этом рассматриваются деревья, как более простые графы. Все методы поиска на деревьях отличаются правилами выбора вершины, подлежащей раскрытию в первую очередь.

МЕТОДЫ СЛЕПОГО ПОИСКА

К методам слепого поиска относятся методы, которые принимают решение о раскрытии очередной вершины по тому результату, который уже достигнут, не принимая во внимание (не прогнозируя), какой успех ожидается при дальнейшем движении по тому или иному пути.

К методам слепого поиска относятся:

- метод полного перебора (перебора в ширину);
- метод перебора в глубину;
- метод равных цен.

Применение перечисленных методов удобно рассмотреть на примере решения задачи коммивояжера, граф которой изображен на рис. 1.14.

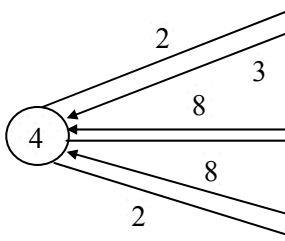


Рис. 1.14 Граф задачи коммивояжера

Коммивояжеру требуется последовательно обойти все города, начиная с первого, и снова вернуться в него. При этом стоимость пути должна быть минимальной. Стоимость пути из города в город (из вершины в вершину) указана на ребрах графа (рис. 1.14), причем стоимость на прямом и обратном пути может быть различной.

Стоимость дорог для разных направлений приведена в табл. 1.1.

1.1 Стоимость дорог

Города	1	2	3	4
1	0	2	15	3
2	1	0	5	8
3	6	2	0	2
4	2	8	3	0

МЕТОД ПЕРЕБОРА В ШИРИНУ (ПОЛНОГО ПЕРЕБОРА)

В данном методе вершины раскрываются в том порядке, в котором они появляются, т.е. первой раскрывается вершина, которая первой и появляется. Другими словами, в этом методе вершины раскрываются слоями: сначала вершины первого уровня, затем второго и т.д.

Вершиной первого уровня будет корень дерева, соответствующий стартовому городу 1.

Далее необходимо формализовать состояние списка городов, которые коммивояжер уже прошел. Эти состояния в рамках стоят рядом с соответствующей вершиной. Первому городу соответствует состояние

1

.

1

Вершина раскрывается, появляются вершины второго уровня (рис. 1.15), соответствующие городам 2, 3, 4, в которые имеет право направиться коммивояжер. Соответствующие состояния дочерних вершин будут 12 , 13 , 14 . Эти состояния представляют собой список уже пройденных городов.

Затем применяется оператор раскрытия к вершинам второго уровня. При этом учитывается, что из вершины "а" коммивояжер может направиться только в города 3 и 4, в город 1 возвращаться рано, так как коммивояжер не прошел еще всех городов. Применяя этот алгоритм, можно построить весь путь, т.е. перевести имплицидное задание дерева в эксплицидное.

При раскрытии вершин 4-го уровня алгоритм отправляет коммивояжера назад в город 1, так как все города пройдены.

Для каждого пути внизу проставлена суммарная цена пути, равная сумме цен соответствующего перехода. Как видно из рис. 1.15, оптимальным маршрутом является путь 1-4-3-2-1. Потери на этом пути

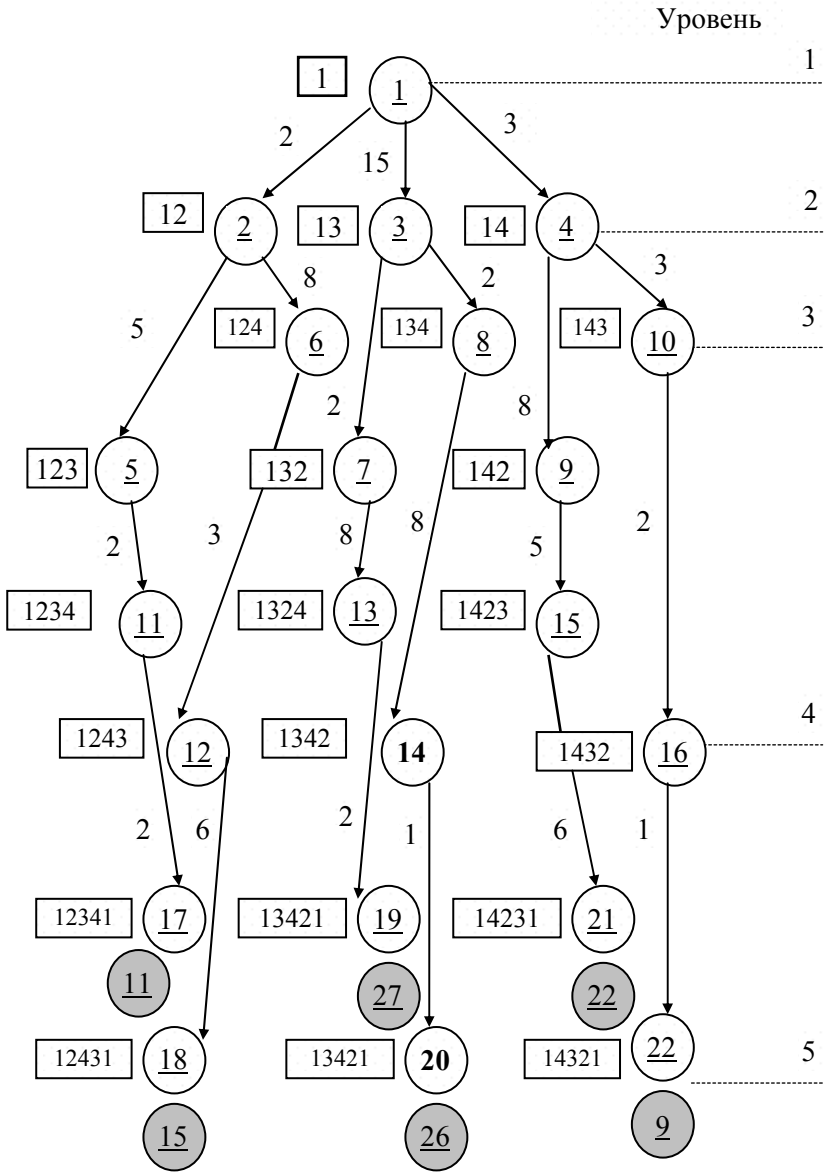


Рис. 1.15 Метод перебора в ширину

минимальны и соответствуют 9 единицам. Всего было построено 22 вершины, из которых раскрыто 16.

Задача коммивояжера не относится лишь только к путешественникам, это модель весьма широкого класса задач. Например, задачи последовательной обработки деталей, обрабатывающихся на разных станках в произвольном порядке; производство красителей на одном и том же оборудовании, когда очистка оборудования от черного цвета для производства белого очень дорогостояща и длительна, а производство черного после белого не требует очистки вовсе. То же относится и к пластмассовым изделиям (пленки разной толщины, технические и пищевые, разной окраски) и ко многим другим задачам. На рис. 1.16 представлена блок-схема алгоритма поиска методом перебора вершин. Этот алгоритм работает со списком вершин "о" и "з".

В список "о" помещаются нераскрытые дочерние вершины, перед раскрытием эти вершины перемещаются в список "з", т.е. вычеркиваются из списка "о" и вносятся в список "з". Всякий раз в список "о" помещается и список "з", а вершины перемещаются вместе с соответствующими указателями на материнскую вершину. Метод поиска вширь всегда находит глобальный экстремум, так как он строит полный эксплицидный граф, т.е. находит стоимостную оценку всех возможных путей. Это дает возможность выбрать среди них наилучший.

1.3.3 МЕТОД ПЕРЕБОРА В ГЛУБИНУ

Согласно этому методу прежде всего раскрывается та вершина, которая имеет наибольший уровень (наибольшую глубину).

Наибольшую глубину имеет вершина, которая построена последней. Таким образом, раскрывается прежде всего вершина, построенная последней (рис. 1.17).

На рис. 1.18 изображена последовательность раскрытия вершин методом перебора вглубь. В верхней части вершины цифра обозначает порядок раскрытия вершины. В центре кружка указывается уровень (глубина) вершины. Алгоритм (рис. 1.17) раскрывает первой вершину самого большого уровня.

Вначале раскрывается a_0 (рис. 1.18), соответствующая городу 1. После ее раскрытия в списке "о" образуются три дочерние вершины 12 , 13 , 14 .

После раскрытия одной из них, в частности 12 (рис. 1.17), образуются две вершины третьего уровня и 123 , 124 , которые согласно алгоритму (рис. 1.17) раскрываются первыми. Таким образом, выявлен первый путь 1-2-3-4-1 и его стоимость С = 11.

Теперь раскрываются оставшиеся вершины, причем на любом пути раскрытие будет прекращено, если затраты на нем превысят или будут равны 11.

Путь 1-2-3-4 обрывается, так как затраты равны 14. Путь 1-3 обрывается сразу, поскольку затраты при переходе из города 1 в город 2 превышают 11 и равны 15 и т.д. Обрыв пути на рис. 1.18 обозначен знаком "х".

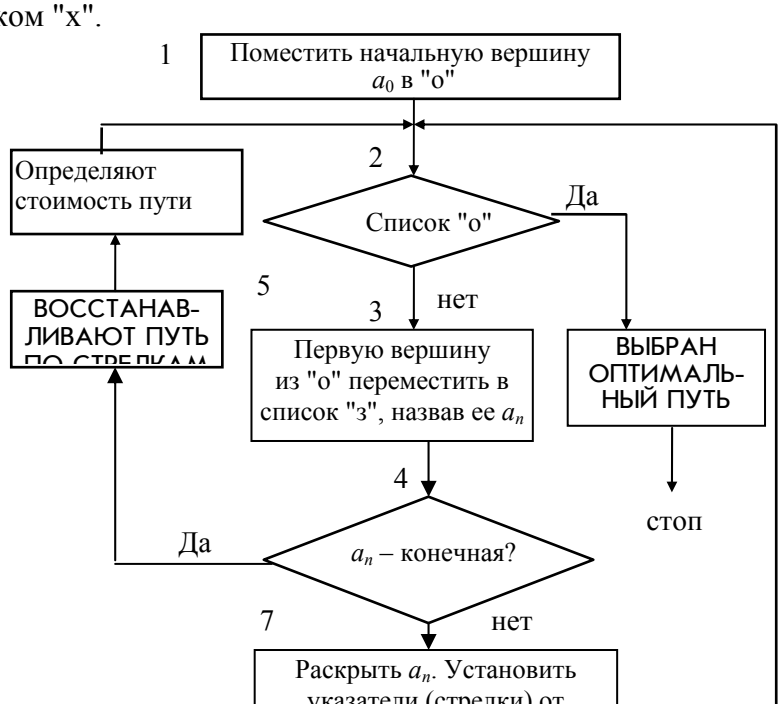


Рис. 1.16 Алгоритм полного перебора (перебора в ширину)

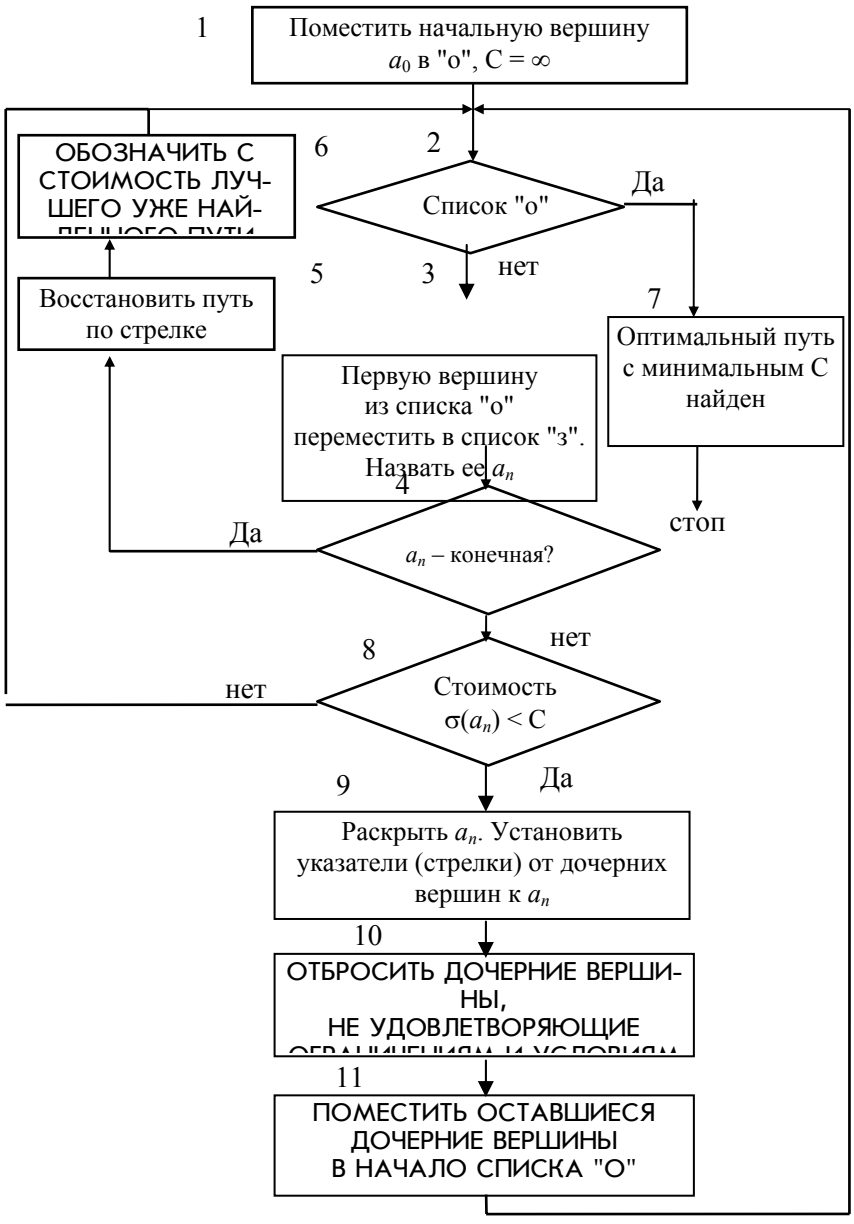


Рис. 1.17 Алгоритм перебора вглубь

Рассмотренный алгоритм значительно эффективнее метода перебора вширь. Действительно для нахождения оптимального пути 1-4-3-2-1 потребовалось раскрыть всего 8 вершин, построить же 13 вершин.

В этом алгоритме движение идет вдоль одного пути до конечной вершины, определяется стоимость этого пути. Затем раскрываются другие пути. При этом движение вдоль следующего пути ведется до тех пор, пока накопленные затраты не превысят уже достигнутых.

Для того, чтобы предотвратить слишком длительное движение вглубь в то время, как путь может оказаться неперспективным, данный алгоритм дополняется некоторой предельной глубиной $\gamma_{пр}$.

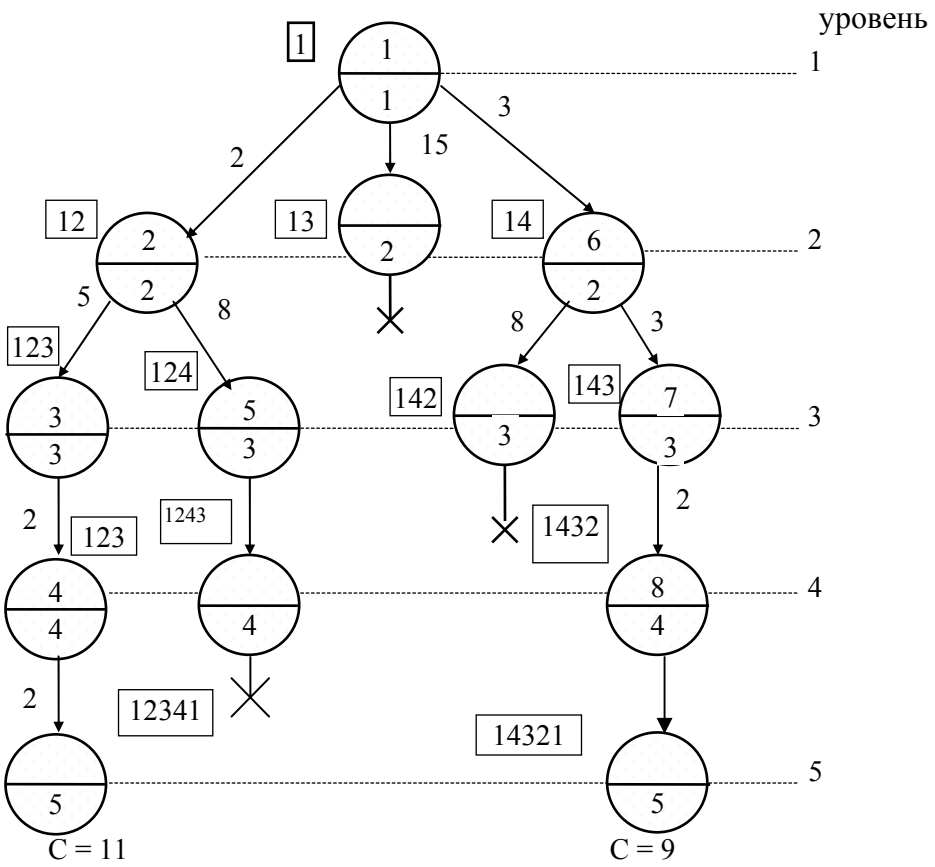


Рис. 1.18 Метод перебора в глубину

Движение вдоль пути прекращается, если глубина достигла $\gamma_{пр}$. После этого раскрывается самая глубокая вершина при условии, что ее глубина не превышает $\gamma_{пр}$.

После того, как все пути достигли глубины $\gamma_{пр}$, кроме тех, которые были оборваны, глубина увеличивается и поиск продолжается до достижения большей глубины или конечного состояния.

1.3.4 МЕТОД РАВНЫХ ЦЕН

Метод равных цен заключается в том, что каждой вершине ставится в соответствие стоимость пути от начальной вершины до рассматриваемой. При этом начальной вершине ставится в соответствие стоимость нуль.

Алгоритм (рис. 1.19) раскрывает ту вершину, стоимость пути для которой минимальна.

Как и раньше, блок 8 проверяет, не превышают ли уже достигнутые на новом пути затраты $\sigma(a_n)$ до вершины a_n стоимости М ранее построенного до конечной вершины пути.

Работа алгоритма равных цен проиллюстрирована на рис. 1.20.

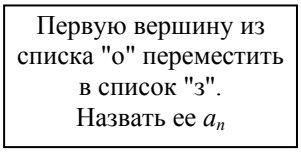
Потребовалось построить всего 11 вершин, а раскрыть 6 вершин, чтобы найти наилучший путь 1-4-3-2-1.

Данный алгоритм всегда находит глобальный оптимальный путь. Как и метод поиска вглубь, он не рассматривает лишь те ветви, на которых не может быть минимальной стоимости, так как затраты уже больше, чем достигнутые на всем ранее построенном пути.

1.4 Метод ветвей и границ

Все алгоритмы, рассмотренные в п. 1.3, относятся к методам слепого поиска. Про них можно сказать, что это "близорукие" стратегии. Так, в методе равных цен для раскрытия выбирается вершина, путь до которой имел минимальную стоимость. Однако дальнейший путь до конечной вершины может оказаться любым, и вполне возможно, что раскрываемая вершина не только не стоит на наилучшем пути, а, может быть, на очень неудачном. Напротив, вершина, путь которой очень дорог, может оказаться перспективной, если дальнейший путь от нее до конечной вершины будет дешевым.

Метод ветвей и границ учитывает не только, какова цена пути до некоторой вершины a_n, но и прогнозирует стоимость дальнейшего пути от этой вершины.



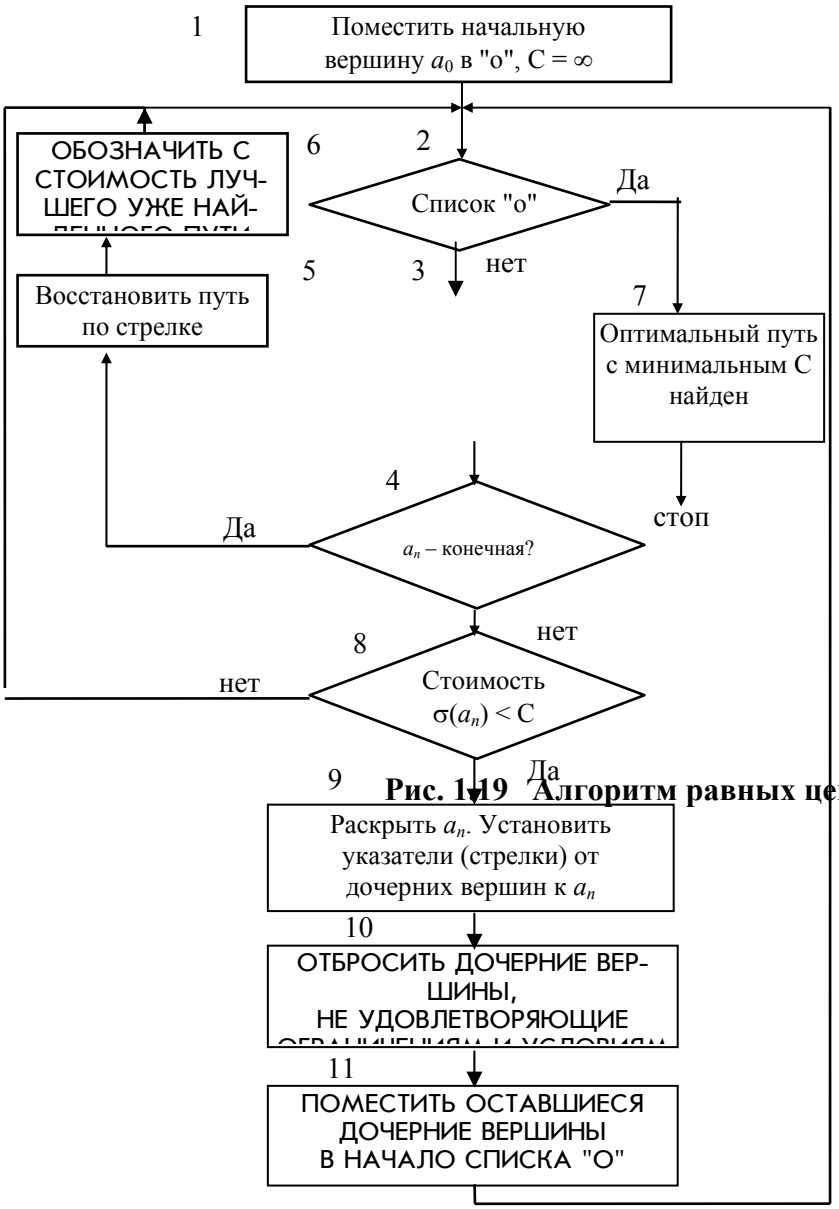


Рис. 1.19 Алгоритм равных цен

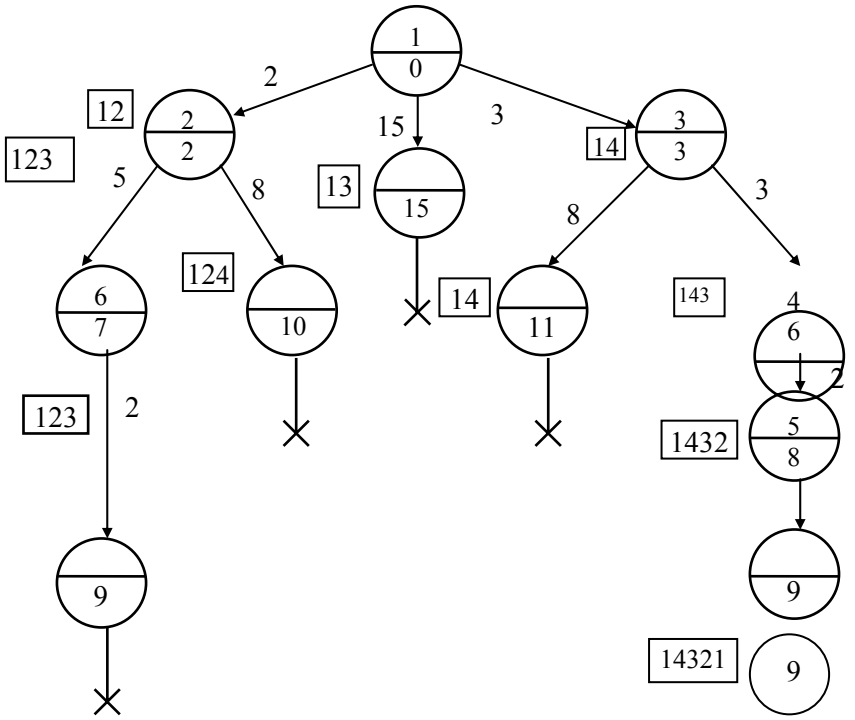


Рис. 1.20 Метод равных цен

В методе ветвей и границ каждая вершина a_n оценивается функцией $Q(a_n) = q(a_n) + \psi(a_n)$.

Эта функция содержит две составляющие. Функция $q(a_n)$ оценивает стоимость пути $\sigma(a_n)$ от начальной вершины a_0 до вершины a_n.

Функция $\psi(a_n)$ называется функцией прогноза, обещанием, оценкой. Эта функция прогнозирует, какая наилучшая (минимальная) стоимость пути от вершины a_n до конечной вершины.

Метод ветвей и границ на каждом шаге раскрывает ту вершину, для которой функция оценки $Q(a_n)$ – минимальна.

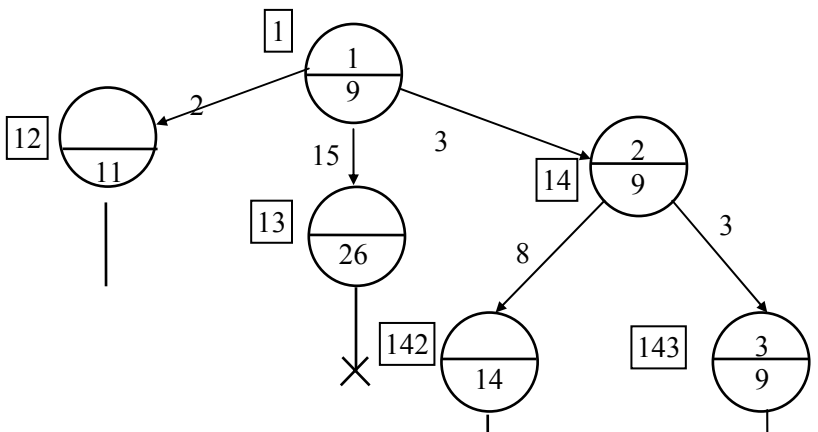
Выбор функции $\psi(a_n)$ – это всегда нестандартная, творческая задача. От правильного задания функции $\psi(a_n)$ зависит не только эффективность метода, т.е. число раскрытых вершин, но и сам результат. В отличие от вышерассмотренных методов в методе ветвей и границ оптимальный (в глобальном смысле) путь может быть потерян, если $\psi(a_n)$ выбрана неправильно.

Если допустить, что для каждой вершины a_n известна функция $\psi(a_n)$, которая точно позволяет назвать стоимость дальнейшего пути от a_n до конечной вершины при условии, что этот путь оптимален. Таким образом, $\psi(a_n)$ – максимально возможная стоимость при движении от точки a_n к конечной. Доказано, что в этом случае метод ветвей и границ будет более эффективен, так как найдет именно оптимальный путь при минимальном числе раскрытых вершин.

Рассмотренную ранее задачу коммивояжера решим методом ветвей и границ, считая, что откуда-то известен алгоритм $\psi^*(a_n)$ – точная функция прогноза.

Для первой точки a_0 (корня дерева) (рис. 1.21) можно записать $Q(a_0) = q(a_0) + \psi^*(a_0)$. Принимается, как и раньше, $q(a_0) = 0$. Функция $\psi^*(a_n)$ "знает" цену оптимального пути.

Раскрывается вершина , ее дочерними вершинами являются вершины , и . Потери $q(a_i)$ для этих вершин указаны на ребрах.



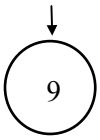


Рис. 1.21 Метод ветвей и границ

Функция $\psi^*(a_\iota)$ вычисляет оптимальный путь от соответствующей вершины до конечной. Используя граф (рис. 1.14), можно подсчитать, какие значения, как гипотезу, примет функция $\psi^*(a_i)$: $\psi^*($ 12 $) = 9$; $\psi^*($ 13 $) = 11$; $\psi^*($ 14 $) = 6$.

Таким образом, $Q($ 12 $) = 11$; $Q($ 13 $) = 26$ и, наконец, $Q($ 14 $) = 9$. Эти значения внесены в соответствующие вершины (рис. 1.21).

Наилучшей, согласно оценке Q, является вершина 14 . Раскрывая ее, получают вершины 142 и 143 .

Очевидно, что $Q($ 142 $) = 11$; $Q($ 143 $) = 6$.

С помощью графа полного перебора (рис. 1.14) устанавливают, что прогноз $\psi^*(a_\iota)$ был бы следующим: $\psi^*($ 142 $) = 1$; $\psi^*($ 143 $) = 3$. Таким образом, $Q($ 142 $) = 22$; $Q($ 143 $) = 9$, поэтому раскрытию подлежит вершина 143 и т.д.

Для нахождения оптимального пути методом ветвей и границ с идеальной (абсолютно точной) функцией прогноза потребовалось раскрыть всего 4 вершины, именно столько вершин, сколько их на оптимальном пути, исключая конечную.

Этот алгоритм выводит на оптимальный путь, так как он дает сразу точное знание того, какова же будет цепь того или иного пути.

Однако абсолютно точную функцию прогноза $\psi^*(a_i)$ найти невозможно. Вместо $\psi^*(a_i)$ используется какая-либо другая функция $\psi(a_i)$, являющаяся оценкой функции $\psi^*(a_i)$. Очевидно, чем ближе функция $\psi(a_i)$ к $\psi^*(a_i)$, тем меньше границ нужно перебрать, чтобы найти оптимальный путь. При неправильном выборе $\psi(a_i)$ может оказаться, что нашли и посчитали за оптимальный путь, который в действительности таковым не является.

Доказано, что если для всех a_i выполняется условие

$$\psi(a_i) \le \psi^*(a_i), \qquad (1.1)$$

то алгоритм метода ветвей и границ найдет оптимальное решение.

Если условие (1.1) не соблюдается, то алгоритм может пропустить оптимальный путь.

Если принять $\psi(a_i) \equiv 0$, то (1.1) соблюдается, и оптимальный путь будет найден. Но при этом алгоритм метода ветвей и границ превратится в алгоритм равных цен, для нахождения оптимального пути потребуется раскрыть слишком много вершин.

Следовательно, необходимо, чтобы прогноз $\psi(a_i)$, оставаясь меньше $\psi^*(a_i)$ для каждой a_i, давал как можно большую оценку. Другими словами, требуется, чтобы функция $\psi(a_i)$ была как можно ближе к нижней границе оценки функции $\psi^*(a_i)$.

В качестве примера решим рассмотренную уже ранее задачу комивояжера (рис. 1.14).

Оценочная функция $\psi(a_i)$ выбирается, как среднее расстояние от города a_i до тех городов, в которых комивояжер еще не побывал, включая и город 1, в который ему предстоит вернуться.

Итак, для первой вершины , как всегда, $q(\quad) = 0$, а обещание $\psi(\quad) = (0 + 2 + 15 + 3)/4 = 5$. Таким образом, $Q(\quad) = 5$ (рис. 1.22).

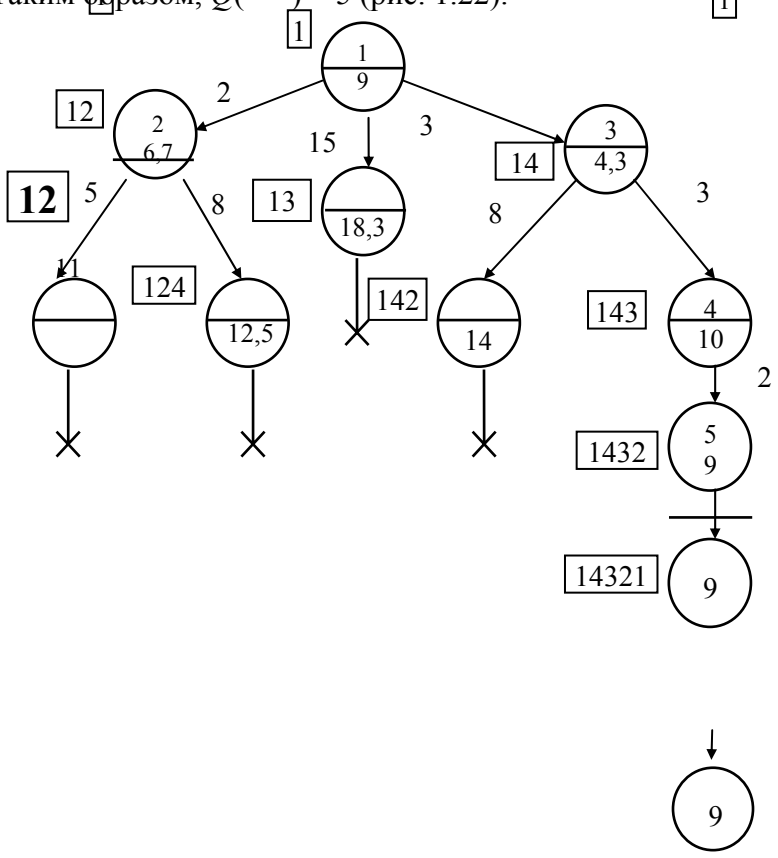


Рис. 1.22 Метод ветвей и границ с оценочной функцией среднего расстояния до непройденных городов

Для ее дочерних вершин , , расчеты $Q(a_\iota)$ сведены в табл. 1.2. Наименьшее значение Q соответствует вершине (рис. 1.22).

Оценки дочерних вершин приведены в табл. 1.2

1.2 Оценки вершин

Порядок раскрытия вершин	Вершина a_i	Пройденный путь $q(a_i)$	Города, которые необходимо посетить	Расстояние до города	$\psi(a_i)$	$Q(a_i)$

1	1	0	2 3 4 1	2 15 3 0	(2 + 15 + + 3 + 0)/4 = 5	5
2	12	2	3 4 1	5 8 1	(5 + 8 + + 1)/3 = 4,7	6,7
–	13	15	2 4 1	2 2 6	(2 + 2 + + 6)/13 = 3,3	18,3
3	14	3	2 3 1	8 3 2	(8 + 3 + 2)/3 = = 4,3	7,3
	123	7	4 1	2 6	(6 + 6)/2 = 4	11
	124	10	3 1	3 2	(3 + 2)/2 = = 2,5	12 5
	142	11	3 1	5 1	(5 + 1)/2 = 3	14
4	143	6	2 1	2 6	(2 + 6)/2 = 4	10
5	1432	8	1	1	1/1 = 1	9
–	1442	9	–	–	–	9

Следующей раскрывается вершина 14, так как она имеет наименьшую оценку Q = 7,3. Расчет оценок ее дочерних вершин и также приведен в табл. 1.2.

Далее раскрывается вершина 143, так как ее оценка Q = 10 наименьшая.

Затем раскрывается вершина 1432, последняя вершина заканчивает путь с результатом Q = 9. Этот результат окончательный: действительно все другие ветви имеют оценку, большую 9.

При нахождении оптимального пути потребовалось раскрыть всего 5 вершин (рассмотрено же 10 вершин). Это меньше, чем в любом из рассмотренных методов, исключая метод ветвей и границ с идеальной функцией прогноза $\psi^*(a_i)$.

Метод ветвей и границ находит широкое применение в самых различных задачах принятия решения и является одним из основных приемов построения алгоритмов искусственного интеллекта (при доказательстве теорем; играх в шахматы, шашки, карты; в экспертных системах, постановке диагноза, автоматических приемах обучения и воспитания и др.).

1.5 Поиск на графах

При поиске на графах применяются те же самые алгоритмы: перебор вширь, вглубь, метод равных цен, метод ветвей и границ, что и при поиске на деревьях. Однако, они усложняются проверкой: не находится ли "вновь" появляющаяся вершина уже в списке "о" или "з", т.е. не появлялась ли она уже раньше, при раскрытии предыдущих вершин.

Если она уже появлялась, т.е. содержится в списках "о" или "з", производится коррекция уже полученных результатов, приходится возвращаться к предыдущим вершинам, чего никогда не бывает при работе с деревьями, и корректировать, если нужно стрелку (указание материнской вершины), глубину вершины, ее оценку.

Для примера рассмотрим метод слепого поиска вширь. Пусть фрагмент графа имеет вид, изображенный на рис. 1. 23. В табл. 1.3 приведены этапы алгоритма, представленного на рис. 1.14, дополненного блоками проверки списков "о" и "з" и отбрасыванием тех вершин, которые уже есть в списке.

На первом этапе в список "о" перемещается вершина 1.

На втором этапе вершина 1 перемещается в список "з" и раскрывается, т.е. определяются ее дочерние вершины 2 и 3.

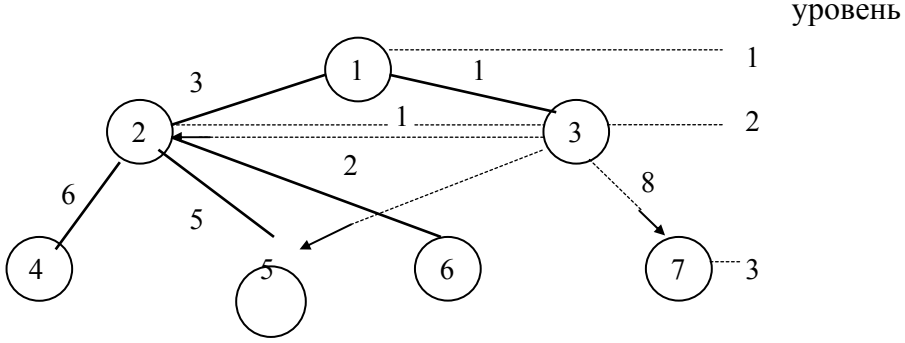


Рис. 1.23 Фрагмент графа полного перебора (перебора вширь)

На третьем этапе проверяется, не находятся ли вершины 2 и 3 уже в списке "о" или "з".

На четвертом этапе вершины $2^{(1)}$ и $3^{(1)}$, где индекс (1) указывает, что их материнской вершиной является вершина 1, помещаются в список "о".

Действия этапов 5, 6, 7 аналогичны действию этапов 1, 3, 4.

На этапе 8 раскрывается вершина $3^{(1)}$, первая из списка "о", ее дочерними вершинами являются $2^{(3)}$, $5^{(3)}$, $7^{(3)}$, где индекс (3) указывает, что их материнской вершиной является вершина 3.

Однако, на этапе 9 блок проверки обнаруживает, что вершина 5 уже находится в списке "о", при этом ее материнской вершиной является вершина 2. Кроме того, блок проверки находит, что цена дуги 2-5 больше, чем цена дуги 3-5. В связи с этим блок коррекции меняет для вершины 5 материнскую вершину. Вместо вершины $5^{(2)}$ в список "о" записывается вершина $5^{(3)}$.

Фрагмент графа (рис. 1.23) приобретает вид, представленный на рис. 1.24.

Вершины $2^{(3)}$, $7^{(3)}$ не содержатся в списке "о". Далее следует проверить, содержатся или нет вершины $2^{(3)}$, $7^{(3)}$ в списке "з".

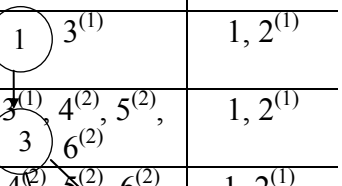
Оказывается, что вершина 2 находится в списке "з". Указатель (стрелка) направлен (табл. 1.3) на вершину 1. Затраты на дуге 1-2 при этом составляют 3 единицы. Так как на дуге 3-2 затраты меньше (всего лишь 1 единица), алгоритм изменяет направление стрелки вершины 2. Теперь материнской вершиной для нее становится вершина 3 (табл. 1.3).

Фрагмент графа приобретает вид, представленный на рис. 1.25, вместо графа на рис. 1.24.

1.3 Этапы раскрытия графа

Этап	Действие	Дочерние вершины	Вершины	
			Список "о"	Список "з"
1	Заполнение "о"	–	1	–
2	Раскрытие 1	$2^{(1)}$, $3^{(1)}$	–	1
3	Проверка	$2^{(1)}$, $3^{(1)}$	–	1
4	Заполнение "о"	–	$2^{(1)}$, $3^{(1)}$	1
5	Раскрытие	$4^{(2)}$, $5^{(2)}$,	$3^{(1)}$	1, $2^{(1)}$

	$2^{(1)}$	$6^{(2)}$		
6	Проверка	$4^{(2)}$, $5^{(2)}$, $6^{(2)}$	$3^{(1)}$	1, $2^{(1)}$
7	Заполнение "о"	–	$3^{(1)}$, $4^{(2)}$, $5^{(2)}$, $6^{(2)}$	1, $2^{(1)}$
8	Раскрытие $3^{(1)}$	$2^{(3)}$, $5^{(3)}$, $7^{(3)}$	$4^{(2)}$, $5^{(2)}$, $6^{(2)}$	1, $2^{(1)}$, $3^{(1)}$
9	Проверка	$2^{(3)}$, $5^{(3)*}$, $7^{(3)}$	$4^{(2)}$, $5^{(2)*}$, $6^{(2)}$	1, $2^{(1)}$, $3^{(1)}$
	Коррекция $5^{(2)}$	$2^{(3)}$, $7^{(3)}$	$4^{(2)}$, $5^{(3)*}$, $6^{(2)}$	1, $2^{(1)}$, $3^{(1)}$
	Проверка	$2^{(3)*}$, $7^{(3)}$	$4^{(2)}$, $5^{(3)}$, $6^{(2)}$	1, $2^{(1)*}$
	Коррекция $2^{(3)}$	$7^{(3)}$	$4^{(2)}$, $5^{(3)}$, $6^{(2)}$	1, $2^{(3)*}$
	Заполнение	–	$4^{(2)}$, $5^{(3)}$, $6^{(2)}$, $7^{(3)}$	1, $2^{(3)*}$



Таким образом, граф (рис. 1.23) преобразуется в дерево (рис. 1.25).

В этом случае, если бы стоимость дуг от вершины 3 к вершине 2 или вершине 5 оказалась бы больше уже рассчитанного, были бы оставлены те же материнские вершины, новые дуги были бы отброшены, и граф (рис. 1.23) снова превратился бы в дерево.

Рассмотрим теперь пример перебора в глубину на графе с ограниченной глубиной перебора.

Пусть максимальная глубина равна 4. Граф имеет вид, представленный на рис. 1.26. В этом случае порядок раскрытия вершин будет 1, 2, 5, 6 (табл. 1.4).

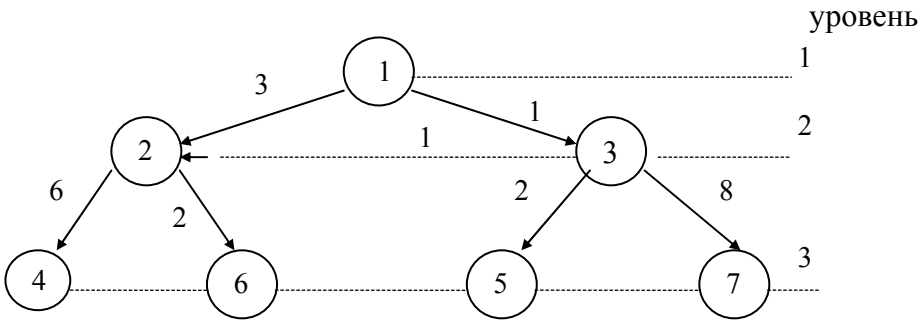


Рис. 1.24 Преобразование фрагмента графа (рис. 1.23)

Рис. 1.25 Преобразование фрагмента графа (рис. 1.23) в дерево

уровень

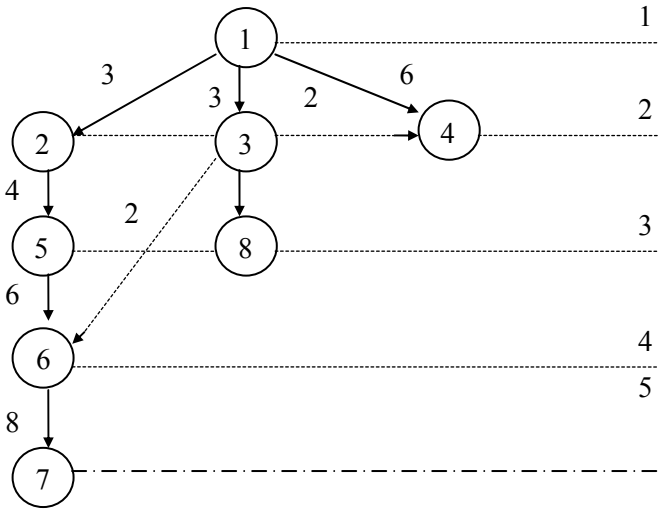


Рис. 1.26 Фрагмент графа, раскрываемого перебором в глубину

На этапе 14 должна была бы раскрываться вершина 7, как имеющая наибольшую глубину, но так как ее глубина превышает предельный уровень, равный 4, то она не раскрывается.

На этапе 14 вместо вершины 7 раскрывается самая глубокая вершина, уровень которой не превышает максимального. Это вершина, стоящая первой в списке "о", т.е. $3^{(1)}$. Ее дочерними вершинами, как это следует из рис. 1.26, будут вершины 6, 8, 4.

Следующим этапом после раскрытия вершины $3^{(1)}$ (табл. 1.4) будет проверка и коррекция списков "о" и "з".

1.4 Порядок раскрытия графа в глубину

Этап	Действие	Дочерние вершины	Вершины	
			Список "о"	Список "з"
1	Заполнение "о"	–	1	–
2	Раскрытие 1	$2^{(1)}$, $3^{(1)}$, $4^{(1)}$	–	1
3	Проверка	$2^{(1)}$, $3^{(1)}$, $4^{(1)}$	–	1
4	Заполнение "о"	–	$2^{(1)}$, $3^{(1)}$, $4^{(1)}$	1
5	Раскрытие $2^{(1)}$	$5^{(2)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$
6	Проверка	$5^{(2)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$
7	Заполнение "о"	–	$5^{(2)}$, $3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$
8	Раскрытие $5^{(2)}$	$6^{(5)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$, $5^{(2)}$
9	Проверка	$6^{(5)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$, $5^{(2)}$
10	Заполнение "о"	–	$6^{(5)}$, $3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$, $5^{(2)}$
11	Раскрытие $6^{(5)}$	$7^{(6)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$, $5^{(2)}$, $6^{(5)}$
12	Проверка	$7^{(6)}$	$3^{(1)}$, $4^{(1)}$	1, $2^{(1)}$, $5^{(2)}$, $6^{(5)}$
13	Заполнение	–	$7^{(6)}$, $3^{(1)}$,	1, $2^{(1)}$, $5^{(2)}$, $6^{(5)}$

	"о"		$4^{(1)}$	
14	Раскрытие $3^{(1)}$	$6^{(3)}, 8^{(3)}, 4^{(3)}$	$7^{(6)}, 4^{(1)}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)}, 3^{(1)}$
а	Проверка "о"	$6^{(3)}, 8^{(3)}, 4^{(3)*}$	$7^{(6)}, 4^{(1)*}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)}, 3^{(1)}$
б	Коррекция "о"	$6^{(3)}, 8^{(3)}$	$7^{(6)}, 4^{(3)*}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)}, 3^{(1)}$
15 в	Проверка "з"	$6^{(3)*}, 8^{(3)}$	$7^{(6)}, 4^{(3)}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)*}, 3^{(1)}$
г	Коррекция "з"	$8^{(3)}$	$7^{(6)}, 4^{(3)}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)}, 3^{(1)}$
д	Заполнение "о"	–	$7^{(6)}, 8^{(3)}, 4^{(3)}$	$1, 2^{(1)}, 5^{(2)}, 6^{(5)}, 3^{(1)}$

Вначале проверяется, не содержится ли какая-нибудь дочерняя вершина ($6^{(3)}$, $8^{(3)}$, $4^{(3)}$) в списке "о". Оказывается, что в этом списке есть вершина 4. Далее проверяется, что лучше: $4^{(3)}$ или $4^{(1)}$.

Двигаясь по стрелкам от дочерних вершин к материнским, получают путь 4-3-1 и второй путь 4-1, ясно, что эти пути создают цикл 4-3-1-4. Причем затраты на пути 4-3-1 меньше (равны), чем на пути 4-1, которые равны 6.

Поэтому выгодно считать предком вершины 4 не вершину 1, а вершину 3. В этом случае список "о" корректируется (этап 15, б табл. 1.4), и граф на рис. 1.26 трансформируется в граф, представленный на рис. 1.27.

Затем на этапе 15, в проверяется, не содержится ли одна из оставшихся вершин $6^{(3)}$ или $8^{(3)}$ в списке "з".

Оказывается, в списке "з" есть вершина $6^{(5)}$, а среди дочерних вершин $6^{(3)}$. Следуя от потомков к предкам, получают два пути, составляющие вместе цикл. Один путь 6-3-1, второй – 6-5-2-1. При этом путь 6-3-1 стоит 5 единиц, а путь 6-5-2-1 – 13 единиц. Путь 6-3-1 выгоднее и на этапе 15, г предок вершины 6 заменяется.

Вершина $8^{(3)}$ не находится ни в списке "о", ни в списке "з", и поэтому вносится в список "о". При этом граф, изображенный на рис. 1.27, преобразуется в граф, представленный на рис. 1.28.

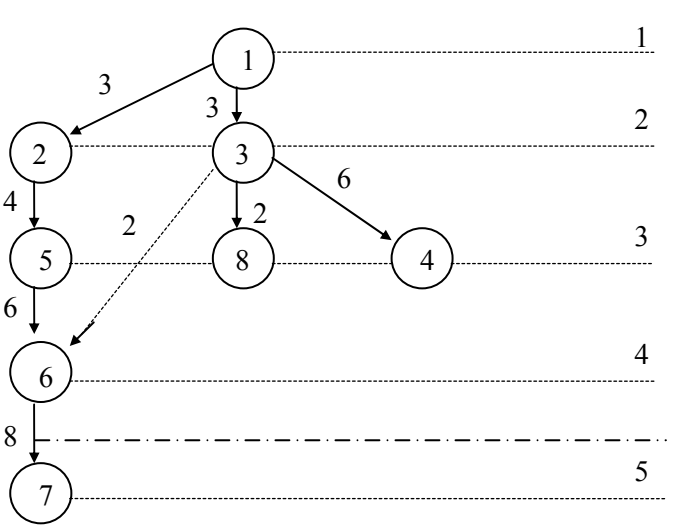


Рис. 1.27 Преобразованный граф (рис. 1.26)

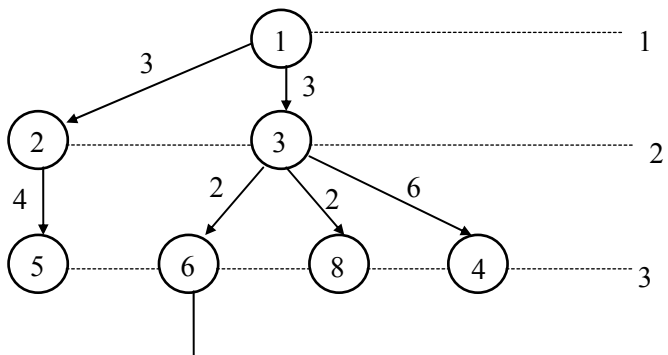


Рис. 1.28 Преобразование графа (рис. 1.26) в глубину

Необходимо отметить, что так как вершина $7^{(6)}$ находится за чертой допустимости (это показано в табл. 1.4 пунктирной линией), то вершина $8^{(3)}$ вносится в начало списка, но за пунктирную черту. Вершина 7 (рис. 1.26) стала теперь выше предельного уровня и подлежит раскрытию в первую очередь.

Аналогичными приемами осуществляются поиски оптимальных путей на графах и при других алгоритмах. При этом всегда в конечном итоге граф преобразуется в дерево, т.е. граф без циклов.

2 СЕТЕВЫЕ ЗАДАЧИ

2.1 Основные понятия

Сетью называется граф, каждая вершина которого представляет работу (событие), а дуги – порядок выполнения работ. При этом каждая работа (вершина) характеризуется временем (продолжительностью) выполнения работы.

Сеть удобно изображать в виде, представленном на рис. 2.1. Здесь кружочками обозначаются работы, цифры в кружочках обозначают продолжительность выполнения работы, стрелки – последовательность. Цифры $i = 1, 2, 3, 4, 5, 6$ обозначают номера (названия) работ.

Отношения между работами обозначаются знаками " ", " ". Так i j обозначает, что работа i предшествует работе j, т.е. работа i выполняется раньше, чем работа j; работа j выполняется позже работы i.

Знак " " обозначает непосредственное следование работ. Так i j обозначает, что работа i непосредственно предшествует работе j (рис. 2.2) или работа j непосредственно следует за работой i.

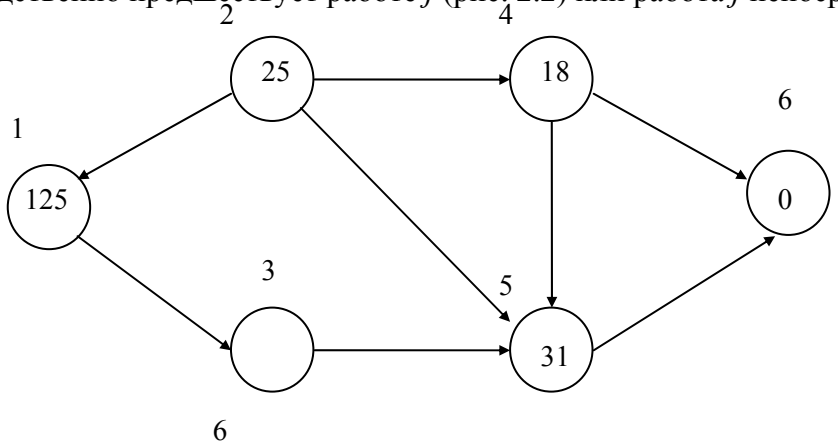


Рис. 2.1 Граф последовательности выполнения работ – сеть

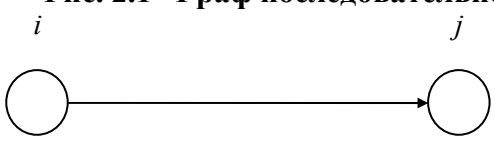


Рис. 2.2 Непосредственное предшествование работ

Для графа, изображенного на рис. 2.1, можно записать: 1 2, 1 3, 2 4, 2 5, 4 5, 4 6, 5 6.

Сеть считается заданной, если заданы:

а) все работы (вершины);

б) продолжительность работ ρ_i;

в) порядок следования работ.

Сеть (рис. 2.1) позволяет увидеть, что работа 2 может начаться только после того, как закончится работа 1 (так как 1 2). Работа 3 также начнется после окончания работы 1 (3 1), работа 5 может начаться только после того, как будут окончены работы 2, 3, 4 (5 3, 5 2, 5 4). Так как 6 4, 6 5, то работа 6 начнется после окончания работ 4 и 5.

Начальная вершина обозначается буквой *B*, конечная вершина – буквой *F*. Иногда продолжительности работ *B* и *F* бывают равными нулю. Это означает, что данная вершина является фиктивной: она просто обозначает "начало" или "конец" выполнения работ всего комплекса.

Необходимо заметить, что граф на рис. 2.1 можно изобразить как граф выполнения работ (рис. 2.3).

Однако изображение (рис. 2.1) удобнее, и именно оно используется в сетевых задачах.

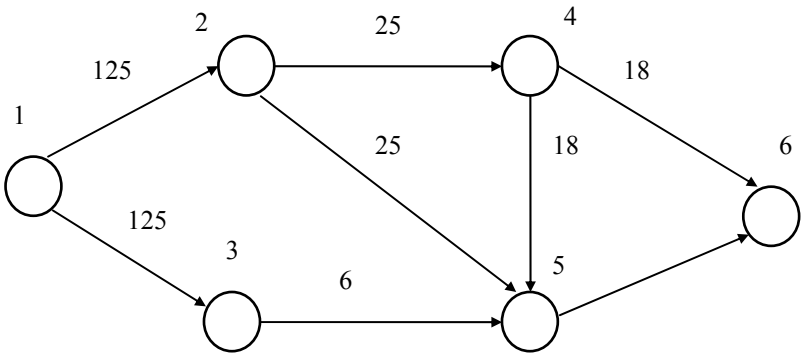


Рис. 2.3 Граф выполнения работ

Для решения сетевых задач можно применять методы и алгоритмы теории графов, однако, здесь графы не являются деревьями, и хотя имеется множество методов решения задач на графах, но они менее удобны. К тому же задачи исследования последовательности работ (сетевые задачи) специфичны и эта специфика сделала возможным использовать более простые алгоритмы исследования.

Сетевые задачи делятся на две группы:

1 Задачи исследования комплекса работ.

2 Задачи оптимизации комплекса работ.

2.2 Задача исследования комплекса работ

Задача исследования комплекса работ является важнейшей из двух типов сетевых задач.

Пусть работа 3 начинается лишь после выполнения работ 1 и 2 (рис. 2.4).

Предполагается, что работы 1 и 2 стартовали в одно и то же время. Однако, работа 1 выполняется 5 единиц времени ($\rho_1 = 5$), а работа 2 выполняется 10 единиц времени ($\rho_2 = 10$). Следовательно, начало выполнения работы 3 определяется временем выполнения работы 2. Если выполнение работы 2 запоздало на любую величину Δt, то на эту же величину запоздает начало выполнения работы 3 (рис. 2.5).

Работа 1 может опоздать на некоторое время (на время, равное 5 единиц) и это не вызовет изменения начала выполнения работы 3, так как все равно работа 3 должна "ждать" конца выполнения работы 2.

Говорят, что работа 2 является критической для работы 3. Этот факт отмечен на рис. 2.4 двойным кружком. Однако, для работы 4 работа 3 может не являться критической.

Путь от вершины *B* до вершины *F* (т.е. от начальной до конечной вершины), который целиком состоит из критических вершин, называется критическим путем.

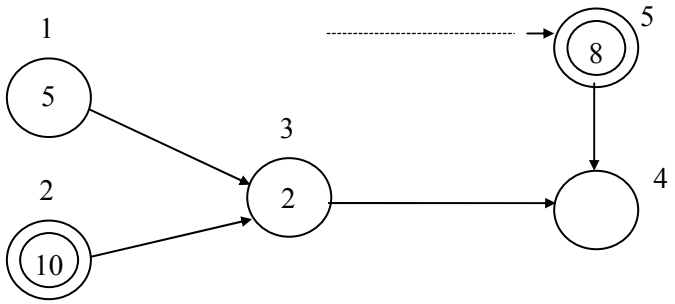


Рис. 2.4 Фрагмент сети

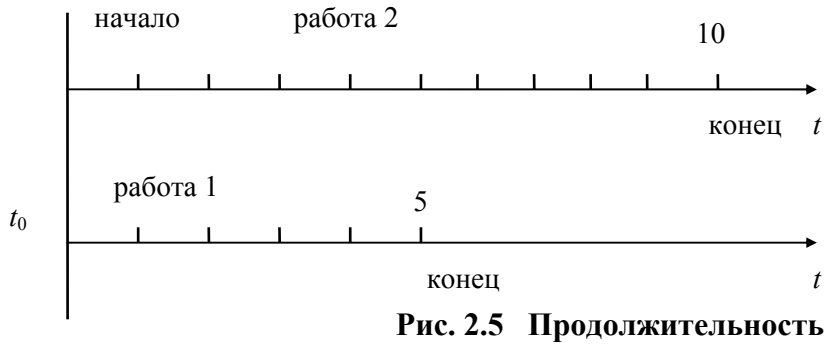


Рис. 2.5 Продолжительность работ

Критический путь определяет продолжительность T выполнения комплекса работ. Эта продолжительность равна сумме продолжительностей работ, составляющих критический путь.

Таким образом, критическим путем называется путь, на котором увеличение продолжительности любой работы приводит к увеличению продолжительности выполнения всех работ, другими словами, к увеличению времени завершения проведения всех работ.

Критической работой или критической операцией, соответственно, называется работа (операция), задержка выполнения которой приводит к задержке выполнения всего комплекса работ.

Критическая работа – вершина критического пути.

Для некритических операций, некритических работ существует определенный временной диапазон Δt, такой, что задержка выполнения работы в пределах этого диапазона не приводит к изменению продолжительности выполнения всего комплекса работ.

Очевидно, диапазон Δt_i i-й работы можно определить как

$$\Delta t_i = t_i^{\max} - t_i^{\min}, \qquad (2.1)$$

где $t_i^{\max}$ – максимально возможное время начала i-й работы; $t_i^{\min}$ – минимально возможное время начала i-й работы.

Можно получить рекуррентную формулу вычисления минимального и максимального возможного начала i-й работы.

Пусть известны минимально возможное значение начала $i-1$-й работы $t_{i-1}^{\min}$ (рис. 2.6) и максимально возможное время начала работы $i+1-t_{i+1}^{\max}$, такие, что $i-1$ i $i+1$.

Если известно минимально возможное время начала работы $i-1--t_{i-1}^{\min}$, то минимально возможное время начала работы i определяется по формуле

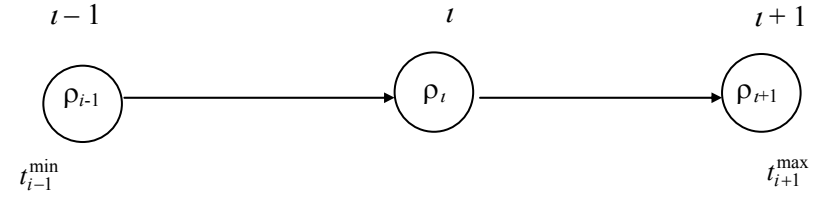


Рис. 2.6 Определение $t_1^{\min}$ и $t_i^{\max}$

$$t_i^{\min} = t_{i-1}^{\min} + \rho_{i-1}, \tag{2.2}$$

где ρ_{t-1} – продолжительность $t-1$-й работы.

Очевидно, что если в цепочке (рис. 2.7) известно $t_1^{\min}$ для первой работы, то рекуррентная формула (2.2) позволит определить $t_i^{\min}$ для каждой работы (т.е. для $t = 2, 3, ..., n$).

Более общий случай представлен на рис. 2.8. Работе 4 предшествует три работы $t = 1, 2, 3$, для которых известно минимальное время начало работ $t_j^{\min}$, $j = 1, 2, 3$.

Очевидно, $t_4^{\min}$ определяется как

$$t_4^{\min} = \max\left(t_1^{\min} + \rho_1, t_2^{\min} + \rho_2, t_1^{\min} + \rho_3\right). \tag{2.3}$$

Если для работы $t+1$ (рис. 2.6) известно максимально возможное время начала работы $t_{i+1}^{\max}$, то максимально возможное время $t_i^{\max}$ начала работы t (рис. 2.9) определяется по формуле

$$t_i^{\max} = t_{i+1}^{\max} - \rho_i. \tag{2.4}$$

Если для цепочки (рис. 2.7) известно максимально возможное время начала работы n (т.е. $t_n^{\max}$), то по соотношению (2.4) можно последовательно найти максимально возможные времена начала работ $t_i^{\max}$ для всех $t = n-1, n-2, ..., 2, 1$.

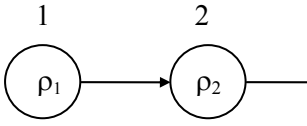


Рис. 2.7 Граф выполнения работ

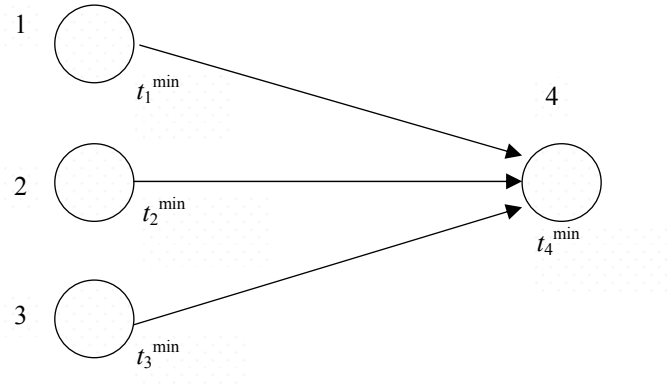


Рис. 2.8 Определение $t_4^{\min}$

Для более общего случая, изображенного на рис. 2.10, когда дочерними вершинами для вершины 4 являются вершины $t_i^{\max}$, $t = 1, 2, 3$ и для каждой из них известно максимально возможное время начала работ, $t = 1, 2, 3$, очевидно, чтобы работа $t = 1$ начиналась не позднее $t_1^{\max}$, начало работы 4 – $t_4^{\text{н}}$ должно быть не позднее

$$t_4' = t_1^{\max} - \rho_4. \tag{2.5}$$

Аналогичным образом можно определить начало работы 4 для выполнения предельных сроков начала работ 2 и 3 (рис. 2.11).

Максимальным временем начала работы 4 будет минимальное время из всех $t_4^{\text{н}}$:

$$t_4^{\max} = \min\left[t_1^{\max} - \rho_4, t_2^{\max} - \rho_4, t_3^{\max} - \rho_4\right] = \min\left[t_1^{\max}, t_2^{\max}, t_3^{\max}\right] - \rho_4. \quad (2.6)$$

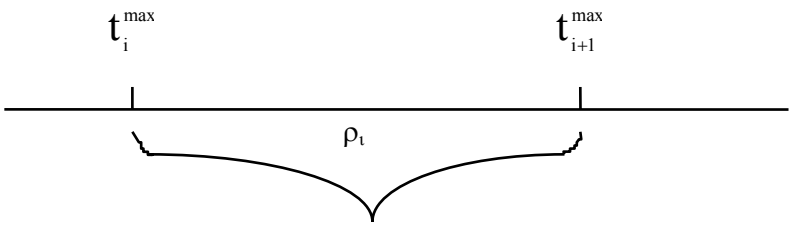


Рис. 2.9 Графическое изображение $t_i^{\max}$

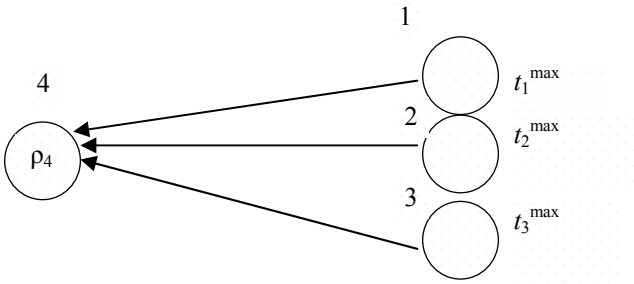


Рис. 2.10 Определение $t_4^{\max}$

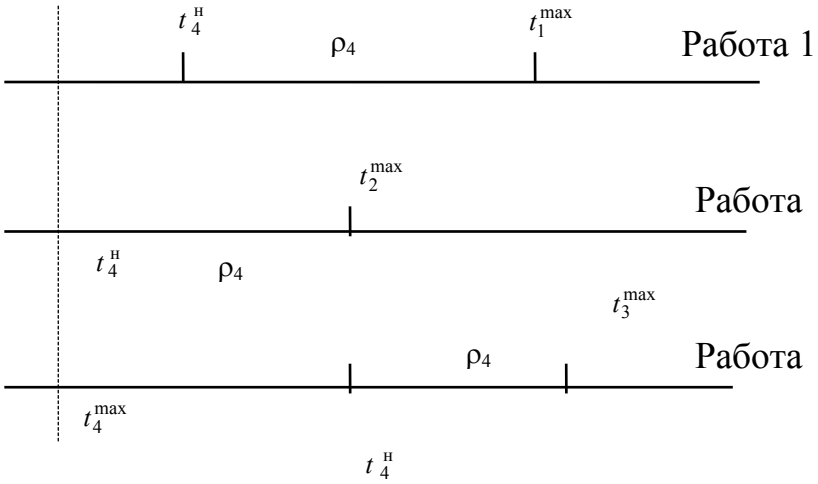


Рис. 2.11 Диаграмма определения $t_4^{\max}$

Действительно, если начало 4-й работы будет позже, чем $t_4^{\max}$, то работа 2 будет начата позже предельно допустимого срока $t_2^{\max}$.

Самый общий случай изображен на рис. 2.12.

Пусть множество вершин, непосредственно предшествующих вершине ρ_i, есть множество $N_i^{\text{вх}}$, а множество вершин, непосредственно следующих за вершиной i – $N_i^{\text{вых}}$. Аналогично формулам (2.3 – 2.6) для данного случая можно записать

$$\begin{aligned} t_i^{\min} &= \max_{j \in N_i^{\text{вх}}} \left[t_j^{\min} + \rho_i\right]; \\ t_i^{\max} &= \min_{j \in N_i^{\text{вых}}} \left[t_j^{\max}\right] - \rho_i. \end{aligned} \quad (2.7)$$

При этом величина $\Delta t_i = t_i^{\max} - t_i^{\min}$ является мерой критической работы и называется резервом времени операции *i*.

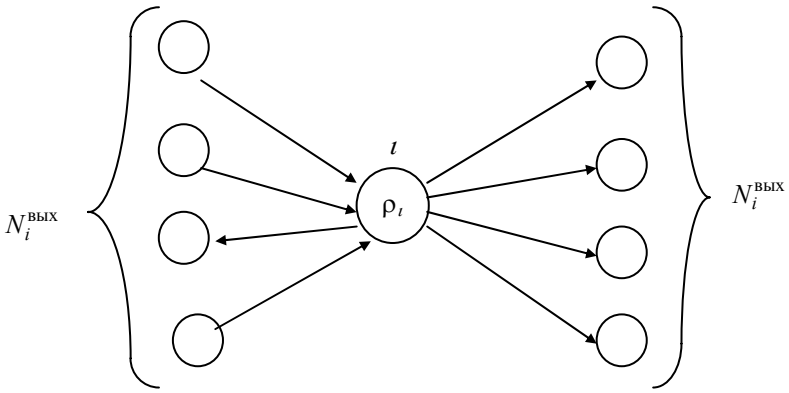


Рис. 2.12 Определение t_i^{min}, t_i^{max}

2.3 Цель исследования сетевой задачи

Целью задачи исследования комплекса работ или сетевой задачи является:

1 Нахождение минимальной продолжительности всего комплекса работ;

2 Определение критического пути;

3 Нахождение критических операций, т.е. операций, в которых t^{min} равно t^{max} и любая задержка в выполнении операций приводит к увеличению времени выполнения всего комплекса работ.

В такой постановке данная задача не является задачей исследования операции как таковой, это задача чистого анализа, в ней нет принятия решения. Однако, задача позволяет выявить узкие места в производстве комплекса работ, т.е. операции, которые нужно усовершенствовать, она служит основой для постановки задач принятия решения, задач оптимизации.

Подобные задачи встают буквально во всех областях как производственной, так и социально-общественной жизни. Любое строительство, научные разработки, конструирование и производство промышленных машин и изделий, военные и медицинские операции, экономические проекты, проведение культурных и общественных мероприятий и т.п., которые представляют сколько-нибудь сложный комплекс работ, требуют составления графика последовательности операций, его анализа с целью определения сроков проведения этапов выявления узких мест, определения критического пути, последовательности операций, продолжительность которых определяет срок выполнения комплекса работ, определения для остальных операций минимальных и максимальных сроков начала работ.

Множество всех вершин, у которых резерв времени $\Delta t(x) = 0$, составляет множество $R_{кр}$ критических операций (вершин), причем у каждой вершины путь заканчивается стрелкой. Эти вершины и связывающие их стрелки образуют критические пути.

Блок-схема описанной процедуры представлена на рис. 2.13. На рис. 2.14 представлен пример прохождения критических путей для сети. Вершины обозначены кружками. Порядок вершин ι показан рядом с кружками цифрами. В верхнем полукруге пишется продолжительность операции ρ_ι, в левой части нижнего полукруга $t_{min}(\iota)$, в правой части – $t_{max}(\iota)$.

В табл. 2.1 представлены результаты работы алгоритма (рис. 2.13) по нахождению критических путей сети (рис. 2.14).

Алгоритм формирует список (множество) "открыто" {0} и список (множество) "закрыто" {з}.

В блоке 1 эти множества обнуляются. 1 – 9 определяют $t_{min}(\iota)$ для всех $\iota = 0, 1, ..., n$.

В блоке 2 для нулевой вершины присваивается $t_{min} = 0$.

Блок 3 проверяет, есть ли среди множества вершин $N_{вых}(\iota)$ вершины, которые были помещены в список {0}.

Если такие вершины есть, то они удаляются из $N_{вых}(\iota)$. После этого блок 4 оставшиеся вершины переносит в конец списка {0}. Соответствующая вершина заносится в список {з}, если список {0} оказывается непустым.

Блок 5 передает управление блоку 6, который выбирает новые (первые) ι из списка {0}.

Блок 7 проверяет, все ли входные вершины $N_{вх}(\iota)$ уже были рассмотрены, т.е. содержатся ли они в списке {з}. Если это не так, то управление передается блокам 9 и 6 для выбора нового ι.

Так, при рассмотрении первой вершины 1 из списка {0} = 1, 2, 3 оказалось (строка 2 табл. 2.1), что вершины 2 из списка $N_{вх}(1)$ нет в списке {з}, т.е. она еще не рассмотрена. Поэтому рассмотрение вершины 1 невозможно, и блок 6 выбирает для рассмотрения вторую вершину – 2.

В этом случае, если все вершины из {0} перебраны и все они не подошли, блок 9 останавливает работу, так как в сети имеется цикл, которого, естественно, быть не должно.

В том случае, если все вершины $N_{вх}(\iota)$ содержатся в {з}, блок 8 определяет $t_{min}(\iota)$ по формуле (2.7), т.е. выбирает среди всех непосредственно предшествующих вершин $N_{вх}(\iota)$ ту, для которой время возможного начала работ максимально. Блок 8 устанавливает также стрелку (указатель), направленную на эту вершину. Далее управление снова передается блоку 3.

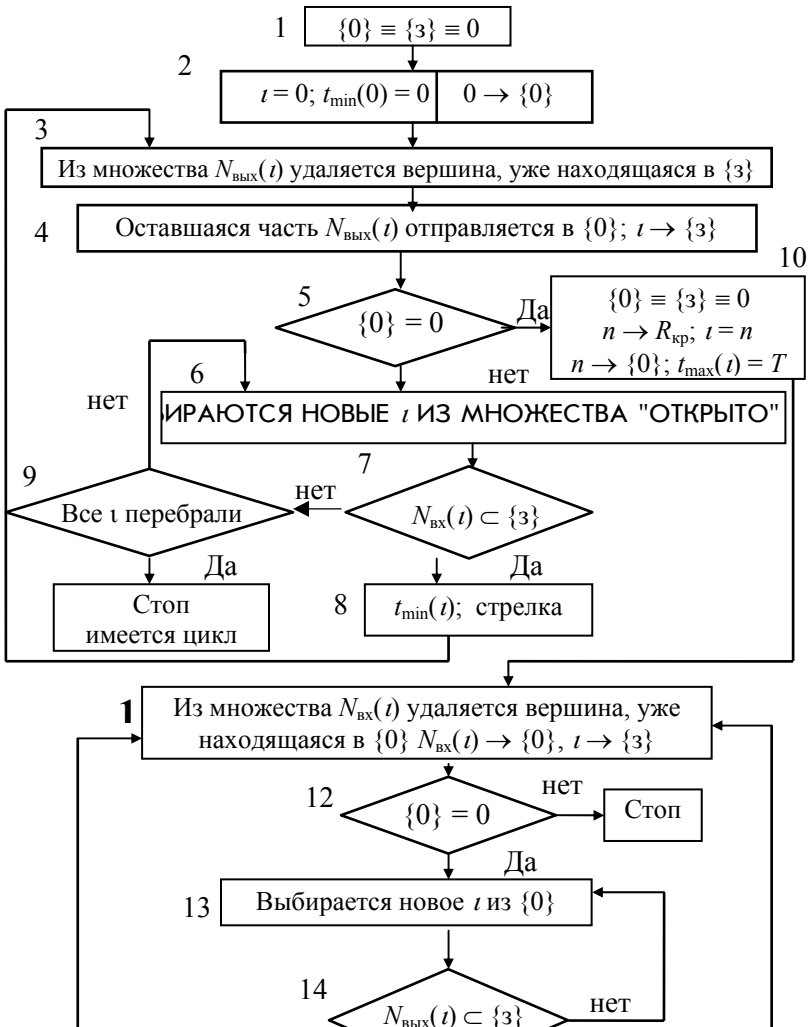


Рис. 2.13 Блок-схема поиска критического пути

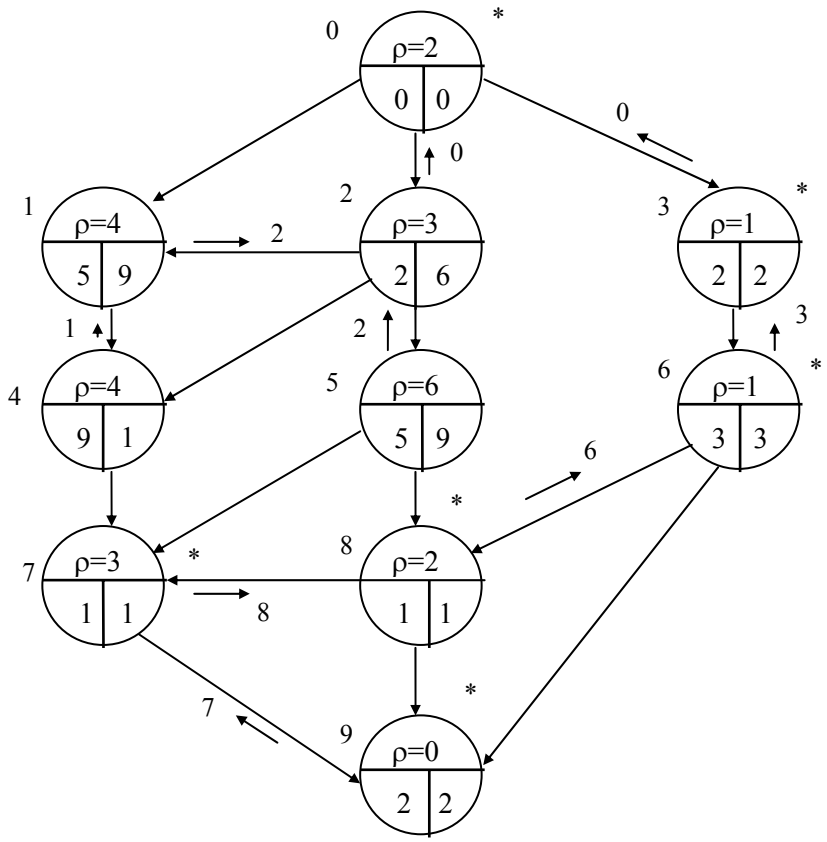


Рис. 2.14 Граф прохождения работ

Если при работе блока 5 оказалось, что множество {0} пусто, то это означает, что минимальные времена $t_{min}(\iota)$ начала работ найдены для всех вершин. При этом управление передается блокам 10 – 17, которые определяют максимальные времена $t_{max}(\iota)$ начала работ для каждой вершины.

Блок 10 обнуляет списки {0} и {з} и направляет последнюю вершину n в список критических вершин $\{R_{кр}\}$. Значение $t_{min}(n)$ для каждой вершины n принимается равным $t_{max}(n)$ для этой вершины $t_{min}(n) = T = t_{max}(n)$.

Работа блоков 11 – 17 аналогична работе блоков 3 – 8. В список критических вершин направляется та вершина, у которой минимально возможное время $t_{min}(\iota)$ начала работы совпадает с максимально возможным временем $t_{max}(\iota)$.

2.1 Результаты работы алгоритма поиска критического пути

Шаг	ι	$N_{вх}$	{з}	$N_{вых}$	{0}	$t_{min}(\iota)$	Стрелка на вершину
1	0	–	–	1, 2, 3	0	0	–
2	1	0, 2 =	0	4	1, = 2 ,3	не рассматривается	
3	2	0	0	4,5	1, 2, = 3	2	0
4	1	0, 2	0, 2	4	1, = 3, 4, 5	5	2
5	3	0	0, 2, 1	6	3, = 4, 5,	2	0

					6		
6	4	1,2	0, 2, 1, 3	7	4, 5, 6, = 7	9	1
7	5	2	0, 2, 1, 3, 4	7, 8	5, 6, 7 =	5	2
8	6	3	0, 2, 1, 3, 4, 5	8, 9	6, 7, 8 =	3	3
9	7	4, 5, 8 =	0, 2, 1, 3, 4, 5, 6	9	7, 8, 9 =	не рассматривается	
10	8	5, 6	0, 2, 1, 3, 4, 5, 6	9	7, 8, 9 =	15	6
11	7	4, 5, 8	0, 2, 1, 3, 4, 5, 6, 8	9	7, 9 =	17	8
12	9	7, 8, 6	0, 2, 1, 3, 4, 5, 6, 8, 7	–	9 =	20	7
13	–	–	0, 2, 1, 3, 4, 5, 6, 8, 7	–	пусто	–	–

Результаты работы алгоритма представлены в табл. 2.2.

Критический путь, отмеченный в табл. 2.2 и звездочками на рис. 2.14, будет 0-3-6-8-7-9. Этот путь (в общем случае несколько путей) восстанавливают по множеству $R_{кр}$ и стрелкам, связывающим каждую последующую вершину с предыдущей (начиная с нулевой вершины).

2.2 Нахождение критического пути

Шаг	i	$N^{вых}$	{з}	$N^{вх}$	{о}	t_i^{max}	$\Delta t(i)$	$R_{кр}$
1	9	–	–		9	20	0	9
2	6	8, 9 =	9	3	6, 7, 8	не рассматривается		
3	7	9	9	4, 5, 8	6, 7, 8, 4, 5	17	0	9, 7
4	6	8, 9 =	9, 7	3	6, 8, 4, 5	не рассматривается		
5	8	7, 9	9, 7	5,6	6, 8, 4, 5	15	0	9, 7, 8
6	6	8, 9	9, 7, 8	3	6, 8, 4, 3	3	0	9, 7, 8, 6
7	4	7	9, 7, 8, 6	1, 2	4, 5, 3, 1, 2	13	4	9, 7, 8, 6
8	5	7, 8	9,7,8,6, 4	2	5, 3, 1, 2	9	4	9,

								7, 8, 6
9	3	6	9, 7, 8, 6, 4, 5	0	3, 1, 2, 0	2	0	9, 7, 8, 6, 3
10	1	4	9, 7, 8, 6, 4, 5, 3	0	1, 2, 0	9	4	9, 7, 8, 6, 3
11	2	1, 4, 5	9, 7, 8, 6, 4, 5, 3, 1	0	2,0	6	4	9, 7, 8, 6, 3
12	0	1, 2, 3	9, 7, 8, 6, 4, 5, 3, 1, 2	–	0	0	0	9, 7, 8, 6, 3, 0
13	–	–	9, 7, 8, 6, 4, 5, 3, 1, 2	–	пусто	–	–	–

В описанном методе отсутствуют варьируемые параметры, а следовательно, нет места принятию какого-либо (в частности оптимального) решения.

Метод анализирует комплекс работ, вычисляет интервалы времени выполнения работ, определяет критические пути, т.е. операции, которые должны быть выполнены точно в срок и не имеют запаса времени. Этот анализ позволяет сделать вывод о допустимости или недопустимости предложенного плана выполнения работ. Однако, в такой постановке отсутствуют управления, позволяющие скорректировать план выполнения работ. Решение об изменении продолжительности работ критического пути, определяющем продолжительность выполнения всего комплекса работ, оставляется вышестоящей организации.

2.4 Задача оптимизации комплекса работ

Пусть теперь продолжительность работ ρ_i является переменной величиной, зависящей от затрат на эту работу. Обычно с увеличением вложений в работу (привлечением дополнительных механизмов, машин и т.п.) уменьшается продолжительность работы. Зависимость продолжительности работы ρ_i от ее стоимости C_i представлена на рис. 2.15.

Эта зависимость часто выражается в виде прямой линии:

$$\rho_i = \alpha_i - \beta_i C_i. \qquad (2.8)$$

Продолжительность работы изменяется в пределах $\underline{\rho_i} \le \rho_i \le \overline{\rho_i}$. При этом нижний предел $\underline{\rho_i}$ и верхний $\overline{\rho_i}$ соответствуют стоимостям $\underline{C_i} = \frac{\alpha_i}{\beta_i} - \frac{\underline{\rho_i}}{\beta_i}$ и $\overline{C_i} = \frac{\alpha_i}{\beta_i} - \frac{\overline{\rho_i}}{\beta_i}$.

Таким образом, стоимость изменяется в пределах $\underline{C_i} \le C_i \le \overline{C_i}$.

Из формулы (2.8) можно получить обратную зависимость стоимости от продолжительности: $C_i = \frac{\alpha_i}{\beta_i} - \frac{\rho_i}{\beta_i} = a_i - b_i\rho_i$, где $a_i = \alpha_i / \beta_i$, $b_i = 1/\beta_i$ – стоимость единицы уменьшения продолжительности работы.

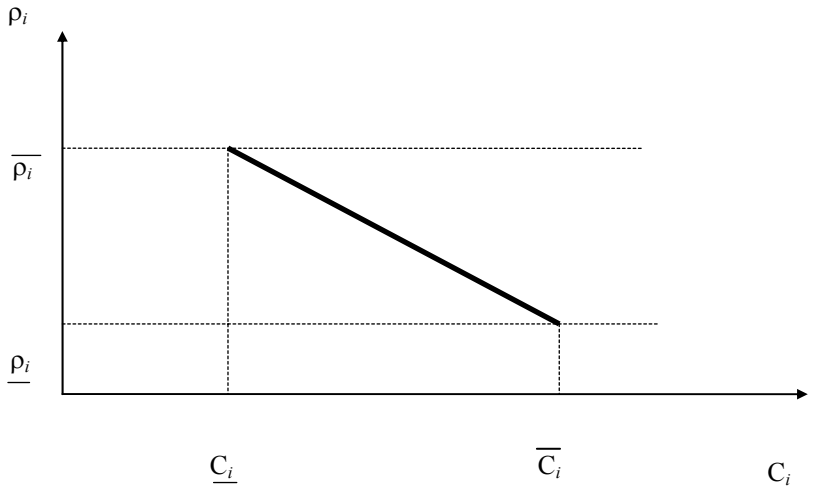


Рис. 2.15 Зависимость продолжительности работы ρ_i от стоимости этой работы C_i

Общая стоимость всех затрат составит:

$$Z = \sum_{i=1}^{n} C_i = \sum_{i=1}^{n} \left(\frac{\alpha_i}{\beta_i} - \frac{\rho_i}{\beta_i} \right) = \sum_{i=1}^{n} (a_i - b_i\rho_i),$$

где n – число работ (вершин).

Условия (2.7) определяют минимальное время начала каждой работы. Очевидно, что эти условия могут быть переписаны в следующей форме:

$$t(i) \le t(k) + \rho(k), \quad k \in N_{\text{вх}}(i), \quad i = 1, 2, ..., n. \tag{2.9}$$

Теперь можно поставить задачу выбора такой продолжительности работ, при которой будет минимальна стоимость работ, а время выполнения комплекса работ будет не больше заданного.

Так как $\sum_{i=1}^{n} \left(\alpha_i / \beta_i \right)$ не зависит от продолжительности работ ρ_i и является постоянным числом, минимизация Z соответствует максимизации функции

$$Q = \sum_{i=1}^{n} \rho_i / \beta_i = \sum_{i=1}^{n} b_i\rho_i. \tag{2.10}$$

При такой постановке задача оптимизации заключается в выборе ρ_i^* таких, при которых целевая функция (2.10) принимает максимальное значение $\sum_{i=1}^{n} b_i\rho_i \to \max$ и при этом удовлетворяются ограничения

$$t(i) \ge t(k) + \rho(k), \quad i = 1, 2, ..., n-1, \quad k \in N_i^{\text{вх}};$$
$$t(0) = 0; \quad t(n) \le T_з;$$
$$\underline{\rho_i} \le \rho_i \le \overline{\rho_i},$$

где $T_з$ – предельное время выполнения работ.

Эта задача относится к классу задач линейного программирования и может быть решена симплекс-методом.

В представленной постановке задачи $\rho(i)$ в некритических вершинах будет принимать максимально возможные значения, чтобы увеличить целевую функцию (2.10) (уменьшить стоимость работ).

Если последнее не желательно, то можно вместо целевой функции (2.10) использовать другую целевую функцию

$$Q = \sum_{i=1}^{n} b_i \rho_i - \sum_{i=1}^{n} M_i t_i, \qquad (2.11)$$

где M_i – некоторое большое число – штраф на время выполнения операций.

При большом t_i целевая функция уменьшается, поэтому в оптимальном решении будет компромисс между стоимостью комплекса работ и временем выполнения каждой операции. Подбором чисел M_i можно выделить те операции, в которых желательно уменьшить время их выполнения, и те, в которых желательно уменьшить стоимость работ.

Представленную задачу можно сформулировать и как задачу минимизации времени $t(k)$ выполнения всего комплекса работ при затратах не меньше заданных.

Для решения этих задач разработано много алгоритмов, использующих их особенности и обладающих большим быстродействием (например, алгоритм Келли), чем симплекс-метод.

В качестве примера рассмотрим эвристический метод, позволяющий как вручную, так и с помощью вычислительной техники найти достаточно близкое к оптимальному решение.

Этот метод рассматривается на примере графа (рис. 2.14).

В табл. 2.3 представлены для данного примера диапазоны изменения продолжительности работ [$\underline{\rho_i}$, $\overline{\rho_i}$], цены b_i единицы уменьшения продолжительности работы и интервалы изменения продолжительности $\Delta\rho_i^q$.

2.3 Результаты расчета графа эвристическим путем

Вершина	$\underline{\rho_i}$	$\overline{\rho_i}$	b_i	$\Delta\rho_i^q$
0	1	2	10	1
1	2	4	2	2
2	2	3	2	1
3	1	1	–	0
4	4	4	–	0
5	4	6	2	2
6	4	12	3	8
7	1	3	4	2
8	1	2	4	1
9	0	0	–	0
Продол-житель-ность	10	20		

Как видно из рис. 2.14, анализ графа проводится при продолжительностях работ, равных максимальным $\overline{\rho_i}$. При этом время выполнения всех работ составило $T = 20$.

При минимальных значениях продолжительностей работ $\underline{\rho_i}$ время выполнения комплекса работ согласно графу (рис. 2.16) в соответствии с данными табл. 2.3 составляет $T = 10$.

Значения допустимых изменений продолжительности работ приведены в табл. 2.3.

Алгоритм решения задачи оптимизации продолжительности работ сети представлен на рис. 2.17. Этот алгоритм позволяет построить зависимость увеличения стоимости С работ от уменьшения времени T выполнения работ, что позволяет найти необходимый компромисс среди этих величин.

Согласно блок-схеме алгоритма вначале выписываются все возможные пути графа от начальной вершины до конечной (табл. 2.4).

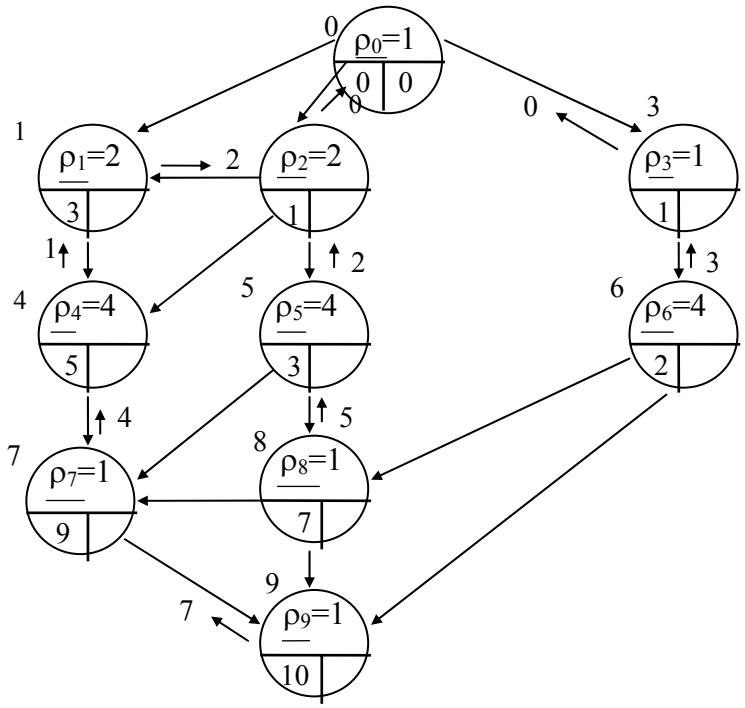


Рис. 2.16 Граф прохождения работ при минимальном времени выполнения комплекса работ

Для графа (рис. 2.14, 2.16) таких путей 9. Затем блок 2 (рис. 2.17) для каждой вершины каждого пути записывает в табл. 2.4 цены в b_i уменьшения на единицу стоимости продолжительностей ρ_i соответствующих работ. Таким образом, увеличение стоимости работ при изменении на $\Delta\rho_i$ продолжительности этих работ будет рассчитываться по формуле $\Delta Q = -\sum_{i=1}^{n} b_i \Delta\rho_i = -\sum_{i=1}^{n} b_i \left(\rho_i - \overline{\rho_i}\right)$, где $\overline{\rho_i}$ – максимальная продолжительность i-й работы.

Блок 3 заполняет строку **0** и столбец **0** табл. 2.4. При этом в столбец **0** заносятся минимальные времена выполнения соответствующих путей при максимальной продолжительности работ $\overline{\rho_i}$. В строку **0** заносятся диапазоны допустимых уменьшений $\Delta\rho_i^q$ продолжительности работ.

Затем блоки 4, 5 последовательно (итерационно) начинают корректировать (уменьшать) продолжительности работ, при этом возрастает их стоимость ΔQ.

В табл. 2.4 критическим путем, как это и указывалось раньше, является путь 7. Продолжительность прохождения этого пути $T = 20$ и она определяет продолжительность комплекса работ.

Очевидно, уменьшать надо продолжительность именно критического пути, так как уменьшение продолжительности других путей не влияет на продолжительность всего комплекса. Наоборот, в некоторых вариантах там, где это возможно, целесообразно увеличить продолжительность работ для некритических путей с целью уменьшения стоимости соответствующих работ.

Продолжительность критического пути 7 можно уменьшить с $T = 20$ до $T = 17$. Дальнейшее уменьшение нецелесообразно, так как путь 8 имеет время выполнения 17 и, вполне вероятно, будут два критических пути и их надо уменьшать уже совместно.

Уменьшить продолжительность пути 7 можно, уменьшив одну из следующих работ: 0, 6, 7 или 8. Продолжительность третьей работы уменьшать нельзя, так как допустимый диапазон изменен $\Delta\rho_3^q = 0$ (табл. 2.4). Из работ 0, 6, 7, 8 самой "дешевой" является шестая работа: увеличение единицы продолжительности стоит $b_i = 3$. Диапазон возможного изменения продолжительности $\Delta\rho_3^q = 8$.

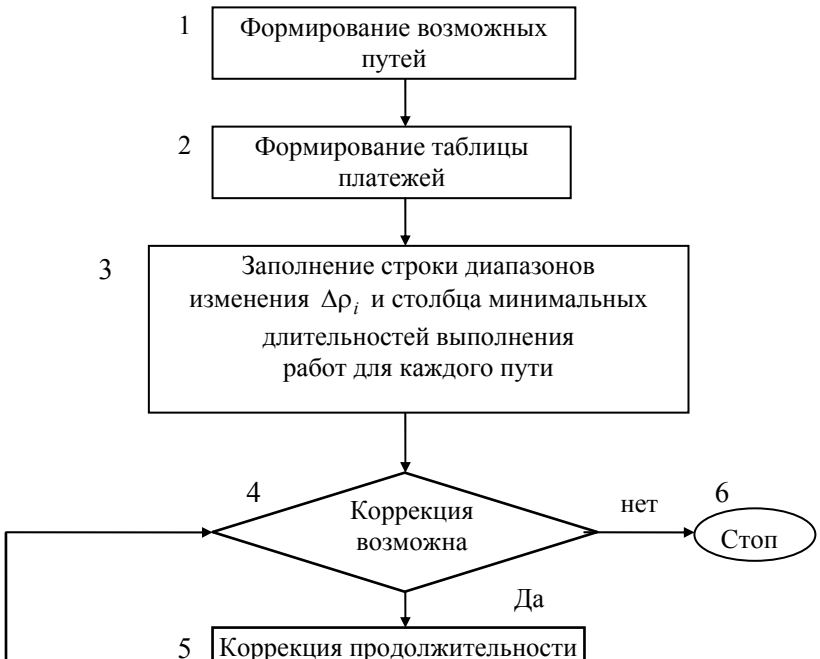


Рис. 2.17 Блок-схема алгоритма построения зависимости

Для того, чтобы продолжительность пути 7 стала равной 17, необходимо уменьшить продолжительность работы 6 на три единицы, так как работа 8 уже имеет продолжительность, равную 17. Однако работа 6 входит также в путь 8 и путь 9. Следовательно, продолжительность этих путей также уменьшится на 3 единицы. Результат этих преобразований представлен в табл. 2.4 в строке **1** и столбце **1**.

Для первой коррекции в табл. 2.4 под столбцом с номером **1** заносится приращение стоимости работ, которое в первой итерации составляет $\Delta Q_i = -b_3\Delta\rho_3 - \Delta Q_1 = 3\cdot 3 = 9.$

Критическим путем продолжает оставаться путь 7. Его еще можно уменьшить на единицу с тем, чтобы он сравнялся по продолжительности с путями 2 и 5. Результат этой коррекции представлен в строке и столбце с номерами **2**. При этом $\Delta Q_2 = \Delta Q_1 - b_3\Delta\rho_3;\ \Delta Q_2 = 9 + 3 = 12.$

Теперь критических путей три: 2, 5 и 7. Их надо уменьшать совместно на две единицы, так как дальше их продолжительность становится такой же, как и у пути 4 ($T = 14$).

Имеется несколько вариантов снижения на две единицы критических работ.

Вариант 1. Уменьшают самую "дешевую" работу 1 на две единицы (диапазон $\Delta\rho_1^q = 2$ позволяет это сделать). При этом путь 2 уменьшается на две единицы. Уменьшить работу 5 на две единицы $\left(\Delta\rho_5^q = 2\right)$ При этом

$$\Delta Q_3 = \Delta Q_2 - b_1\Delta\rho_1 - b_5\Delta\rho_5 - b_6\Delta\rho_6 - \Delta Q_3 = 12 + 2\cdot 2 + 2\cdot 2 + 3\cdot 2 = 26.$$

Вариант 2. Работа 7 входит во все критические пути и позволяет уменьшить их продолжительность на две единицы $\left(\Delta\rho_7^q = 2\right)$ Отсюда $\Delta Q_3 = \Delta Q_2 - b_7\Delta\rho_7 - \Delta Q_3 = 12 + 4\cdot 2 = 20.$

Таким образом, выбирается вариант 2. Результаты расчета представлены в строке и столбце **3**. Критическими остались те же пути (2, 5, 7).

Наилучшим вариантом сокращения продолжительности критических работ на единицу, т.е. до t = 13, будет сокращение работы 2 $\left(\Delta\rho_2^q = 1\right)$ и работы 6 $\left(\Delta\rho_6^q = 3\right)$ При этом $\Delta Q_4 = \Delta Q_3 - b_2\Delta\rho_2 - b_6 - \Delta\rho_6;$
$\Delta Q_4 = 20 + 2\cdot 1 + 3\cdot 1 = 25.$

Критическими путями становятся пути 2, 5, 7. Их необходимо уменьшить на единицу.

Вариант 1. Уменьшается на единицу работа 1 $\left(\Delta\rho_1^q = 2\right)$, работа 5 $\left(\Delta\rho_5^q = 2\right)$ и работа 6 $t_{j+1}^{\text{н}}$.

Тогда $\Delta Q_5 = \Delta Q_4 - b_1\Delta\rho_1 - b_5\Delta\rho_5 - b_6\Delta\rho_6 - \Delta Q_5 = 32.$

Вариант 2. Уменьшается на единицу работа 7 $\left(\Delta\rho_1^q = 1\right)$ и работа 8 $\left(\Delta\rho_8^q = 4\right) - \Delta Q_5 = 31.$

Выбирается вариант 2. Результаты расчета представлены в строке и столбце **5** табл. 2.4.

Критическими путями становятся пути 2, 5 и 7. Уменьшение их до T = 11 возможно лишь уменьшением на единицу работы 1, 5 и 6, диапазон изменения остальных работ равен нулю, за исключением нулевой, имеющей слишком высокую цену (b_1 = 10). При этом ΔQ_6 = 38. Результаты представлены в строке и столбце **6** табл. 2.4. Критические пути остались теми же, однако уменьшить их все на единицу воз-

можно лишь, изменив на единицу продолжительность первой работы, и тогда $\Delta Q_7 = 48$. Результаты представлены в строке и столбце **7** табл. 2.4.

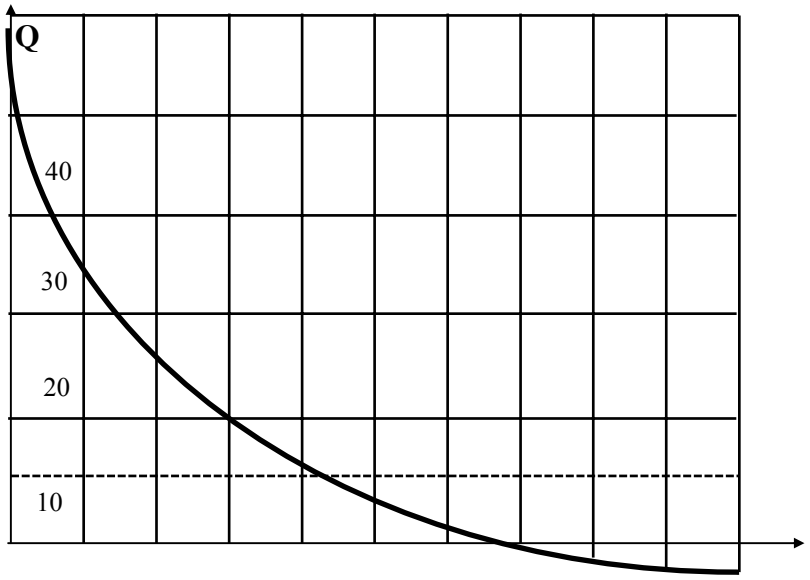


Рис. 2.18 Зависимость удорожания
от продолжительности комплекса работ

Возможность дальнейшей коррекции продолжительностей работ исчерпана, так как для критического пути 10 все работы имеют нулевой диапазон $\Delta\rho_i^q = 0$.

Результаты зависимости удорожания ΔQ выполнения комплекса работ от времени его выполнения T (табл. 2.4) представлены на рис. 2.18.

Критическим решением (рис. 2.18) является результат компромисса между увеличением стоимости комплекса работ и уменьшением его продолжительности.

При заданном ограничении на продолжительность комплекса работ $T \le T_{зад}$ график позволяет найти оптимальную стоимость комплекса работ ΔQ и соответствующую ей продолжительность ρ_i работ комплекса (табл. 2.4).

3 ТЕОРИЯ РАСПИСАНИЯ

Расписание – это последовательность выполнения работ. Составить расписание – это составить (запланировать) последовательность выполнения работ.

Наука, разрабатывающая методы составления оптимального, т.е. наилучшего с точки зрения какой-то целевой функции, расписания – порядка выполнения работ, называется теорией расписания.

3.1 Задачи теории расписания

Формально задачи теории расписаний можно представить следующим образом.

Пусть имеется всего две работы A и B. Выполнение работы A перед работой B ($A << B$) приводит к последствиям P_A, а ситуация $B << A$ – к последствиям P_B. В том случае, если оценка P_A предпочтительнее P_B, работа A должна выполняться раньше работы B.

Задачи теории расписаний возникают в самых различных областях составления расписаний аэропортов, вокзалов, порядка приема посетителей и больных, работы, производства и т.д.

Таким образом, в теории расписания предполагается, что

а) имеется набор работ (требований), которые должны быть выполнены;

б) порядок выполнения этих работ неизвестен, более того, именно порядком выполнения работ можно

варьировать произвольно, добиваясь наиболее эффективного расписания.

В теории расписания принята следующая терминология.

Работой (задачей) называется любое требование, которое надо выполнить.

Составляющие работы называются операциями. Таким образом, работа состоит из операций (из одной или многих). Операция – это неделимая подзадача, которую необходимо решить.

Устройство, которое может выполнить некоторую операцию, называется машиной. Множество машин, предназначенных для выполнения всех операций некоторого множества работ, называется цехом обслуживания.

Понятие машины в теории расписания употребляют в обобщенном смысле. Под этим термином понимают и собственно машины и механизмы, и людей, и вычислительные устройства, и различные организации, и т.д.

Совокупность работ и составляющих их операций, машин и дисциплин обслуживания (порядок назначения операций для выполнения их в тот или иной момент, правил назначения операций) называется системой обслуживания.

Составление расписания в самом общем случае заключается в том, что для каждой операции, для каждой работы необходимо:

а) назначить машину, на которой эта операция должна быть выполнена;

б) определить интервал времени, в течение которого эта операция осуществляется таким образом, что выполняются технологические условия и ограничения и целевая функция принимает бы минимальное (максимальное) значение.

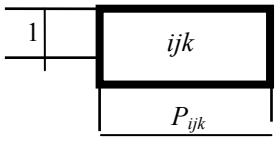


Рис. 3.1 Представление *j*-й операции *i*-й работы

Подобная задача сложна даже для формализации, а не только для ее решения. В практике теории расписания такая задача решалась численно очень немного раз.

Более простым является случай, когда операция каждой работы закреплена за машиной, т.е. каждая операция выполняется лишь на одной машине. В этом случае составление расписания заключается в установлении интервала времени, когда должна осуществляться каждая конкретная операция на соответствующей машине, при этом целевая функция должна быть минимальна (максимальна) и все технологические условия и ограничения выполнены.

В этом случае каждую операцию можно изобразить прямоугольником (рис. 3.1) с единичной высотой и длиной, равной продолжительности этой операции.

Индексы *ijk* на рис. 3.1 обозначают:

i – номер работы; *j* – номер операции *i*-й работы; *k* – номер машины, на которой выполняется *j*-я операция *i*-й работы.

Длительность P_{ijk} операции *j* работы *i*, выполненной на машине *k*, численно равна длине прямоугольника (рис. 3.1), но также и его площади, так как высота этого прямоугольника равна единице.

Множество всех работ может быть задано диаграммой (рис. 3.2)

Блоки, расположенные в одной строке (рис. 3.2), относятся к одной работе, первый индекс каждого блока в строке соответствует номеру работы. Второй индекс обозначает номер операции – это порядковый номер (1, 2, 3, ...). Третий индекс – это номер машины, на которой выполняется операция, поэтому этот номер беспорядочен.

Работа 1	112	124	133	141
Работа 2	213	224	233	
Работа 3	311	322	331	344
Работа 4	413	422	433	444

Рис. 3.2 Задание P_{ijk} для составления расписания

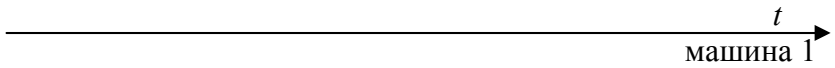
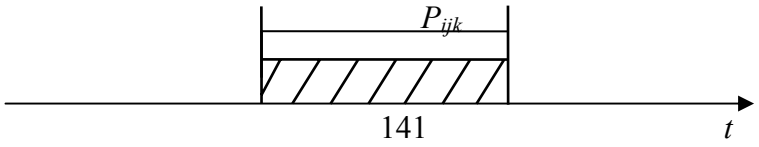


Рис. 3.3 Представление машины



машина 1

Рис. 3.4 Длительность операции

При составлении расписания обычно принимаются следующие допущения:

1 Машина под одним номером единственна, эта машина всегда исправна, т.е. в любой момент времени можно назначить выполнение операции, если эта операция действительно выполняется на данной машине и если в необходимый момент машина свободна.

Таким образом, машину можно представить себе как временную ось (рис. 3.3).

2 Прерывание операций отсутствует. Это означает, что на временной оси каждая операция обозначается отрезком [a, b], причем длительность операции при этом $P_{ijk} = b - a$. Таким образом, назначить выполнение операции на машине 1 в момент t – это значит поместить соответствующий блок из табл. 3.1 на ось t машины 1 (рис. 3.4).

В записи (141) цифра 1 – это и есть номер машины, на которой выполняется четвертая операция первой работы.

3 В каждый момент времени машина может выполнять не более одной работы. Это означает, что прямоугольники не могут налезать друг на друга, т.е. если есть две операции $[a_x, b_x]$ и $[a_y, b_y]$, то для соответствующих отрезков должно выполняться либо $[a_x, b_x] < [a_y, b_y]$, либо $[a_y, b_y] < [a_x, b_x]$.

4 Операции строго упорядочены и не пересекаются. Это очень важное допущение, оно означает что $j + 1$-я операция одной и той же работы i должна начаться только после того, как кончится j-я операция той же работы независимо от машины, на которой они выполняются (рис. 3.5)

Для операций j и $j + 1$ одной и той же работы справедливо $t^{\text{н}}_{j+1} \geq t^{\text{к}}_{j}$.

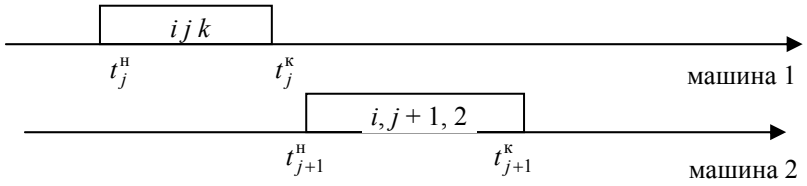


Рис. 3.5 Отношение между операциями

Сформулированные допущения не обязательны, есть задачи, в которых они не приемлемы, но эти задачи чрезвычайно сложны и почти не рассматривались.

Задача теории расписания считается заданной, если заданы число машин – m, число работ – n, число операций для каждой работы – q_j, $j = 1, 2, 3, ..., n$, время выполнения j-й операции i-й работы на k-й машине – P_{ijk}, причем множество P_{ijk} может быть задано в форме, представленной на рис. 3.2, момент готовности к выполнению i-й работы (момент наступления i-й работы) – r_i, плановый срок выполнения i-й работы – d_i.

В теории расписания обычно рассматривается случай, когда момент прихода всех работ единовременный, т.е. $r_1 = r_2 = r_3 = ...$. Таким образом, рассматривается статический случай: все работы известны, нужно их распределить по машинам. При этом можно считать $r_i \equiv 0$. Иногда, однако, r_i различны, т.е. необходимо рассматривать динамический процесс составления и коррекции расписания.

Однако характерными случаями разновременного прихода работ (r_i = var) занимается теория массового обслуживания.

В дальнейшем считается, что r_i = const, величина d_i – срок выполнения i-й работы.

Таким образом, продолжительность i-й работы $T_i^{\text{пр}}$ определяется как

$$T_i^{\text{пр}} = d_i - r_i \qquad (3.1)$$

или, если $r_i = 0$, то

$$T_i^{\text{пр}} = d_i . \qquad (3.2)$$

Предельная продолжительность – это интервал времени, в котором должны быть выполнены все операции *i*-й работы.

3.2 Диаграмма Гратта

Составить расписание можно с помощью диаграммы Гратта.

Пусть для трех машин задание будет в виде табл. 3.1.

3.1 Задание для трех машин

Работа 1	113	122	131
Работа 2	212	223	
Работа 3	312	321	333

Каждую машину изображают в виде временной оси, и операции табл. 3.1 разносят по машинам.

Теперь последний индекс операции соответствует своей машине и выполняется допущение о том, что машина в любой момент времени выполняет одну работу, прерывания операций отсутствуют. Но не выполнено допущение об упорядоченности и о непересекаемости операций.

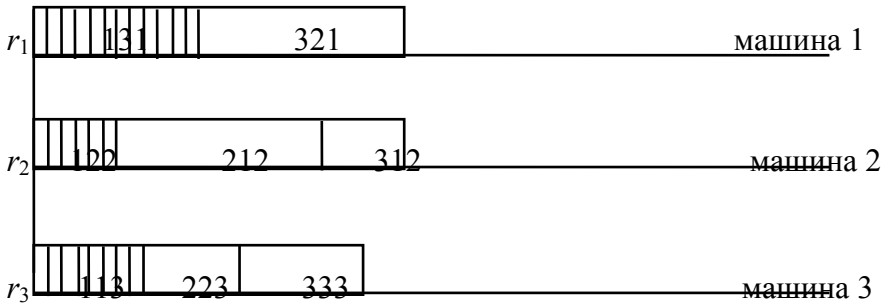


Рис. 3.6 Распределение работ по машинам

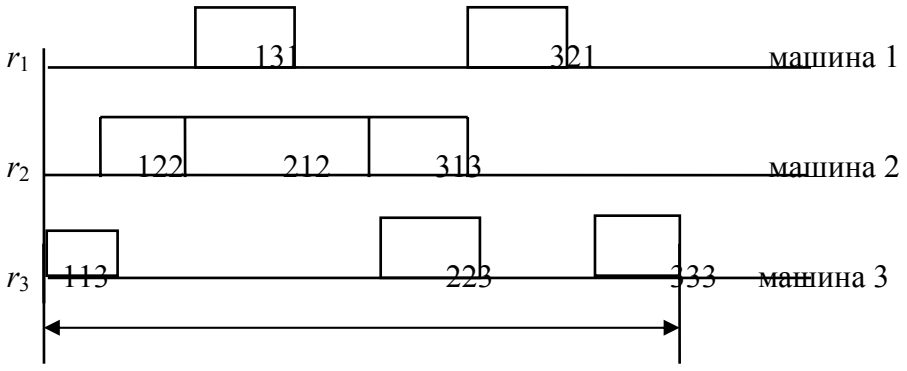
Так, для первой работы все операции выполняются совместно. Этого не может быть. Сначала должна быть окончена первая операция, и только потом начинаться вторая операция. Поэтому диаграмму (рис. 3.6) необходимо преобразовать к этим условиям.

Если спроектировать операции работы 1 на горизонтальную ось, то получается график (рис. 3.8) работы 1.

Из графика (рис. 3.8) видно, что операции идут последовательно, не пересекаясь. Это и означает, что рис. 3.7 есть расписание.

Однако это плохое расписание с точки зрения любого критерия, и его можно улучшить.

Методами оптимизации и занимается теория расписания.



$T^{\max}$

Рис. 3.7 Расписание работы машин

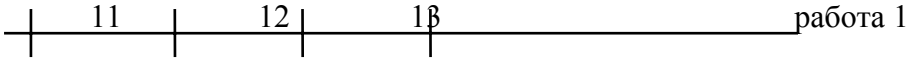


Рис. 3.8 График операций работы 1

3.2 Связь между операциями и машинами

Операции i-й работы	1	2	3	…	g_i
Номер (m_{ij}) машины i-й работы j-й операции	m_{i1}	m_{i2}	m_{i3}	…	m_{ig}
Длительность	P_{i1}	P_{i2}	P_{i3}	…	P_{ig}

Учитывая, что между операциями и машинами существует однозначная связь, часто трехиндексную P_{ijk} продолжительность работ, где первый индекс – номер работ, второй – номер операции, третий – номер машины, заменяют на двухиндексную P_{ij}, где первый индекс – номер работы, второй может быть либо номер машины, либо номер операции в зависимости от постановки задачи. Однако при этом задается связь между операциями и машинами в виде табл. 3.2

В табл. 3.2 m_{ij} – номер машины, на которой выполняется j-я операция i-й работы; P_{ij} – продолжительность выполнения j-й операции i-й работы.

Длительность выполнения всех операций i-й работы (длительность выполнения работы)

$$P_i = \sum_{j=1}^{g_i} P_{ij}. \tag{3.3}$$

Очевидно, что P_i – минимальная длительность выполнения i-й работы. Максимально возможная длительность выполнения (говорят "прохождения") работ определяется плановыми заданиями

$$T_i^{\text{пр}} = d_i - r_i, \tag{3.4}$$

где r_i – момент готовности i-й работы (поступление i-й работы); d_i – плановый срок выполнения i-й работы.

Между минимальной и максимально возможной длительностями естественно выполняется неравенство

$$P_i \le T_i^{\text{пр}}.$$

Величины $P_i, T_i^{\text{пр}}, d_i, r_i$ являются заданными величинами, известными до начала составления расписания.

В качестве примера рассмотрим фрагмент расписания для двух работ и двух машин (рис. 3.9)

Здесь в (i, j) первый индекс означает номер работы, второй – номер операции. Заштрихованные прямоугольники (рис. 3.9) – это интервалы времени, в течение которых выполняются операции.

Время начала j-й операции i-й работы обозначается $t_{ij}^{\text{н}}$, а время конца – $t_{ij}^{\text{к}}$. Время окончания всей i-й работы обозначается $t_i^{\text{к}}$. Момент готовности первой работы r_1, величина же W_{11} – время ожидания

первой операции первой работы (потерянное время). В момент времени $t_{11}^{к}$ первая операция заканчивается, однако вторая операция (1,2) этой работы начинается не сразу, а спустя время ожидания W_{12}.

Таким образом, общее время ожидания для первой работы составит

$$W_1 = W_{11} + W_{12}.$$

Начало второй работы также происходит не в момент r_2, а спустя некоторое время ожидания W_{21} для первой операции и W_{22} для второй операции. Общее время ожидания для второй работы составит

$$W_2 = W_{21} + W_{22}.$$

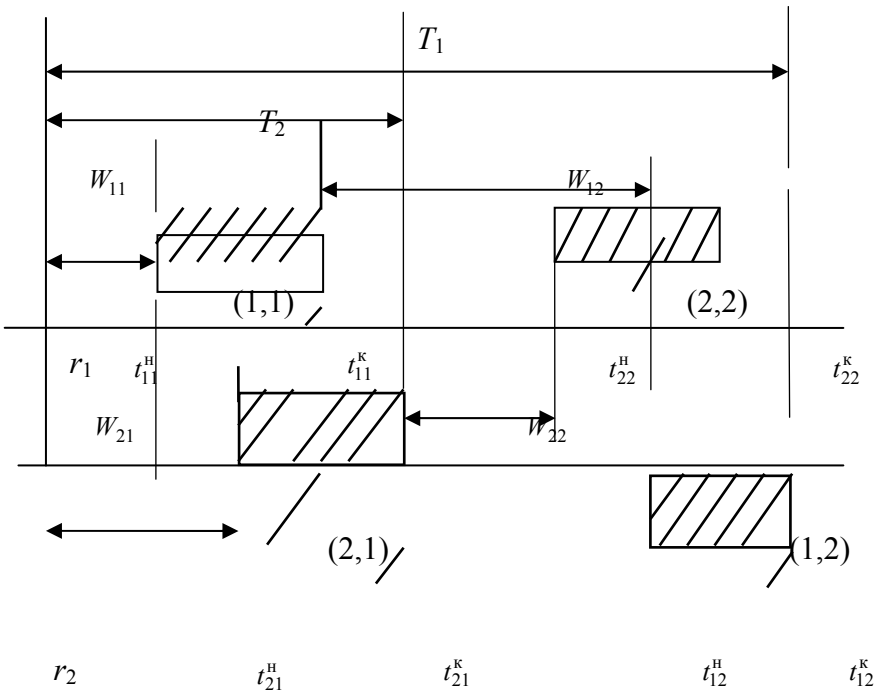


Рис. 3.9 Расписание для двух работ и двух машин

Таким образом, общее время ожидания для любой работы определяется по формуле

$$W_i = \sum_{j=1}^{g_i} W_{ij}. \qquad (3.5)$$

Пусть T_i – продолжительность (прохождение) *i*-й работы, тогда она может быть рассчитана как

$$T_i = t_i^{к} - r_i,$$

где $t_i^{к}$ – конец последней операции работы *i*. Очевидно, что

$$T_i = W_i + P_i. \qquad (3.6)$$

Решить задачу составления расписания – значит найти все W_{ij}, $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, g_i$. Причем начало и конец всех работ можно рассчитать как

$$t_{i1}^{к} = r_i W_{i1};$$

$$t_{i1}^{к} = t_{i1}^{к} + P_{i1};$$

$$t_{ij}^{н} = t_{i,j-1}^{к} + W_{ij}; \qquad (3.7)$$

$$t_{ij}^{к} = t_{ij}^{н} + P_{ij};$$

$$j = 2, \ldots, g_i.$$

Подведя итог, можно сказать, что расчет, необходимый для составления расписания, проводится по следующим формулам.

1 Момент окончания работы

$$t_i^{к} = r_i + P_i + W_i.$$

2 Длительность прохождения работы (интервал времени между началом и окончанием работы)

$$T_i = P_i + W_i$$

или

$$T_i = t_i^{к} - r_i\ .$$

3 Временное смещение работы

$$L_i = t_i^{к} - d_i$$

или

$$L_i = T_i - T_i^{пр},$$

где $T_i^{пр} = d_i - r_i.$

Таким образом, смещение работы L_i является разностью между реальным и плановым сроком окончания работы или разностью между реальной и плановой длительностями работ.

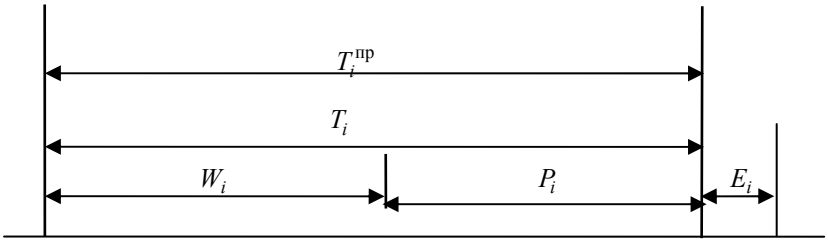
4 Запаздывание и опережение работы. Если величина L_i положительна, то она называется запаздыванием – z_i, если же L_i – отрицательна, то она называется опережением E_i, т.е.

$$z_i = \begin{cases} L_i,\ t_i^{к} \ge d_i,\ \text{т.е.}\ T_i \ge T_i^{пр}; \\ 0,\ t_i^{к} < d_i,\ \text{т.е.}\ T_i < T_i^{пр}; \end{cases} \qquad (3.8)$$

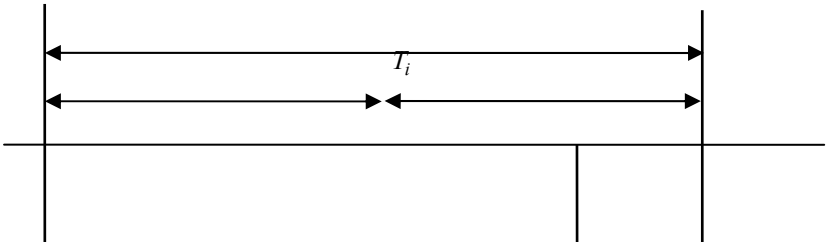
$$E_i = \begin{cases} |L_i|,\ t_i^{к} < d_i,\ \text{т.е.}\ T_i < T_i^{пр}; \\ 0,\ t_i^{к} \ge d_i,\ \text{т.е.}\ T_i \ge T_i^{пр} \end{cases} \qquad (3.9)$$

или $L_i = z_i - E_i.$

Условно эти соотношения можно изобразить так, как представлено на рис. 3.10



r_i $t_i^{к}$ d_i t



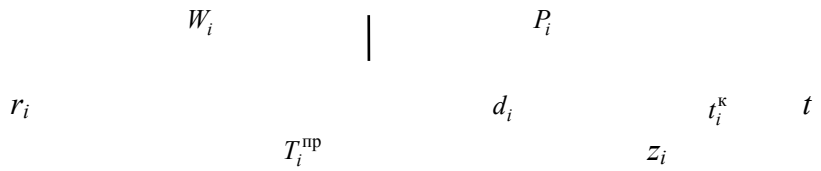


Рис. 3.10 Условные соотношения

3.3 Целевые функции задач теории расписания

3.3.1 СРЕДНИЕ ПОКАЗАТЕЛИ

Основными средними показателями, используемыми в теории расписания являются следующие:

– средняя длительность ожидания

$$\overline{W} = \frac{1}{n}\sum_{i=1}^{n} W_i; \tag{3.10}$$

– среднее значение времени окончания работ

$$\bar{t}^{к} = \frac{1}{n}\sum_{i=1}^{n} t_i^{к}\ ; \tag{3.11}$$

– среднее значение длительности прохождения работ

$$\overline{T} = \frac{1}{n}\sum_{i=1}^{n} T_i; \tag{3.12}$$

– среднее временное смещение

$$\overline{L} = \frac{1}{n}\sum_{i=1}^{n} L_i\ ; \tag{3.13}$$

– среднее запаздывание

$$\bar{z} = \frac{1}{n}\sum_{i=1}^{n} z_i. \tag{3.14}$$

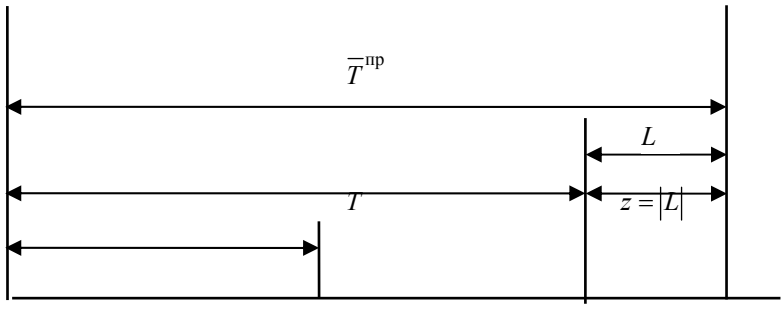
Между средними показателями существует однозначное соответствие

$$\overline{L} = \bar{t}^{к} - \overline{d} = \overline{T} - \overline{T}^{пр};$$
$$\bar{t}^{к} = \bar{r} + \overline{p} + \overline{W};$$

так как $\overline{d} = \frac{1}{n}\sum_{i=1}^{n} d_i,\ \overline{p} = \frac{1}{n}\sum_{i=1}^{n} p_i,\ \bar{r} = \frac{1}{n}\sum_{i=1}^{n} r_i,\ \overline{T}^{пр} = \frac{1}{n}\sum_{i=1}^{n} T_i^{пр}$ относятся к исходным данным и не зависят от расписания, величины $\overline{L},\ \bar{t}^{к},\ \overline{T},\ \overline{W}$ тождественны.

Между средними показателями существует связь, как и между индивидуальными показателями. Эта связь показана на рис. 3.11.

Средний момент окончания работ $\bar{t}^{к}$ равен среднему времени прохождения $\overline{T}$, так как $\overline{T} = \bar{t}^{к} - \bar{r}$, а $\overline{T}^{пр} = \overline{d}$.



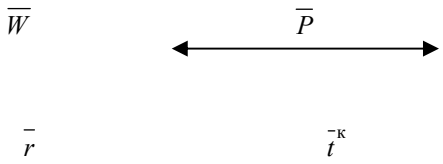


Рис. 3.11 Соотношение между средними величинами

Очевидно, вследствие связей оптимизация расписания по одному из средних показателей приводит к оптимизации по всем средним показателям.

Среднее смещение и среднее запаздывание связаны соотношением

$$\overline{L} = \bar{z} - \overline{E}\,.$$

Таким образом, максимум L совпадает с максимумом $\bar{z}$ только в случае, если $L_i > 0$ при всех $i = 1, 2, \ldots, n$.

3.3.2 МАКСИМАЛЬНЫЕ КРИТЕРИИ

К максимальным критериям относятся:

– максимальное значение длительности прохождения работ

$$T^{\max} = \max_i T_i; \tag{3.15}$$

– максимальное значение момента времени окончания работ

$$t^{\max} = \max_i t_i^{\text{к}}; \tag{3.16}$$

– максимальное временное смещение

$$L^{\max} = \max_i L_i; \tag{3.17}$$

– максимальное запаздывание

$$z^{\max} = \max_i z_i\,; \tag{3.18}$$

3.3.3 НЕРЕГУЛЯРНЫЕ ЦЕЛЕВЫЕ ФУНКЦИИ

К нерегулярным целевым функциям, используемым в теории расписания, относятся:

– коэффициент использования U, который рассчитывается по формуле

$$U = \frac{\sum_{i=1}^{n} P_i}{mT^{\max}}. \tag{3.19}$$

Соотношение $\sum_{i=1}^{n} P_i / m$ определяет среднюю продолжительность по машинам, которая известна, поэтому коэффициент использования определяется максимальной длительностью $T^{\max}$ продолжения работ;

– среднее число работ в системе

$$\overline{N} = \frac{1}{t^{\max}} \int_0^{t^{\max}} N(t)dt, \qquad (3.20)$$

где $N(t)$ – число работ в системе в момент времени t.

На рис. 3.12 представлен ранжированный график количества работ в системе.

Очевидно

$$\overline{N} = \frac{1}{t^{\max}} \int_0^{t^{\max}} N(t)dt = \frac{1}{t^{\max}} \left[nT_1 + (n-1)(T_2 - T_1) + \ldots + 2(T_{n-1} - T_{n-2}) + T_n \right].$$

После преобразования получают

$$\overline{N} = \frac{\sum_{i=1}^{n} T_i}{t^{\max}} = \frac{n\overline{T}}{t^{\max}}. \qquad (3.21)$$

Таким образом $\frac{\overline{N}}{n} = \frac{\overline{T}}{t^{\max}}$, т.е. отношение среднего числа работ $\overline{N}$ в системе к максимальному равно отношению средней длительности прохождения работ в системе к максимальной.

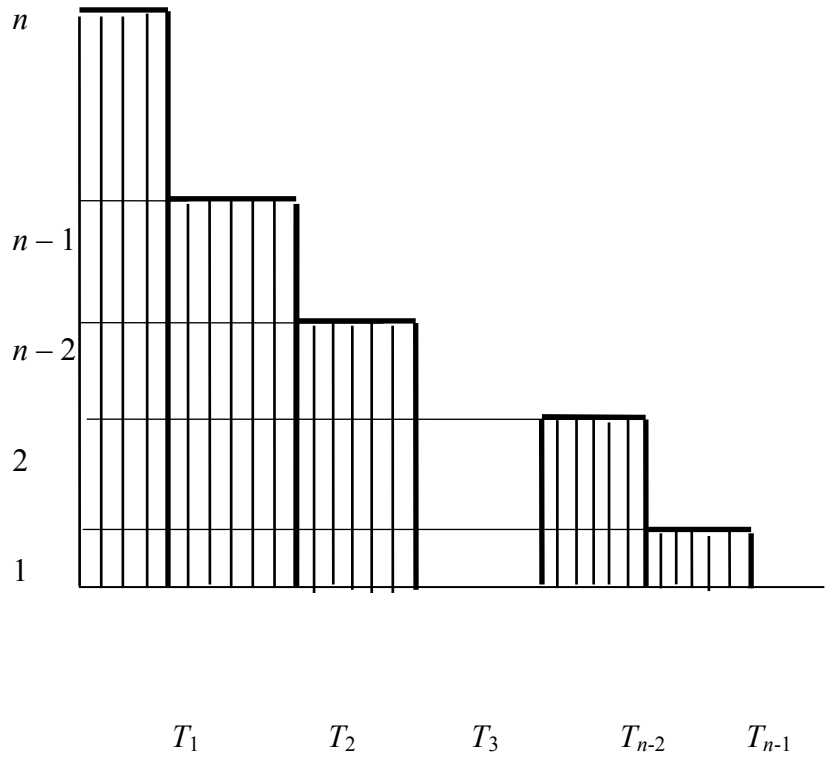


Рис. 3.12 График изменения числа работ

3.3.4 ЭКОНОМИЧЕСКИЕ КРИТЕРИИ

К экономическим критериям относятся:

– стоимость эксплуатации машин;

– стоимость хранения запасов для производства работ;

– стоимость запаздывания z;

– стоимость простоев машин;

Уменьшить стоимость эксплуатации машин – это значит составить расписание, в котором машина за одно и то же время выполняет большую работу, чем при другом расписании, или выполняет ту же работу меньшим числом устройств. Это приводит либо к увеличению валовой продукции, либо к сокращению парка машин. Косвенным показателем этой целевой функции является коэффициент использования.

Стоимость хранения запасов пропорциональна объему незавершенных работ. Косвенным показателем здесь является средняя длительность продолжения работ.

Запаздывание при выполнении работ влечет штрафы, срыв ритмичности производства, увеличение простоев оборудования и т.п. Экономическое выражение ущерба от запаздывания весьма сложно. Косвенным показателем ущерба от запаздывания является сама величина запаздывания.

В связи с тем, что экономические критерии вычисляются не просто, обычно при решении задач теории расписания используют регулярные и нерегулярные целевые функции.

Улучшения всех абсолютно критериев можно добиться, сдвигая влево все операции, пока это возможно, т.е. пока не мешает предстоящая операция на этой машине, и пока операция не "налезает" на предыдущую операцию этой же работы, выполняемой на другой машине.

На примере расписания на рис. 3.7 преобразование "сдвиг влево" приводит к расписанию, представленному на рис. 3.8 .

Расписание, в котором нельзя сдвинуть ни одной операции влево, называется квазикомпактным (рис. 3.13).

Операцию (312) можно перенести на свободное место перед операцией (122), а операцию (321) поставить перед операцией (131) (рис. 3.14)

Расписание, когда ни одну операцию нельзя ни перевести влево, ни передвинуть влево, называется компактным расписанием (рис. 3.14).

Буквально по всем разумным критериям оптимальное расписание нужно искать среди множества компактных расписаний.

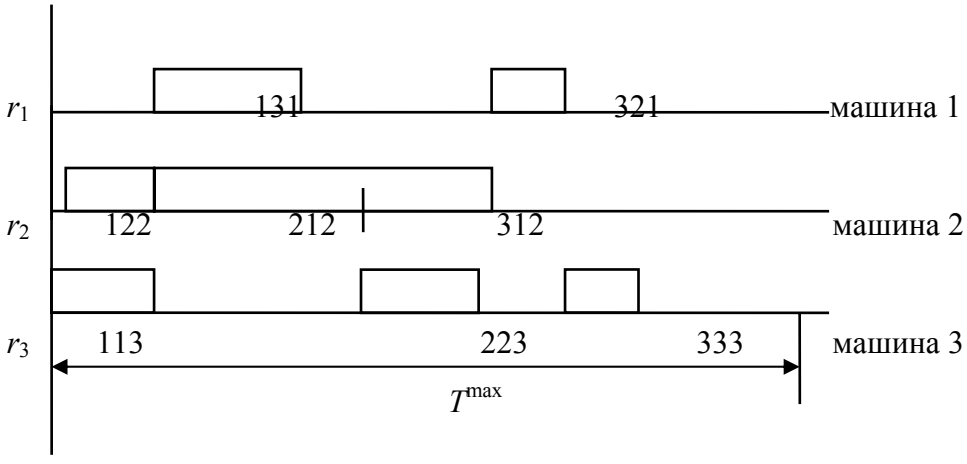
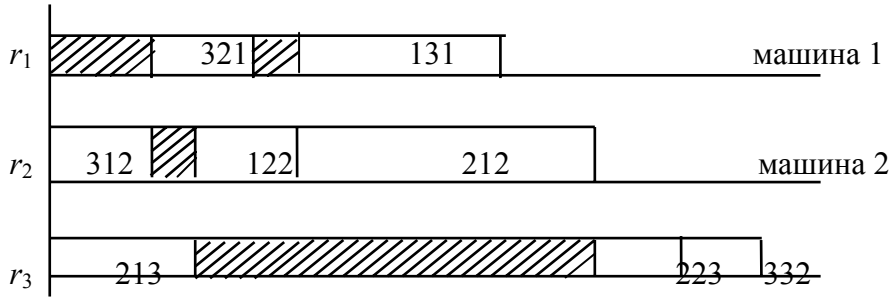


Рис. 3.13 Квазикомпактное расписание



машина 3

Рис. 3.14 Компактное расписание

3.4 Частные задачи теории расписания

3.4.1 УПОРЯДОЧЕНИЕ ЧИСЛА РАБОТ НА ОДНОЙ МАШИНЕ

Пусть все работы поступают одновременно, т.е. без уменьшения общности можно считать, что $r_i = 0$ для всех i; выполнение работ происходит либо без перенастройки машин, либо настройка не зависит от порядка выполнения работ.

Такая задача имеет большое практическое применение. Так, на химических, фармацевтических, пищевых производствах часто продукцию производят на одном оборудовании. То же относится к тяжелой, приборостроительной, бумажной и другим видам промышленности.

Иногда работает много машин, но одна из них является узким местом, и для этой одной машины необходимо составить расписание. Это же относится и к транспорту, обслуживающему клиентов, аэропортам, морским портам, если в их системе есть узкие места (один аэропорт, обслуживающий общий поток – тогда для этого аэропорта необходимо составить расписание). Задача директора также относится к типу составления расписания для одной машины.

Так как $r_i \equiv 0$, то очевидно $\bar{t}^{\,к} = \bar{T}$ и $t^{\max} = T^{\max}$.

В теории расписания доказывается, что при одной машине оптимальное расписание не имеет прерываний (рис. 3.15).

Как видно из рис. 3.15, значения критериев $T^{\max}$ (максимальное время прохождения работ) или равное ему значение $t^{\max}$ (максимальный момент окончания работ) не зависят от расписания (порядка прохождения работ). Но, однако, средние показатели зависят.

Пусть имеются две работы с продолжительностью $P_1 = 2$ и $P_2 = 4$. Для одной машины может быть два типа расписания, которые представлены на рис. 3.16.

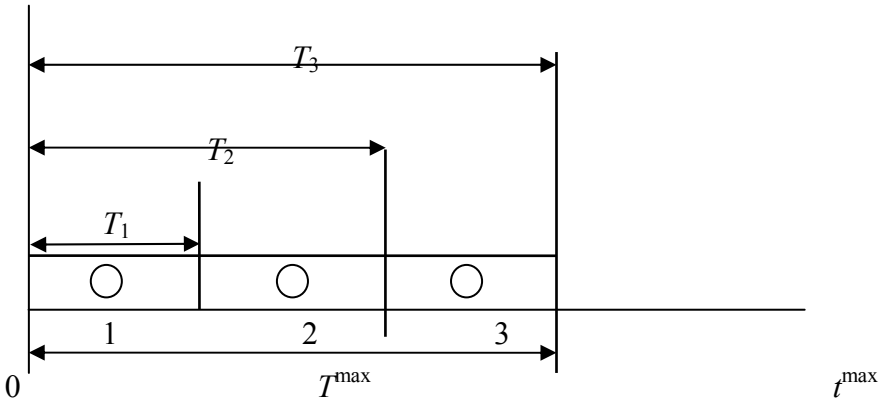
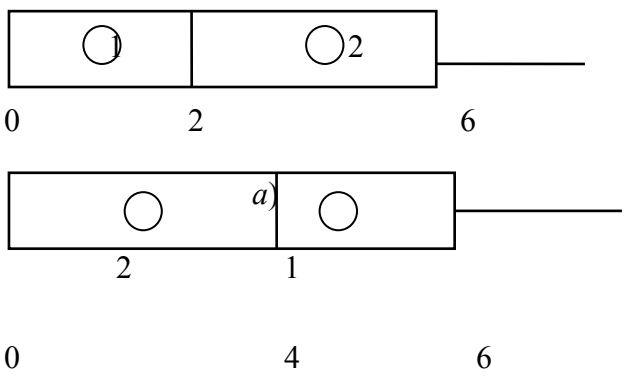


Рис. 3.15 Расписание для одной машины



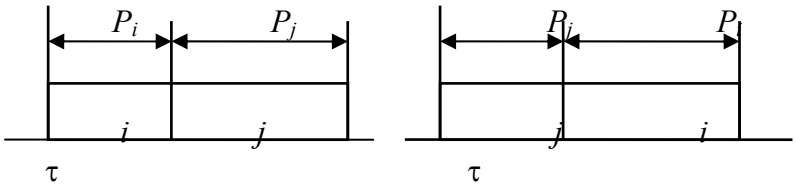
б)

Рис. 3.16 Варианты расписаний:
а – первый вариант; *б* – второй вариант

Для первого варианта (рис. 3.16, *а*) имеем $T_1 = 2, W_1 = 0,$ $W_1 = 2$ и следовательно, $\overline{T} = (2+6)/2 = 4,\ \overline{W} = (0+2)/2 = 1.$

Для второго варианта (рис. 3.16, *б*) – $T_2 = 4,\ T_1 = 6,\ W_2 = 0,\ W_1 = 4;\ \overline{T} = (4+6)/2 = 5,\ \overline{W} = (0+4)/2 = 2$. Оба средних показателя $\overline{T},\ \overline{W}$ лучше там, где вначале проходила короткая работа.

Для общего случая последовательности работ, отличающихся лишь тем, что работы *i*, *j* меняются местами, варианты расписаний представлены на рис. 3.17.



а) *б*)

Рис. 3.17 Варианты расписаний для общего случая:
а – первый вариант; *б* – второй вариант

Средние значения времени ожидания в расписании первого (I) и второго (II) вариантов будут

$$\overline{W}_{\mathrm{I}} = \frac{1}{n}(W^{\mathrm{I}}[1] + W^{\mathrm{I}}[2] + \ldots + W_i^{\mathrm{I}} + W_j^{\mathrm{I}} + \ldots + W^{\mathrm{I}}[n]);$$
$$\overline{W}_{\mathrm{II}} = \frac{1}{n}(W^{\mathrm{II}}[1] + W^{\mathrm{II}}[2] + \ldots + W_i^{\mathrm{II}} + W_j^{\mathrm{II}} + \ldots + W^{\mathrm{II}}[n]),$$

где $\overline{W}_{[\alpha]}$ – время ожидания работ, стоящих на α-м месте.

Очевидно, что

$$W_i^{\mathrm{I}} = \tau,\ W_j^{\mathrm{I}} = \tau + P_i;$$
$$W_j^{\mathrm{II}} = \tau,\ W_i^{\mathrm{II}} = \tau + P_i.$$

Отсюда

$$\overline{W}_{\mathrm{I}} - \overline{W}_{\mathrm{II}} = \frac{1}{n}(\tau + \tau + P_i - \tau - \tau - P_j) = \frac{1}{n}(P_i - P_j).$$

Пусть $P_i > P_j$, тогда $\overline{W}_{\mathrm{I}} - \overline{W}_{\mathrm{II}} > 0$, т.е. $\overline{W}_{\mathrm{I}} > \overline{W}_{\mathrm{II}}.$

Таким образом, работа с меньшей длительностью должна выполняться раньше, чем работа с большей длительностью.

Расписание должно быть составлено так, чтобы работы возрастали по длительности $P_{[1]} \le P_{[2]} \le \ldots \le P_{[n]}.$

Это правило носит название правила кратчайших операций.

В оптимальном решении первой должна выполняться операция с наименьшей длительностью.

Алгоритм, который ранжирует работы и назначает порядок в соответствии с возрастанием ее продолжительности, называется алгоритмом SPT (shortest-processing-timesennencing). Он основан на теореме: среднее время ожидания $\overline{W}$ при одной машине минимально, если $P_{[1]} \le P_{[2]} \le \ldots \le P_{[n]},$ и максимально, если после упорядочения длительности работ последняя не возрастает $P_{[1]} \ge P_{[2]} \ge \ldots \ge P_{[n]}.$

Оптимальное по одному среднему критерию расписание будет оптимально и по другому среднему критерию $\overline{W},\ \overline{t^{\kappa}},\ \overline{T},\ \overline{L},\ \overline{z}.$

SPT минимизирует и целый ряд других целевых функций: минимальную длительность прохождения, минимальное время окончания работ, минимальное ненулевое ожидание, максимальное ожидание.

Из соотношения (3.21) следует

$$\overline{N} = n\frac{\overline{T}}{t^{\max}},$$

т.е. SPT минимизирует и среднее число работ $\overline{N}$ в системе.

3.4.2 РАБОТЫ РАЗНОЙ ВАЖНОСТИ

Для работ разной важности иногда в качестве целевой функции берется $\overline{W}_\alpha$, которая определяется как

$$\overline{W}_\alpha = \frac{1}{n}\sum_{i=1}^{n}\alpha_i W_i \,. \tag{3.22}$$

Расписание будет минимизировать функцию (3.22), если работы выполняются в порядке

$$\frac{P_{[1]}}{\alpha_{[1]}} \le \frac{P_{[2]}}{\alpha_{[2]}} \le \frac{P_{[3]}}{\alpha_{[3]}} \le \ldots \le \frac{P_{[n]}}{\alpha_{[n]}} \,. \tag{3.23}$$

С точки зрения других целевых функций, расписание будет другим. Так, расписание, минимизирующее $L^{\max}$ или $z^{\max}$, таково, что должно выполняться условие

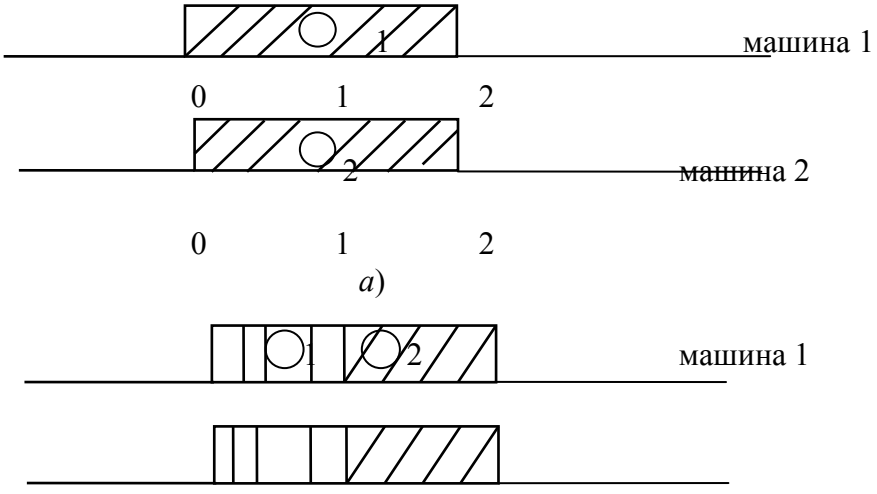
$$d_{[1]} \le d_{[2]} \le d_{[3]} \ldots \le d_{[n]}. \tag{3.24}$$

3.4.3 ПАРАЛЛЕЛЬНЫЕ МАШИНЫ

Когда говорят о параллельных машинах, то под этим понимают случай, при котором имеется не одна, а несколько одинаковых машин, и каждая операция может выполняться на любой машине.

3.3 Исходные данные для двух параллельных машин и двух работ

Работы \ Машины	1	2
1	2	2
2	2	2



а)

б)

Рис. 3.18 Варианты расписаний для двух работ на двух машинах:
а – первый вариант; *б* – второй вариант

Пусть имеются две машины и две работы. Требуется составить расписание, используя исходные данные, представленные в табл. 3.3.

Каждая работа выполняется любой из двух одинаковых машин за время $T = 2$. Здесь могут быть два варианта расписаний (рис. 3.18).

В первом варианте (рис. 3.18, *а*) каждая работа полностью выполняется на своей машине. Во втором варианте (рис. 3.18, *б*) на обеих машинах сначала выполняется первая работа, затем на обеих машинах выполняется вторая работа.

В рассматриваемых вариантах очевидно, что максимальные показатели $T^{\max}$ и $t^{\max}$ одинаковы, а средние нет. Действительно, среднее время прохождения работ для первого варианта $\overline{T} = (2+2)/2 = 2$, а для второго варианта $\overline{T} = (1+2)/2 = 1{,}5.$

Следовательно, второй вариант лучше. Работы лучше производить одновременно на параллельных машинах. Здесь встает вопрос о том,

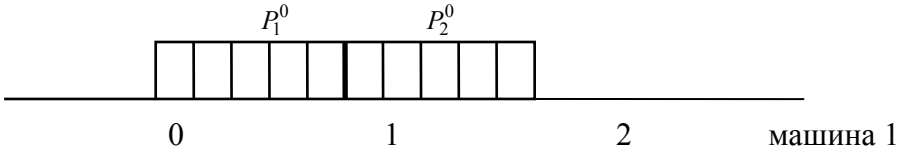


Рис. 3.19 Эквивалентная машина

как же назначать порядок работ. Из рис. 3.18, *б* следует, что можно считать, что у нас есть одна машина с производительностью в два раза большей (рис. 3.19), которая называется эквивалентной.

На эквивалентной машине за единицу времени выполняют первую работу и за вторую единицу времени – вторую работу.

Эквивалентные времена выполнения работ этой машиной будут

$$P_1^0 = \frac{P_1}{2}; \quad P_2^0 = \frac{P_2}{2}.$$

Если имеются m одинаковых машин, то эквивалентная продолжительность работ определяется по формуле

$$P_i^0 = \frac{P_i}{m}.$$

Если машины разные, то

$$P_i^0 = \frac{1}{\sum_{j=1}^{m} \frac{1}{P_{ij}}}.$$

Как и для одной машины, при минимизации средних показателей: среднего времени ожидания $\overline{W}$, среднего времени прохождения работ $\overline{T}$ и других – используется алгоритм SPT:

$$P_{[1]}^0 \le P_{[2]}^0 \le \ldots \le P_{[n]}^0.$$

3.4.4 СИСТЕМЫ КОНВЕЙЕРНОГО ТИПА

Система называется системой конвейерного типа, если машины можно ранжировать таким образом, что первая операция каждой работы выполняется на первой машине, вторая операция на второй и т.д. Именно так обстоит дело при работе на конвейерах, и в связи с этим такие задачи получили название конвейерных.

Пусть производительность работ задана табл. 3.4

3.4 Производительность машин конвейерного типа

Работа 1	111	122	133	144
Работа 2	211	222	233	244
Работа 3	311	322	333	344
Работа 4	411	422	433	444

Как и раньше, первый индекс i обозначает номер работы, второй индекс j обозначает номер операции. Третий индекс здесь уже лишний, так как он совпадает со вторым (ведь j-я операция любой работы делается в конвейерной системе на j-й машине).

Конвейерные системы являются значительным упрощением задачи теории расписания общего вида, хотя и имеет широкое применение. Для этих задач получен ряд частных результатов.

3.4.5 СИСТЕМА ИЗ ДВУХ МАШИН КОНВЕЙЕРНОГО ТИПА

Каждая работа такой системы имеют две операции (имеются две машины). Первая операция производится на первой машине, вторая операция – на второй машине.

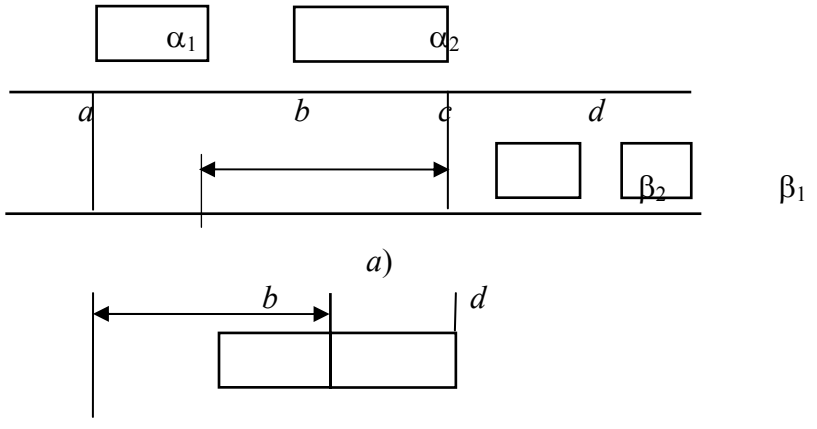
Для обозначения длительности здесь достаточно двух индексов. Так, P_{ij} обозначает длительность j-й операции для i-й работы, причем $j = 1, 2$. Или можно считать P_{ij} длительностью i-й работы, выполняемой на машине j ($j = 1, 2$).

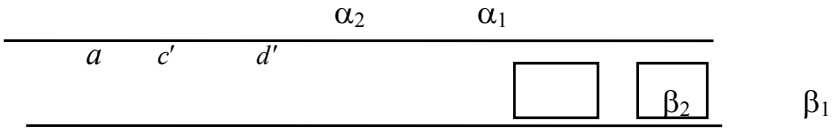
В теории расписания доказывается, что в случае двух машин в системе конвейерного типа порядок выполнения операций на первой машине в оптимальном случае совпадает с порядком выполнения операций на второй машине.

Действительно, пусть имеются две работы $I_1 = (\alpha_1, \beta_1)$ и $I_2 = (\alpha_2, \beta_2)$, для которых указанное правило не выполняется (рис. 3.20, *а*), а расписание оптимально: $\alpha_1 < \alpha_2$, $\beta_2 < \beta_1$.

Очевидно, α_1 можно подвинуть, отрезок "*bd*" сдвинуть так, чтобы точка b' совпала с точкой *a*. Тогда, так как отрезок "*ad*" не трогали, то после работы α_2 можно поместить операцию α_1.

В результате получили α_2 α_1 и β_2 β_1. Новое расписание (рис. 3.20, *б*) не хуже старого, а может быть и лучше, если β_2 можно сдвинуть влево и β_1 также сдвинуть влево.





б)

Рис. 3.20 Расписание для двух машин:
а – правило конвейерной системы не выполняется;
б – оптимальное расписание

Таким образом, можно считать, что при работе на двух машинах порядки выполнения работ на одной машине и на второй совпадают.

Для целевой функции максимальной длительности прохождения работ Джонсоном доказана теорема.

В конвейерной системе из двух машин при одновременном доступе всех работ минимизация максимальной длительности прохождения этих работ $\min(T^{\max})$ осуществляется следующим образом:

работа $I_1 = (\alpha_1, \beta_1)$ предшествует работе $I_2 = (\alpha_2, \beta_2)$, т.е. $I_1 < I_2$, если

$$\min(\alpha_1, \beta_2) \le \min(\alpha_2, \beta_1). \qquad (3.25)$$

Пример 3.1

Пусть $I_1 = (5, 2)$, $I_2 = (10, 4)$, тогда $\min(5, 4) = 4$, $\min(10, 2) = 2$, откуда $4 > 2$ и следовательно $I_2 < I_1$.

На основе теоремы Джонсона построен алгоритм ранжирования работ.

Доказано, что условие (3.25) тождественно следующему.

Работа $I_1 < I_2$, если

а) $\alpha_1 \le \beta_1$, $\alpha_2 \le \beta_2$, $\alpha_1 < \alpha_2$;

б) $\alpha_1 \le \beta_1$, $\alpha_2 > \beta_2$; (3.26)

в) $\alpha_1 \ge \beta_1$, $\alpha_2 \ge \beta_2$, $\beta_1 > \beta_2$.

Условия (3.26) делят все множество работ на два типа (табл. 3.5).

3.5 Формирование расписания для двух машин

Работа I типа	Работа II типа
$W_{\text{I}} = \{I_i\}$	$W_{\text{II}} = \{I_j\}$
$\alpha_i \le \beta_i$	$\alpha_j > \beta_j$
Порядок следования	
$I_{[1]}, I_{[2]}, \ldots, I_{[k]}$	$I_{[k+1]}, I_{[k+2]}, \ldots, I_{[n]}$
$\alpha_{[1]} \le \alpha_{[2]} \le \ldots \le \alpha_{[k]}$	$\beta_{[k+1]} \ge \beta_{[k+2]} \ge \ldots \ge \beta_{[n]}$

Пример 3.2

3.6 Исходные данные

N	1	2	3	4	5	6	7	8
α_i	7	9	1	3	4	6	10	3
β_i	5	11	8	1	9	3	2	6

3.7 Ранжирование работ

Номер работы	1	2	3	4	5	6	7	8
Номер ранжировки	[5]	[4]	[1]	[8]	[3]	[6]	[7]	[2]

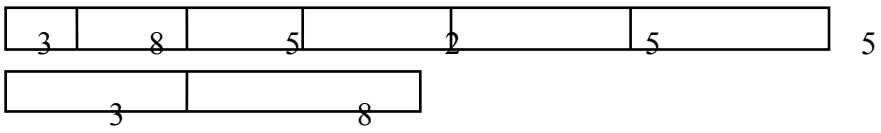


Рис. 3.21 Расписание для двух машин конвейерного типа

3.4.6 СИСТЕМА ИЗ ДВУХ МАШИН КОНВЕЙЕРНОГО ТИПА

Каждая работа в конвейерной системе из трех машин состоит из трех операций: $I_i = (\alpha_i, \beta_i, \gamma_i)$, где α_i – операция, выполняемая на первой машине; β_i – операция, выполняемая на второй машине; γ_i – операция, выполняемая на третьей машине.

Целевой функцией является максимальное время прохождения работ $T^{\max}$, которое необходимо минимизировать.

Здесь возможны следующие варианты:

1 Работы располагаются в порядке $i = 1, 2, 3, \ldots, n$, если

$$\alpha_i \le \beta_i \le \gamma_i;$$
$$\alpha_1 \le \alpha_2 \le \alpha_3 \le \ldots \le \alpha_n;$$
$$\beta_1 \le \beta_2 \le \beta_3 \le \ldots \le \beta_n;$$
$$\gamma_1 \le \gamma_2 \le \gamma_3 \le \ldots \le \gamma_n.$$

2 Операции на второй машине малы, т.е.

$$\min_i \alpha_i \ge \max_i \beta_i$$

или

$$\min_i \gamma_i \ge \max_i \beta_i.$$

В этом случае следует рассматривать комплексы работ $\alpha_i + \beta_i$ и $\gamma_i + \beta_i$ и уже к ним применять теорему Джонсона.

Работа $I_1 < I_2$, если $\min(\alpha_1 + \beta_1, \beta_2 + \gamma_2) < \min(\alpha_2 + \beta_2, \beta_1 + \gamma_1)$. Задача решается так же, как и для двух машин с той лишь разницей, что α_i заменяется на $\alpha_i + \beta_i$, а β_i на $\beta_i + \gamma_i$.

Для непосредственного решения задачи необходимо составить таблицы исходных данных для каждой работы

$$\bar{\alpha}_i = \alpha_i + \beta_i, \ \bar{\beta}_i = \beta_i + \gamma_i$$

и применить Джонсовский алгоритм для ранжирования работ.

3.4.7 СИСТЕМА КОНВЕЙЕРНОГО ТИПА ИЗ *m* МАШИН

В системе конвейерного типа из m машин работа состоит из m операций:

$$I = (\alpha_1, \alpha_2, \ldots, \alpha_m),$$

где α_i выполняется на i-й машине.

Для таких систем мало конечных результатов составления оптимального расписания.

Для целевой функции, представляющей собой средние регулярные критерии $\overline{T}$, $\overline{W}$ и другие, доказана теорема [1]:

В оптимальном расписании должен быть одинаковый порядок выполнения работ на первых двух машинах, если все работы доступны одновременно.

Для целевой функции, минимизирующей максимальную продолжительность $T^{\max}$, доказана следующая **теорема**:

В оптимальном расписании должен быть одинаковый порядок работ на двух первых машинах и на двух последних, т.е. на машинах 1, 2 и на $m-1$, m, если все работы доступны одновременно.

Решение данной задачи осуществляется методом ветвей и границ.

3.5 Алгоритм решения общей задачи составления расписания

Алгоритмы решения общей задачи составления расписания подразделяются на диспетчерские (приближенные) и точные.

3.5.1 АЛГОРИТМЫ ДИСПЕТЧЕРИЗАЦИИ

В алгоритмах диспетчеризации включение в расписание операции производится один раз и навсегда. Операции назначаются одна за другой и также выполняются.

Таким образом, процесс назначения операции и выполнение ее можно совместить, именно поэтому такие алгоритмы называются алгоритмами диспетчеризации.

Пусть $\{S_{so}\}$ – множество ожидающих операций. Если операция взята из $\{S_{so}\}$ и включена в расписание, назад в $\{S_{so}\}$ она вернуться не может, то это алгоритм диспетчеризации.

Наиболее часто используются следующие типы алгоритмов диспетчеризации.

1 Незадерживающая диспетчеризация. Множество $\{S_{so}\}$ разбивается на m подмножеств $\{S_{so}\}^k$ операций, которые ожидают освобождения k-й машины и для которых предшествующие операции выполнены.

Пусть $\{\mathrm{S_{so}}\}^{\mathrm{k}} = \{(\mathrm{i, j, k}), (\mathrm{i_1, j_1, k}), (\mathrm{i_2, j_2, k}), \ldots\}$; $\mathrm{T_k}$ – *момент освобождения* k-*й машины.*

Для машины с номером k выбирается та операция, для которой предыдущая операция выполнена, если такой нет, то выбирается та, у которой минимально время ожидания до начала операции на k-й машине

$$(ij) = \arg\min_{i,j} W_{ijk},$$

где $W_{ijk} = S_{ijk} - T_k$, S_{ijk} – момент возможности начала операции j на машине k, т.е. момент, когда все предыдущие операции этой машины выполнены. Выбор операции для машины с номером k представлен на рис. 3.22.

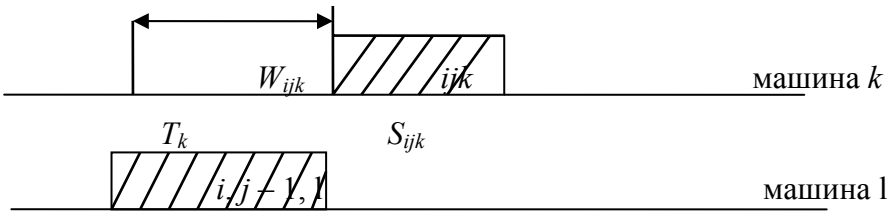


Рис. 3.22 Выбор операции для машины с номером *k*

2 Диспетчеризация по минимуму времени выполнения операций.

При такой диспетчеризации из $\{S_{so}\}^k$ выбирается не та операция, у которой меньше время ожидания от момента освобождения T_k, а та у которой меньше сумма времени ожидания и времени выполнения (рис. 3.23)

$$(ij) = \arg\min_{ij}[\, W_{ijk} + P_{ijk}].$$

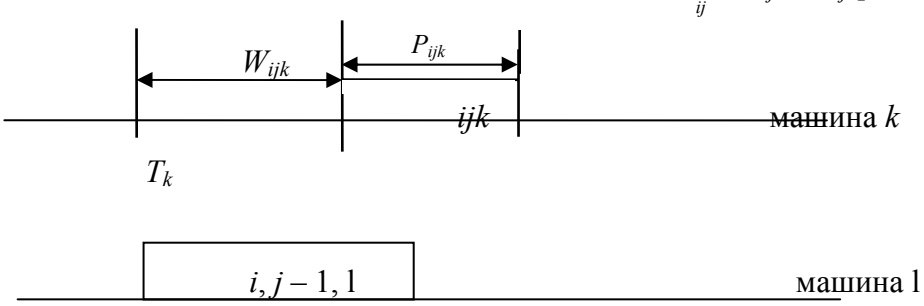


Рис. 3.23 Диспетчеризация по минимуму времени выполнения операций

Диспетчеризация по минимуму времени ожидания и незадерживающимся операциям называется моделированием процесса, так как сам процесс составления расписания моделирует реальное выполнение расписания.

3 Моделирование по приоритетам.

Приоритет является числовой характеристикой, показывающей важность работы.

Сначала выбираются те машины, которые имеют высший приоритет, затем с более низким.

3.5.2 ТОЧНЫЕ МЕТОДЫ ОПРЕДЕЛЕНИЯ РАСПИСАНИЯ

К точным методам составления расписания относится метод ветвей и границ. Для применения метода из теории графов известно, что для этого необходимо:

1) правильно сформулировать, что такое состояние (вершина);
2) формализовать функцию оценки вершины.

Таким образом, состояние – это фрагмент расписания, которое уже построено. Ветвление – операции, идущие за этим фрагментом. Функция оценки

$$Q_n = q_n + \psi_n,$$

где q_n – оценка уже пройденного пути (т.е. оценка фрагмента расписания); ψ_n – функция прогноза, обещание эффективности оставшегося фрагмента расписания, т.е. длительности, ожидания и т.д. до целевой вершины.

Из теории графов известно, что метод будет тем эффективнее, чем ближе функция к (к точному значению $\hat{\psi}_n^*$), тем меньше количество вершин потребуется раскрыть. Причем, если для всех вершин $\hat{\psi}_n$ меньше $\hat{\psi}_n^*$, метод ветвей и границ найдет оптимальное расписание, а если для некоторой вершины будет больше $\hat{\psi}_n^*$ – метод ветвей и границ не гарантирует нахождение оптимального решения.

Для каждой конкретной задачи формализация функции $\hat{\psi}_n$ происходит нестандартно. Часто расчет для каждого узла $\hat{\psi}_n$ сам по себе сложная задача, требующая длительных расчетов. При этом встает задача компромисса между точностью оценки нижней границы $\hat{\psi}_n^*$ и временем, необходимым для расчета этой оценки. Может оказаться, что целесообразно загрубить $\hat{\psi}_n$, что приводит к большему числу вершин, которые нужно раскрыть, однако, время на расчет функции оценки этих вершин затрачивается мало.

Наиболее распространены следующие методы расчета оценок.

1 Используя уже составленный до данной вершины фрагмент расписания, формализуется все расписание одним из методов диспетчеризации. При этом значением ψ_n для этой вершины будет значение максимального идеального времени прохождения работ полученного расписания.

2 Для каждой оставшейся невыполненной работы определяется время его полного выполнения. К этому времени прибавляется время возможного начала выполнения этой работы. За оценку нижней границы для этой вершины берется максимум этой величины по всем работам.

3 Для каждой из машин находится сумма длительностей всех оставшихся операций, выполняемых на данной машине. К этой сумме добавляется время начала выполнения операций на этой машине. За оценку нижней границы для вершины берется максимум этих величин по всем машинам.

Первый из описанных алгоритмов наиболее трудоемок, алгоритмы 2 и 3 требуют значительно меньшего времени для вычисления, однако, они дают часто слишком низкую оценку функции ψ, что приводит к большему числу раскрываемых вершин.

Алгоритм 2 эффективнее, если осталось много работ с высокой длительностью, большей, чем длительности уже запланированных работ.

Алгоритм 3 эффективен, если существует машина, у которой время еще невыполненных работ больше, чем выполненных.

4 ТЕОРИЯ МАССОВОГО ОБСЛУЖИВАНИЯ

Практические требования рациональной организации массового обслуживания: билетные кассы, магазины, автоматы и прочее, а также телефонного дела – физики выдвинули в начале нашего столетия в ряд интересных математических задач нового типа. Задачи подобного типа возникают в самых разнообразных направлениях исследований: в естествознании, в технике, в экономике, транспорте, военном деле, организации производства. Решением этих задач занимается теория массового обслуживания. Итак, теория массового обслуживания занимается изучением вопросов организации и обслуживания потока требований или заявок.

4.1 Некоторые понятия теории массового обслуживания

Требованием или заявкой называется объект, который необходимо обслужить. Такими объектами могут быть станок, подлежащий ремонту, самолет противника, который надо сбить, железнодорожные составы, проходящие через железнодорожный узел, покупатели, приобретающие товар, и т.д. Как видно, объект является носителем запроса. Поэтому в дальнейшем под требованием и заявкой понимается не только объект, но и сам запрос на обслуживание. Например, запрос на ремонт станка, запрос на уничтожение самолета, запрос на продажу товара покупателю и т.д.

Совокупность появляющихся требований называется потоком требований.

Устройства, удовлетворяющие запросу на обслуживание, называются обслуживающими устройствами, аппаратами или приборами. Эти термины используются широко, т.е. прибором могут быть как устройство, собственно прибор, но также и совокупность устройств, человек, коллектив завода и т.д., словом все те люди и механизмы, с помощью которых удовлетворяется запрос на обслуживание.

Совокупность всех обслуживающих устройств называется цехом. Термин "цех" также понимается в широком смысле. Так, магазин с

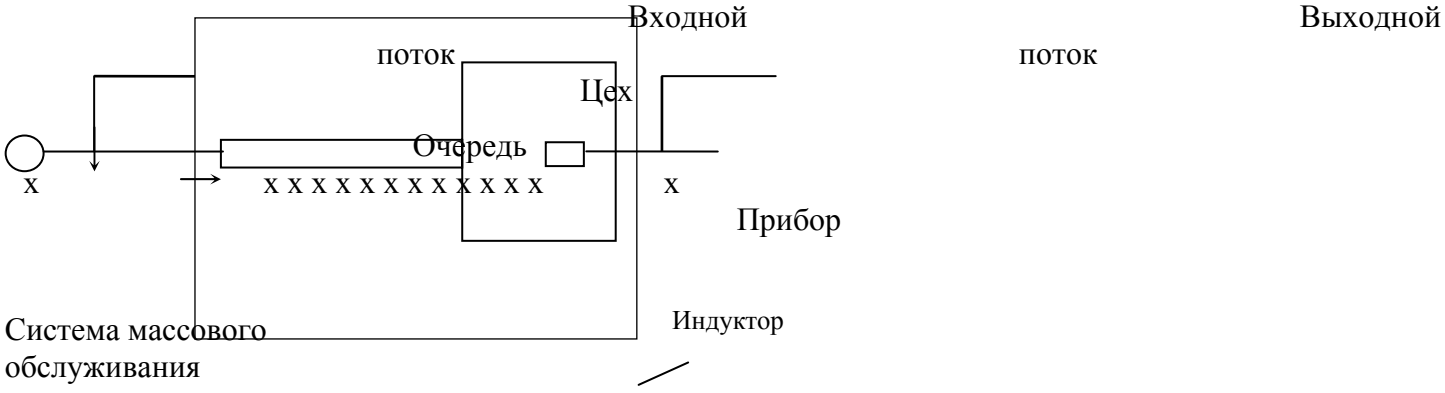


Рис. 4.1 Система массового обслуживания

покупателями – это цех с приборами. Таким образом, цех может содержать один или несколько приборов в зависимости от того, сколько обслуживающих устройств обслуживает поток требований.

Если время обслуживания велико, появляется очередь, т.е. множество требований, желающих быть обслуженными, но еще не обслуженных.

Цех вместе с очередью называется системой массового обслуживания. Эта система изображена на рис. 4.1. Непосредственно сама система выделена пунктирным контуром, который является внешним контуром системы.

Работу системы массового обслуживания можно абстрактно представить следующим образом: генератор (источник) генерирует очередное требование, которое поступает в систему и либо становится в очередь на обслуживание, либо, если очереди нет, поступает в цех, где прибор начинает выполнять запрос на обслуживание.

Последовательность требований, входящих в систему массового обслуживания, называется входящим (входным) потоком, выходящие требования называются выходящим (выходным) потоком.

4.2 Цели и задачи теории массового обслуживания

Целью теории массового обслуживания является создание моделей различных систем массового обслуживания для анализа операционных показателей этих систем и синтеза целесообразных систем массового обслуживания.

Операционными показателями систем массового обслуживания являются:

– вероятность наличия очереди;

– средняя длина очереди;

– среднее время ожидания начала обслуживания;

– степень загруженности обслуживающей системы;

– определение числа необслуженных требований.

Теория массового обслуживания ставит своей целью получение математического описания, позволяющего рассчитать операционные показатели в зависимости от варьируемых параметров таких, как число приборов, эффективность приборов, организация процесса обслуживания и другие, а также параметров входного потока.

Математическую модель массового обслуживания в операторной форме можно представить как

$$Q = f(x, u), \tag{4.1}$$

где Q – вектор операционных показателей; x – параметры входного потока требований; u – варьируемые параметры; f – оператор, устанавливающий связь между Q и x, u.

Вектор варьируемых параметров обычно разбивается на две составляющие $u = (u_1, u_2)$, где u_1 – вектор дисциплины очереди; u_2 – вектор механизма обслуживания.

Дисциплиной очереди называется порядок выбора требований из очереди. Обычно используют следующие дисциплины очереди:

1) "первый пришел – первый обслуживается" – дисциплина "живой очереди";

2) "последний пришел – первый обслуживается" – примером такой системы является склад, заполненный изделиями, из которого на доработку удобно брать изделия, поступившие последними;

3) выбор требований случайным способом;

4) выбор требований в соответствии с присвоенными приоритетами.

К составляющим механизма обслуживания относятся эффективность, т.е. скорость, с которой прибор обслуживает требования; количество каналов обслуживания или, другими словами, число параллельных приборов, обслуживающих требования; наличие последовательных приборов.

Входной поток x характеризуется различной интенсивностью (скоростью возникновения новых заявок), структурой (числом очередей), характером поведения требований (требования могут быть ″терпеливые″ и ″нетерпеливые″), ограниченностью и неограниченностью максимального числа требований и другими показателями.

Если для некоторого типа систем массового обслуживания математическая модель (4.1) создана, то задача теории массового обслуживания считается решенной. Задача оптимизации операционных показателей или зависимости от них экономических показателей решается обычно поисковыми методами или иными методами принятия оптимальных решений и уже, по сути дела, не относится к задачам собственно теории массового обслуживания.

Задачи, решаемые теорией массового обслуживания, чрезвычайно разнообразны, поэтому их следует каким-либо образом классифицировать.

4.3 Классификация задач теории массового обслуживания

4.3.1 КЛАССИФИКАЦИЯ ПО ОБЩИМ ПРИЗНАКАМ

Одна из возможных классификаций систем массового обслуживания представлена на рис. 4.2. Согласно этой классификации система массового обслуживания разделяется на три подсистемы.

Первая подсистема – это система массового обслуживания без потерь. Под термином "система без потерь" (с полным ожиданием) будем понимать систему, в которой, если все приборы заняты, требование становится в очередь и не покидает ее до тех пор, пока не будет обслужено.

Вторая подсистема – это система с частичными потерями. Эта подсистема характеризуется тем, что требование либо не становится в очередь, если эта очередь превышает по длине некоторую величину (система с ограниченной длиной очереди), либо становится в очередь, но покидает ее, если время пребывания в ней превышает определенную величину (система с ограниченным временем пребывания), или, если время ожидания в очереди начала обслуживания превышает определенную величину (система с ограниченным временем ожидания начала обслуживания).

Третья подсистема – это система без очередей. Под этим термином понимают систему, в которой требование покидает систему, если все обслуживающие устройства (приборы) заняты. В такой системе, очевидно, очереди не может быть.

Системы, имеющие очередь, подразделяются на системы с одной очередью и системы с несколькими очередями.

Все системы массового обслуживания делятся на системы с одним каналом и системы с конечным числом каналов. Под термином "канал" понимают обслуживающее устройство в цехе, пропускающее через себя требование (удовлетворяющее запрос на обслуживание). Таким образом, число каналов обслуживания равно числу приборов. В тех случаях, когда приборов много, удобно (математически более просто) считать, что их бесконечное число.

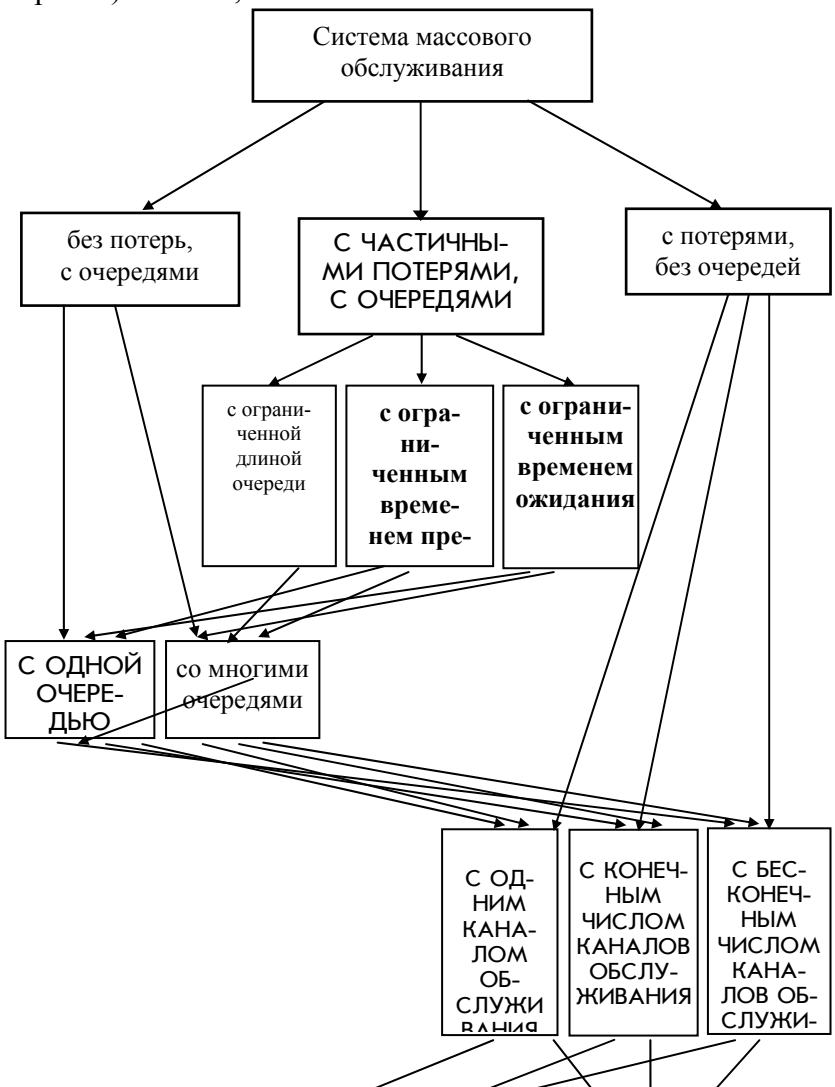


Рис. 4.2 Классификация систем массового обслуживания

Все системы можно разделить на системы с бесконечным числом требований (например, запросы на телефонные переговоры, на обслуживание покупателей, автомашины на бензозаправках и т.д.) и с конечным числом требований в системе (группа ремонта станков в цехе: число станков известно, тренировка футболистов футбольной команды, лечение больных студентов в институтской поликлинике и т.п.). Представленная классификация, конечно, не исчерпывает все множество различных систем массового обслуживания. Эти системы могут классифицироваться и по другим признакам.

Так, весьма важной характеристикой является дисциплина обслуживания, в соответствии с которой системы подразделяются на четыре вида, как уже указано в п. 4.2.

Другими вариантами классификации являются следующие.

Поступление требований может быть единичным и групповым.

Требования могут обслуживаться параллельно работающими приборами, но может быть и система, в которой приборы расположены последовательно, так, что как только будет обслужено требование первым прибором, то начнет обслуживаться и другое и т.д.

Интенсивное обслуживание прибором может быть постоянным или зависеть от длины очереди, приоритетов или каких-либо других факторов.

Наконец системы массового обслуживания различают по характеру входного потока и по характеру обслуживающих устройств.

4.3.2 КЛАССИФИКАЦИЯ ВХОДНЫХ ПОТОКОВ

По характеру входной поток требований разделяется на детерминированный поток требований и стохастический (рис. 4.3).

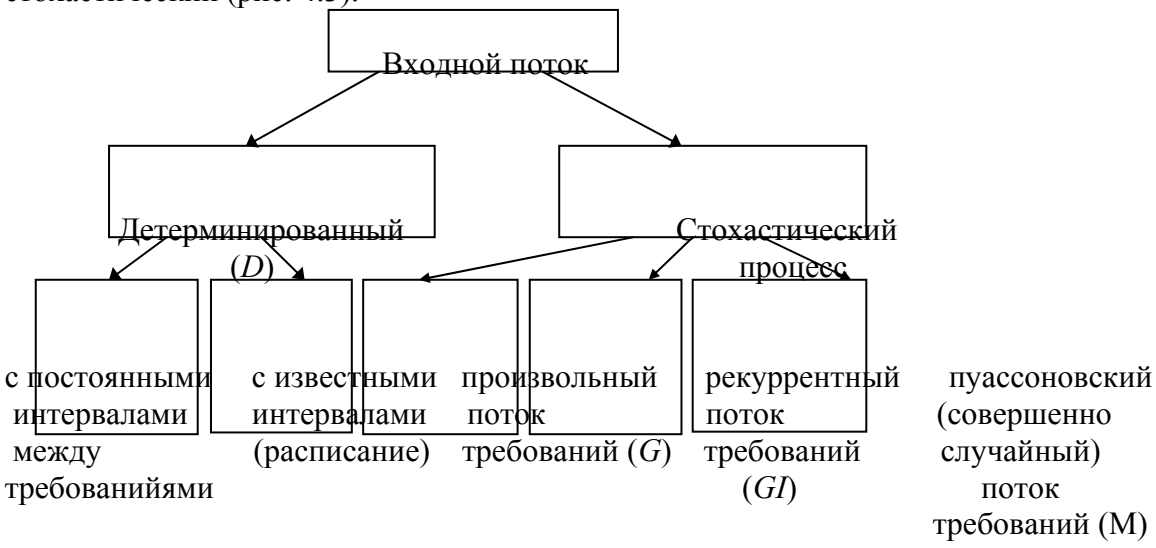


Рис. 4.3 Классификация входного потока

Детерминированный входной поток может быть двух видов. В первом случае требования поступают через равные промежутки времени $\Delta t = 1/\lambda$, где λ – интенсивность потока, т.е. число требований в единицу времени.

Другим видом детерминированного потока является поток, в котором требования хотя и поступают, но не через равные промежутки времени, а по известной программе – расписанию, когда моменты поступления новых требований известны заранее. В этом случае можно определить средний интервал $\overline{\Delta t}$ между поступлениями требований, который определяется как $\overline{\Delta t} = 1/\lambda$, где λ – по-прежнему средняя интенсивность.

Если промежутки времени между поступлениями требований случайны, то это будет стохастический процесс. Этот случай имеет наибольшее практическое значение, так как поступление заявок на обслуживание в подавляющем большинстве случаев случайно.

Стохастический поток требований подразделяется на три вида: поток с произвольными стохастическими свойствами, рекуррентный поток и совершенно случайный или пуассоновский поток требований.

Произвольный поток требований характеризуется тем, что в нем не накладывается никаких ограничений на стохастическую независимость интервалов между поступлениями требований, а также на характер вероятностных законов, описывающих интервалы между требованиями.

Входной поток называется рекуррентным, если он характеризуется следующими свойствами:

а) продолжительность интервалов между поступлениями требований стохастически независимы;

б) продолжительность интервалов описывается одной и той же плотностью распределения.

Рекуррентный поток требований называется процессом восстановления. Средняя продолжительность $\overline{\Delta t}$ интервалов между последовательными поступлениями требований определяется по формуле математического ожидания

$$\overline{\Delta t} = \int_0^\infty \Delta t g(\Delta t) d\Delta t = 1/\lambda. \qquad (4.2)$$

Входной поток называется совершенно случайным или простейшим, если для него характерны следующие признаки:

а) продолжительность интервалов между поступлениями требований статически независимы;

б) продолжительность интервалов описывается одной и той же плотностью распределения;

в) вероятность поступления требований на достаточно малом интервале Δt не зависит от времени t, а зависит только лишь от величины Δt (это свойство называется стационарностью или однородностью прихода);

г) вероятность поступления требований на интервале Δt не зависит от предыстории процесса, т.е. от того, сколько требований поступило, каким образом (через какие промежутки времени) и в какой момент по отношению к интервалу Δt. Это свойство называется отсутствием памяти или свойством отсутствия последствия. Говорят, что процесс поступления требования в этом случае имеет марковский характер;

д) характер потока требований таков, что в любой момент времени может поступить только одно требование. Это свойство называется свойством ординарности. В таких потоках вероятность появления двух требований в малом интервале Δt пренебрежима мала, она имеет порядок малости выше Δt.

Таким образом, простейший поток требований или совершенно случайный поток – это поток, определяющийся свойствами стационарности, ординарности и отсутствием последствия одновременно.

Предположения о совершенно случайном входном потоке требований эквивалентно тому, что плотность распределения интервалов времени между последовательными поступлениями требований описывается экспоненциальным законом

$$g(t) = \lambda e^{-\lambda t}, \quad t \geq 0. \qquad (4.3)$$

Таким образом, можно сказать, что входной поток является совершенно случайным или простейшим, если

а) плотность вероятности интервалов между последовательными поступлениями требований распределена по одному и тому же экспоненциальному закону (4.3);

б) поток ординарен, т.е. в малом интервале Δt поступление двух и более требований невозможно или почти невозможно;

Докажем, что поток требований, интервалы между которыми распределены с плотностью вероятности (4.3), удовлетворяет свойствам стационарности и отсутствия последствия.

Для этого обозначим вероятность отсутствия требований на интервале [0, t]

$$P^0(t) = \text{Вер}\{\tau > t\} = \int_0^\infty \lambda e^{-\lambda\tau} d\tau = e^{-\lambda t}, \qquad (4.4)$$

а на интервале [t, $t + \Delta t$]

$$P^0(t) = \text{Вер}\{t \le \tau \le t + \Delta t\} = \int_t^{t+\Delta t} \lambda e^{-\lambda\tau} d\tau = e^{-\lambda\Delta t} = 1 - \lambda\Delta t + 0(\Delta t^2). \qquad (4.5)$$

При малом Δt эта вероятность (4.5) может быть представлена в виде

$$P^0(\Delta t / t) = P^0(\Delta t) \approx 1 - \lambda\Delta t. \qquad (4.6)$$

Из формулы (4.6) видно, что вероятность $P^0(\Delta t / t)$ зависит лишь от величины интервала времени Δt, поэтому ее будем обозначать $P^0(\Delta t)$.

Вероятность того, что в интервале Δt появится хотя бы одно требование, также не зависит от момента времени t и предыстории процесса и определяется как

$$P(\Delta t) = 1 - P^0(\Delta t) = \lambda\Delta t, \qquad (4.7)$$

что и доказывает стационарность процесса и отсутствие последствия.

Так как вероятность поступления в малый интервал времени двух и более требований ничтожно мала, то (4.7) показывает вероятность $P_1(\Delta t)$ поступления одного требования в интервале Δt.

Таким образом,

$$P_1(\Delta t) = \lambda\Delta t, \qquad (4.8)$$

которая с ростом Δt пропорционально растет, что свидетельствует о появлении на данном интервале нового требования.

Согласно (4.3) средний интервал времени между появлением требований будет $\bar{t} = 1/\lambda$, дисперсия $D_t = 1/\lambda^2$.

Если $P_n(t)$ – вероятность поступления n требований в системе на интервале [0, t] и если интервалы распределены по экспоненциальному закону (4.3), то вероятность $P_n(t)$, n = 0, 1, 2, …, удовлетворяет закону Пуассона

$$P_n(t) = \frac{(\lambda t)^n e^{-\lambda t}}{n!}. \qquad (4.9)$$

Процесс в этом случае называется пуассоновским.

Итак, если интервалы распределены по экспоненциальному закону, то процесс пуассоновский, т.е. $P_n(t)$ удовлетворяет (4.9), и наоборот, если процесс пуассоновский, то интервалы между последовательными поступлениями требований распределены по экспоненциальному закону. Такие процессы называются M-процессами (марковскими).

P 4 P

$\lambda = 5$

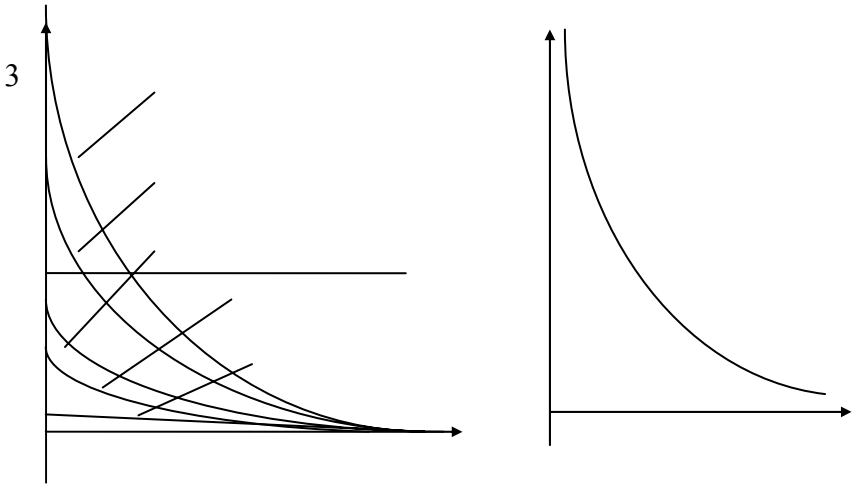


Рис. 4.4 Экспоненциальное распределение

Рис. 4.5 Пуассоновское распределение

Пример экспоненциального распределения приведен на рис. 4.4, а на рис. 4.5 – пуассоновское распределение $P_n(t)$. Пуассоновским законом распределения описываются очень многие реальные потоки требований на обслуживание.

Кроме закона Пуассона? часто применяется закон распределения Эрланга, который используется для аппроксимации широкого круга распределений, отличающихся от экспоненциального, путем подбора параметров k и λ.

$$g(t) = \frac{(\lambda k)(\lambda k t)^{k-1} e^{-\lambda k t}}{(k-1)!}, \ t \geq 0. \qquad (4.10)$$

Если $\lambda = 1$, эрланговский закон превращается в экспоненциальный.

На рис. 4.6 представлена взаимосвязь различных видов входных потоков. Как следует из рисунка и определений, данных выше, пуассоновский поток является частным случаем эрланговского потока, который, в свою очередь, является частным случаем рекуррентного потока. Последний же является частным случаем общего потока – стохастического произвольного.

В соответствии с принятой терминологией Кендалла обозначают: M – экспоненциальное распределение; D – детерминированное распределение; E_k – k-фазное распределение Эрланга; GI – рекуррентный входной поток; G – общий вид распределения.

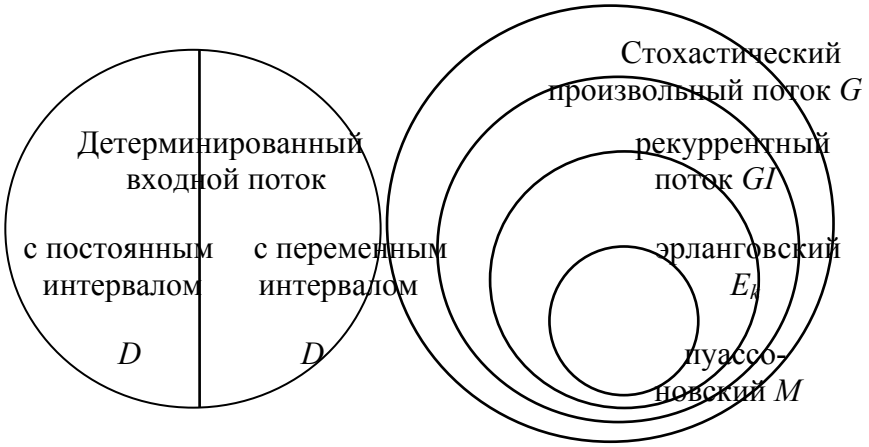


Рис. 4.6 Отношение различных видов входных потоков

4.3.3 КЛАССИФИКАЦИЯ ПРОЦЕССОВ ОБСЛУЖИВАНИЯ

Аналогично входному потоку процесс обслуживания требований может быть детерминированным и стохастическим.

Детерминированный процесс обслуживания характеризуется постоянной величиной времени обслуживания

$$t_0 = 1/\mu,$$

где μ – интенсивность обслуживания, которая представляет собой число требований, обслуживаемых в единицу времени.

Стохастический процесс обслуживания может быть произвольным, рекуррентным или совершенно случайным, как и при описании входного потока требований.

На практике считают и это чаще всего соответствует реальным ситуациям, что время обслуживания подчиняется экспоненциальному закону

$$\omega(t) = \mu e^{-\mu t}, \quad t \geq 0. \tag{4.11}$$

Здесь параметр μ представляет собой среднее время обслуживания

$$\overline{t_0} = \int_0^\infty t\omega(t)dt = 1/\mu.$$

Таким образом, параметр μ – это среднее число требований, обслуживаемых в единицу времени. Дисперсия в этом случае определяется как

$$D_{t0} = 1/\mu^2.$$

Как уже указывалось, экспоненциальный закон распределения времени предполагает, что случайный процесс является стационарным, без последствия. При допущении ординарности процесса, когда в достаточно малом интервале времени не могут окончиться обслуживания двух и более требований, процесс, описываемый (4.11) является совершенно случайным. При этом, как и при классификации входных потоков, поток обслуживания требований является пуассоновским, т.е. вероятность, что за время *t* будет окончено обслуживание *n* требований, определяется по формуле

$$P_n(t) = \frac{(\mu t)^n e^{-\mu t}}{n!}. \tag{4.12}$$

Вероятность того, что при интервале времени меньшем *t* ни одно обслуживание не будет окончено, определяется как

$$G^0(t) = \int_t^\infty \omega(t)dt = e^{-\mu t}. \tag{4.13}$$

Вероятность, что при интервале времени *t* будет закончено хотя бы одно обслуживание, составит

$$G(t) = 1 - e^{-\mu t}. \tag{4.14}$$

При малых интервалах времени Δt, для которых не может закончиться обслуживание более одного требования, как следует из (4.14), вероятность того, что закончится обслуживание одного требования в интервале Δt, будет

$$G^1(\Delta t) = 1 - e^{-\mu\Delta t} \approx \mu\Delta t + 0(\Delta t^2). \qquad (4.15)$$

Аналогично входному потоку можно показать, что это выражение справедливо и для вероятности окончания обслуживания полного требования в интервале $[t, t + \Delta t]$. При экспоненциальном распределении времени обслуживания этот процесс стационарен и без последствия, т.е. совершенно случаен.

4.3.4 ОБОЗНАЧЕНИЯ КЕНДАЛЛА СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ

Для систем массового обслуживания Кендаллом введены следующие обозначения: $H_1|H_2|i$, где H_1 – характеристика входного потока; H_2 – характеристика времени обслуживания прибора; i – число приборов (каналов).

Так, например, $M|M|1$ – обозначает систему с пуассоновским входным потоком, экспоненциальным временем обслуживания, одним прибором; $GI|D|2$ – система с рекуррентным входным потоком, детерминированным временем обслуживания, с двумя потоками (приборами); $E_2|G|4$ – система с временем обслуживания, имеющим общий вид распределения по двухфазному распределению Эрланга, с четырьмя обслуживающими устройствами.

В дальнейшем системы массового обслуживания будут обозначаться следующим образом

$$\frac{K}{R} = \frac{|H_1|H_2|i|}{|\pi|\lambda|\mu|s|r|p|},$$

где K – обозначения Кендалла; R – обозначения подсистем систем массового обслуживания типа, характеризуемого K.

Под R понимается следующая последовательность

$$R\Delta = |\pi|\lambda|\mu|s|r|p|,$$

где π – может быть либо числом N, либо числом ∞: число N обозначает, что в системе число требований оценимо и их не может быть более N; знак ∞ обозначает, что поток требований не ограничен (бесконечен);

λ – характеристика интенсивности входного потока: если на этом месте стоит λ, то это обозначает, что интенсивность постоянна (λ = const), если $\lambda(x)$, то интенсивность зависит от параметра системы x;

μ – характеристика интенсивности обслуживания: знак μ обозначает, что μ = const; знак μ_n обозначает, что интенсивность пропорциональна длине очереди; $\mu(x)$ обозначает, что интенсивность зависит от параметра системы x;

s – характеризует "терпеливость" требований: на этом месте может стоять H, что обозначает отсутствие очереди или абсолютные потери (полностью нетерпеливые клиенты); Т обозначает отсутствие потерь – требования не уходят из очереди (безусловно терпеливые клиенты); уТ – условно терпеливые клиенты, при этом условия могут быть: $m \le M$ – очередь не может быть больше M, $t_0 \le \tau$ – время ожидания в очереди не может быть больше τ и т.д.; r – число очередей;

p – обозначает наличие или отсутствие приоритетов: 0 – отсутствие, 1 – наличие.

Пусть система массового обслуживания обозначена

$$\frac{|M|M|2|}{|\infty|\lambda|\mu|T|1|0|},$$

СОГЛАСНО ПРИНЯТЫМ ОБОЗНАЧЕНИЯМ ЭТО СИСТЕМА С ПУАССОНОВСКИМ ВХОДНЫМ ПОТОКОМ, ЭКСПОНЕНЦИАЛЬНЫМ ВРЕМЕНЕМ ОБСЛУЖИВАНИЯ, С ДВУМЯ ОБСЛУЖИВАЮЩИМИ ПРИБОРАМИ, С НЕОГРАНИЧЕННЫМ ПОТОКОМ ТРЕБОВАНИЙ, С ОДНОЙ ОЧЕРЕДЬЮ, БЕЗ ПРИОРИТЕТОВ.

Система $\frac{|G|M|3|}{|\infty|\lambda|\mu|yT|1|0|}$ отличается от предыдущей тем, что входной поток произволен, имеет три обслуживающих прибора, требования терпеливы при выполнении некоторого условия. Так $m \le 20$ означает, что, если в очереди стоит уже 20 клиентов, то остальные требования теряются.

Ниже рассматриваются конкретные типы систем массового обслуживания.

4.4 Системы массового обслуживания с очередью без потерь

4.4.1 ОБЩАЯ ХАРАКТЕРИСТИКА СИСТЕМЫ

Система массового обслуживания с очередью без потерь обозначается

$$\frac{|M|M|1|}{|\infty|\lambda|\mu|T|1|0|}.$$

Простейшая система данного типа характеризуется следующими свойствами:

а) в системе имеется один канал обслуживания (в цехе один прибор);

б) поток требований не ограничен;

в) входной поток совершенно случайный с параметром λ;

г) длительность обслуживания распределена по показательному закону с параметром μ;

д) параметры λ и μ постоянны.

Условная схема системы массового обслуживания без потерь с очередью и одним каналом обслуживания показана на рис. 4.7.

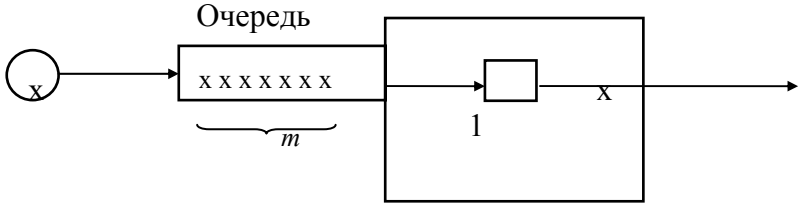


Рис. 4.7 Система массового обслуживания без потерь с очередью и одним каналом обслуживания

Примером такой системы может быть очередь на приеме у должностного лица, директора, юриста, очередь в билетную кассу, а также задачи ремонта техники, снабжения и т.п.

Пусть m – число требований в очереди, n – число требований, находящихся в системе (т.е. в очереди и в цехе).

Очевидно, что

$$n = 0, 1, 2, \ldots;$$
$$m = \begin{cases} 0, & \text{если } n = 0; \\ n - 1, & \text{если } n = 1, 2, \ldots \end{cases} \tag{4.16}$$

Так как входной поток пуассоновский, то

$$P^0(\Delta t) = 1 - \lambda \Delta t + 0(\Delta t^2);$$
$$P^1(\Delta t) = \lambda \Delta t + 0(\Delta t^2). \tag{4.17}$$

Обслуживание подчиняется экспоненциальному закону

$$G^0(\Delta t) = 1 - \mu \Delta t + 0(\Delta t^2);$$
$$G^1(\Delta t) = \mu \Delta t + 0(\Delta t^2). \tag{4.18}$$

Число n требований в системе называется состоянием системы.

Вероятность P_n того, что в системе находится точно n требований, называется вероятностью состояния n:

$$P_n = \text{Вер}\ \{\nu = n\}.$$

Если эта вероятность не меняется во времени, то такой процесс обслуживания называется статическим стационарным. Очевидно, при изменении параметров входного потока или интенсивности приборов эта величина может меняться во времени, тогда процесс называется статическим нестационарным.

Обозначим через $P_n(t)$ вероятность того, что в системе находится точно n требований в момент времени t, если в начальный момент требований не было, и $P_{ni}(t)$ вероятность того, что в системе находится точно n требований в момент времени t, но если в начальный момент в системе было i требований.

4.4.2 ПРОЦЕСС РОЖДЕНИЯ И ГИБЕЛИ

Класс случайных процессов марковского типа начали изучать и в связи с биологическими постановками вопросов о численности популяций, распространении эпидемий и т.п. Это обстоятельство привело к тому, что подобные процессы получили название "процессы рождения и гибели" и нашли широкое применение во многих прикладных вопросах, далеких по своему физическому характеру от биологических. Рассматриваемый класс процессов относится к данному классу и описывается уравнением изменения $P_{ni}(t)$ во времени.

Если s – событие, заключающееся в том, что в момент $t + \Delta t$ состояние системы будет n, а в момент времени $t = 0$ состояние системы было i, то вероятность события s будет $P_{ni}(t + \Delta t)$.

Интервал времени $[0, t + \Delta t]$ разделим на два интервала $[0, t]$ и $[t, t + \Delta t]$. Как указывалось выше, за время Δt, если оно мало, в систему не могут поступить два и более требований и не может закончиться обслуживание двух и более требований. Отсюда следует, что состояние системы за время Δt может изменяться только за счет того, в систему войдет или выйдет одно требование.

Рассмотрим некоторое событие $s = s_A + s_B + s_C + s_D$. На рис. 4.8 представлена полная система событий s_A, s_B, s_C, s_D таких, что событие s произойдет, если произойдет одно из этих событий. На рис. 4.8, *а* и 4.8, *б* показана ситуация, когда в момент времени t состояние системы n, которое останется в момент времени $t + \Delta t$ только в том случае, если в систему не поступит и не выйдет ни одного требования, или, если поступит и выйдет по одному требованию. Событие s_C (рис. 4.8, *в*) заключается в том, что в момент времени t состояние системы было $n - 1$, но за время Δt одно требование поступило, но ни одно не вышло. Событие s_D (рис. 4.8, *г*) состоит в том, что в момент времени t состояние системы было $n + 1$, за время Δt одно требование вышло из системы и ни одного не вошло.

Вероятности P_{sA}, P_{sB}, P_{sC}, P_{sD} соответственно событий s_A, s_B, s_C, s_D выражаются соотношениями

$$P_{s_A} = P_{in}(t)P^0(\Delta t)G^0(\Delta t) \approx P_{in}(t)(1-\lambda_n\Delta t)(1-\mu_n\Delta t);$$
$$P_{s_B} = P_{in}(t)P^1(\Delta t)G^1(\Delta t) \approx P_{in}(t)\lambda_n\Delta t\mu_n\Delta t;$$
$$P_{s_C} = P_{in-1}(t)P^1(\Delta t)G^0(\Delta t) \approx P_{in}(t)\lambda_{n-1}\Delta t(1-\mu_{n-1}\Delta t);$$
$$P_{s_D} = P_{in+1}(t)P^0(\Delta t)G^1(\Delta t) \approx P_{in+1}(t)(1-\lambda_{n+1}\Delta t)(\mu_{n+1}\Delta t).$$

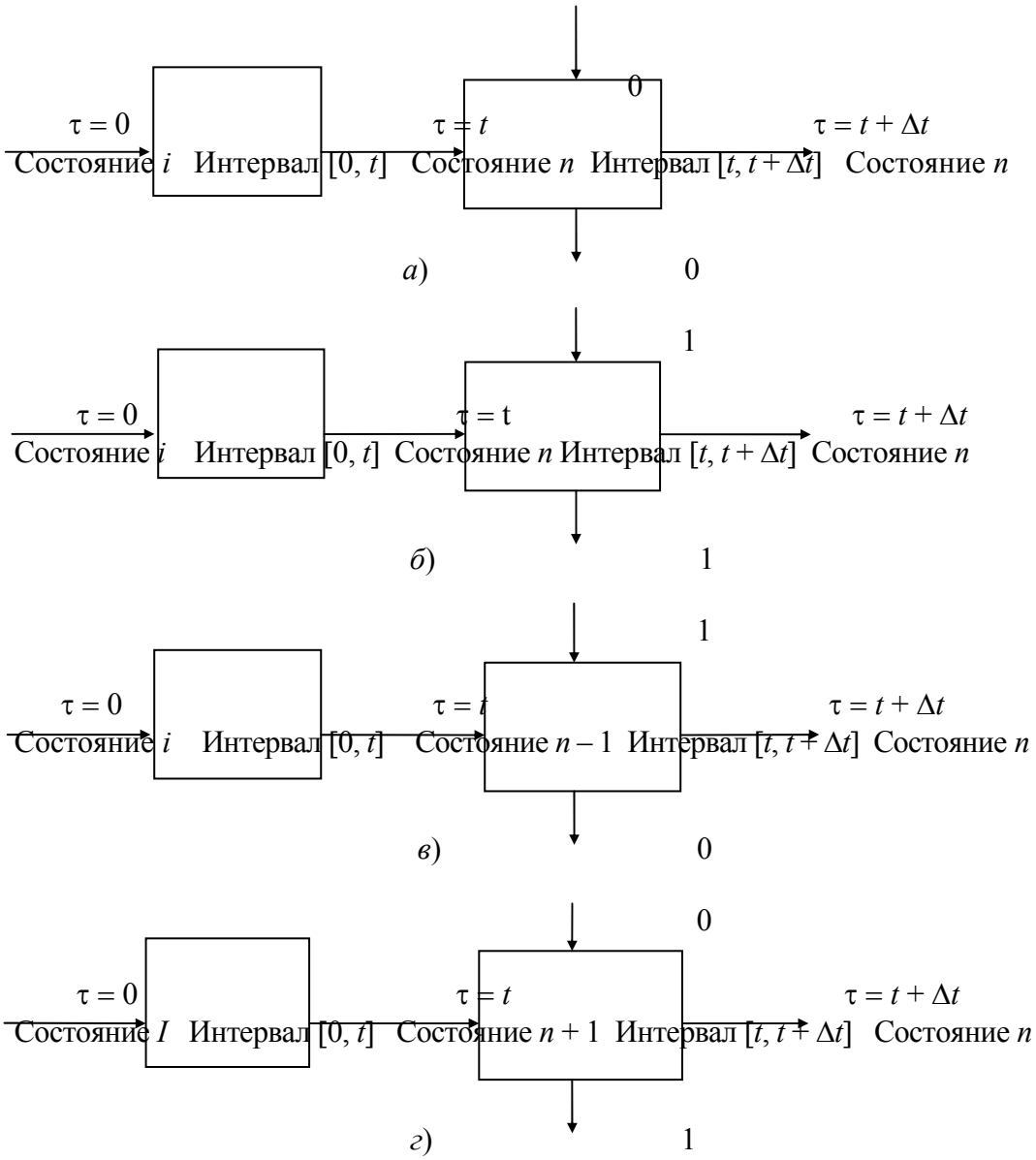


Рис. 4.8 Событие $s = s_A + s_B + s_C + s_D$, составляющие события s:
а – s_A; *б* – s_B; *в* – s_C; *г* – s_D

В этих уравнениях индексы при λ и μ показывают, что в общем случае параметры λ и μ зависят от состояния системы.

Так как $s = s_A + s_B + s_C + s_D$, то $P_{in}(t + \Delta t) = P_{sA} + P_{sB} + P_{sC} + P_{sD}$
и следовательно

$$P_{in}(t+\Delta t) = P_{in}(t)(1-\lambda_n\Delta t)(1-\mu_n\Delta t) + P_{in}(t)\lambda_n\Delta t\mu_n\Delta t +$$
$$+ P_{in}(t)\lambda_{n-1}\Delta t(1-\mu_{n-1}\Delta t) + P_{in}(t)(1-\lambda_{n+1}\Delta t)\mu_{n+1}\Delta t.$$

Если $\Delta t \to 0$, то получим дифференциальное уравнение, описывающее процессы рождения и гибели:

$$\frac{dP_{in}}{dt} = \lambda_{n-1}P_{i_{n-1}}(t) - (\lambda_n + \mu_n)P_{in}(t) + P_{i_{n+1}}(t)\mu_{n+1}, \qquad (4.19)$$

при начальных условиях

$$P_{in}(0) = 0,\ n = 0,\ 1,\ 2,\ \ldots,\ i-1,\ i+1,\ \ldots,$$

$$P_{ii}(0) = 1.$$

Уравнение (4.19) справедливо для $n = 0, 1, 2, \ldots$ Если для $n = 0$ считать, что $\lambda_{n-1} = \lambda_{-1} \equiv 0$ и $\mu_0 \equiv 0$, то это уравнение преобразуется к виду

$$\frac{dP_{0i}}{dt} = -\lambda_0 P_{i0}(t) + P_{i1}\mu_1. \qquad (4.19а)$$

Уравнение (4.19) называется моделью процессов рождения и гибели.

Если принять, что в момент времени $t = 0$ состояние системы равно нулю и $\mu_n \equiv 0$ для $n = 0, 1, 2, \ldots$, то из модели (4.19) получается модель чистого рождения

$$\frac{dP_{0n}}{dt} = \lambda_{n-1} P_{0n-1}(t) - \lambda_n P_{0n}(t),\ \ n = 0,\ 1,\ 2,\ \ldots$$

При условии $\lambda_{-1} \equiv 0$ и начальных условиях $P_{0n}(0) = 0$, $n = 1, 2, 3$; $P_{00}(0) = 1$. Индекс "0" в этих уравнениях, очевидно, можно опустить, и тогда модель чистого рождения принимает вид

$$\begin{cases} \dfrac{dP_0}{dt} = -\lambda_0 P_0(t); \\ \dfrac{dP_n}{dt} = \lambda_{n-1} P_{n-1}(t) - \lambda_n P_n(t),\ \ n = 1,\ 2,\ \ldots; \end{cases} \qquad (4.20)$$

$$P(0) = 1,\ P_n(0) = 0,\ n = 1,\ 2,\ \ldots$$

Решение системы дифференциальных уравнений (4.20) в явном виде дает изменение вероятности состояния $P_n(t)$ во времени

$$P_n(t) = \frac{(\lambda t)^{n-1} e^{-\lambda t}}{n!},$$

но это и есть пуассоновский закон (4.9), которым, по предположению, описывается входной поток требований (рождение требований).

Если принять, что в момент времени $t = 0$, состояние системы будет равно s и $\lambda_n \equiv 0$ для $n = 0, 1, 2, 3, \ldots$, то из модели (4.19) получается модель чистой гибели;

$$\frac{dP_{sn}}{dt} = -\mu_n P_{sn}(t) + \mu_{n+1} P_{sn+1}(t),\ \ n = 0,\ 1,\ 2,\ \ldots$$

при условии $\mu_0 \equiv 0,\ \mu_{s,s+1} \equiv 0$

и начальных условиях $P_{sn}(0) = 0$, $n = 0, 1, 2, \ldots, s-1, s+1, \ldots$, $P_{ss}(0) = 1$. Опуская индекс s в этих уравнениях, получают модель чистой гибели при $\mu_{sn} \equiv 0$ в виде

$$\begin{cases} \dfrac{dP_0}{dt} = \mu_1 P_1(t); \\ \dfrac{dP_n}{dt} = -\mu_n P_n(t) + \mu_{n+1} P_{n+1}(t),\ n = 1,\ 2; \end{cases} \qquad (4.21)$$

$$P_n(0) = 0,\ n = 0,\ 1,\ 2,\ \ldots,\ s-1,\ s+1,\ \ldots,\ P_s(0) = 1.$$

Рассмотрим частный случай общей модели рождения и смерти (4.19), когда при $t = 0$ состояние системы $n = 0$, параметры λ и μ – const. Этот случай соответствует очереди с одним клиентом, пуассоновским входным потоком и экспоненциальным временем обслуживания. В уравнениях модели (4.19) можно оставить в этом случае лишь один индекс, т.е.

$$\begin{cases} \dfrac{dP_0}{dt} = -\lambda P_0(t) + \mu P_1(t); \\ \dfrac{dP_n}{dt} = \lambda P_{n-1}(t) - (\lambda + \mu) P_n(t) + \mu P_{n+1}(t), \ n = 1, \ 2, \ \ldots \end{cases} \tag{4.22}$$

Система дифференциальных уравнений (4.22) описывает изменение во времени вероятностей состояния $P_n(t)$, $n = 0, 1, 2, \ldots$, которые называются неустановившимися вероятностями состояния.

Начальные условия для (4.22)

$$P_0(0) = 1, \ P_n(0) = 0, \ n = 1, 2, \ldots \tag{4.23}$$

Систему уравнений (4.22) с начальными условиями (4.23) можно решить, его решение будет сходится к статистически стационарному процессу

$$P_n = \lim_{t \to \infty} P_n(t), \ n = 0, 1, \ldots$$

только при условии $\lambda/\mu < 1$.

Интуитивно это понятно, поскольку статистическая стационарность наступает только тогда, когда средний интервал $1/\lambda$ между появлениями новых требований меньше среднего интервала $1/\mu$ обслуживания.

Обозначим отношение параметра входного потока λ (интенсивности) к параметру обслуживания (интенсивности) или, что тоже самое, отношение среднего времени обслуживания $1/\mu$ к среднему интервалу между появлениями новых требований как

$$\rho = 1/\mu : 1/\lambda = 1/\mu.$$

Эту величину называют трафик-интенсивностью.

Если $\rho < 1$, то всегда существуют статистические стационарные состояния. Уравнения для их определения получают из (4.22), приравнивая нулю производные

$$\begin{cases} 0 = -\lambda P_0 + \mu P_1; \\ 0 = \lambda P_{n-1} - (\lambda + \mu) P_n + \mu P_{n+1}, \ n = 1, \ 2, \ \ldots \end{cases} \tag{4.24}$$

Эти уравнения называют уравнениями в конечных разностях.

Можно показать, что при $\rho < 1$ система массового обслуживания входит в стационарный режим, при этом установившиеся или стационарные вероятности P_n, $n = 0, 1, \ldots$, удовлетворяют системе уравнений (4.24).

Решение этой системы дает

$$P_n = \lim_{t \to \infty} P_n(t) = \rho^n (1 - \rho), \ n = 0, 1, \ldots \tag{4.25}$$

Из (4.25) следует, что

$$P_0 = 1 - \rho, \ \text{т.е.} \tag{4.26}$$

$$\rho = 1 - P_0 = \underline{P}. \tag{4.27}$$

Так как P_0 является вероятностью отсутствия очереди, $\underline{P} = 1 - P_0$, а, следовательно, и ρ – вероятностью наличия очереди. Другими словами, трафик-интенсивность ρ можно трактовать, как долю времени, в течение которого прибор обслуживает требование (работает). Поэтому величина $\rho = \lambda/\mu$ называ-

ется также коэффициентом загруженности обслуживающего устройства или коэффициентом использования.

4.4.3 ОЦЕНОЧНЫЕ ХАРАКТЕРИСТИКИ

Имея выражения (4.25), (4.26) для расчета P_n, легко рассчитать другие оценочные характеристики, среди которых выделяют следующие.

1 Вероятность нахождения в системе не более n требований

$$F(n) = P_0 + P_1 + \ldots + P_n = \sum_{i=1}^{n} \rho^i (1-\rho) = (1-\rho)\sum_{i=1}^{n} \rho^i.$$

Сумма в последнем выражении составляет геометрическую прогрессию и определяется по формуле

$$\sum_{i=1}^{n} \rho^i = \frac{1-\rho^{n+1}}{1-\rho},$$

откуда

$$F(n) = \text{Вер}(\nu \le n) = 1 - \rho^{n-1}. \qquad (4.28)$$

2 Вероятность $\theta(n) = \text{Вер}\{\nu > n\}$ является дополнительной к $F(n)$:

$$\theta(n) = \text{Вер}\{\nu > n\} = \rho^{n+1}. \qquad (4.29)$$

Вероятность $\theta(0)$ является вероятностью существования очереди и из (4.29) следует, что для $n = 0$

$$\theta(n) = \text{Вер}\{\nu > n\} = \underline{P} = \rho.$$

3 Среднее число требований в системе $\overline{n}$ определяется как

$$\overline{n} = \sum_{n=0}^{\infty} nP_n = \sum_{n=0}^{\infty} n\rho^n(1-\rho) = (1-\rho)\rho\sum_{n=0}^{\infty} n\rho^{n-1} = (1-\rho)\rho\frac{d}{d\rho}\left(\sum_{n=0}^{\infty}\rho^n\right).$$

Так как сумма геометрической прогрессии равна

$$\sum_{n=1}^{\infty} \rho^n = \frac{1}{1-\rho,}$$

то

$$\overline{n} = \frac{\rho}{1-\rho}. \qquad (4.30)$$

Таким образом, между коэффициентом загруженности (трафик-интенсивностью) ρ и средним числом требований $\overline{n}$ в системе существует однозначная связь (рис. 4.9).

4 Среднее число требований в очереди $\overline{m}$ равно

$$\overline{m} = \sum_{n=0}^{\infty}(n-1)P_n = \sum_{n=0}^{\infty} nP_n - \sum_{n=0}^{\infty} P_n + P_0 = \overline{n} - 1 + P_0$$

или

$$\overline{m} = \frac{\rho^2}{1-\rho}. \qquad (4.31)$$

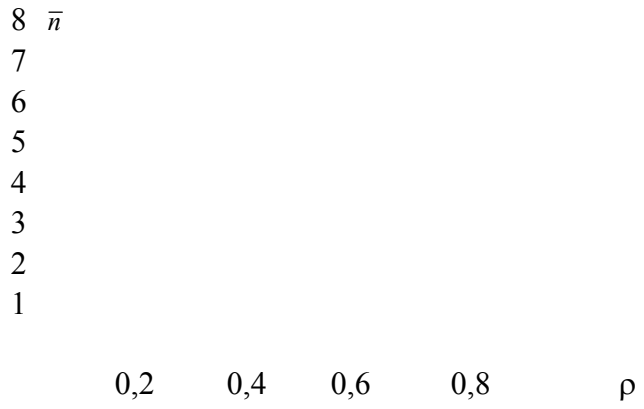


Рис. 4.9 Зависимость среднего числа требований от трафик-интенсивности

Вероятность P_0 иногда называют коэффициентом простоя прибора $k^{\text{пр}}$:

$$k^{\text{пр}} = P_0 = 1 - \rho,$$

отсюда

$$\bar{m} = \bar{n} - 1 + P_0 = \bar{n} - \rho.$$

Сравнивая выражения (4.30) и (4.31), можно записать $\bar{m} = \bar{n}\rho$ или

$$\frac{\bar{m}}{\lambda} = \frac{\bar{n}}{\mu}.$$

5 Время ожидания в очереди t_f связано с временем ожидания в системе t_s соотношением

$$t_s = t_f + t_0,$$

где t_0 – время обслуживания.

Такое же соотношение, очевидно, справедливо и для средних величин

$$\bar{t}_s = \bar{t}_f + \bar{t}_0.$$

Значения средних времен ожидания в системе и в очереди определяются соответственно по формулам

$$\bar{t}_s = \bar{n} / \lambda = \bar{n} / \mu \frac{1}{\rho};$$
$$\bar{t}_f = \bar{m} / \lambda = \bar{n} / \mu.$$

С учетом (4.30) последние соотношения преобразуются к виду

$$\bar{t}_s = \frac{1}{\lambda}\frac{\rho}{1-\rho} = \frac{1}{\mu}\frac{1}{1-\rho}; \qquad (4.32а)$$

$$\bar{t}_f = \frac{1}{\lambda}\frac{\rho^2}{1-\rho} = \frac{1}{\mu}\frac{\rho}{1-\rho}, \qquad (4.32б)$$

откуда

$$\bar{t}_f = \bar{t}_s\rho.$$

Разность между $\bar{t}_s$ и $\bar{t}_f$ равна, с одной стороны, ρ, а с другой

$$\rho = \bar{t}_s - \bar{t}_f = \frac{1}{\mu}\left(\frac{1}{1-\rho} - \frac{\rho}{1-\rho}\right) = 1/\mu.$$

6 Распределение времени ожидания в очереди $F(t_f)$, т.е. распределение вероятности того, что время ожидания в очереди меньше t_f: $\text{Вер}\{\tau \le t_f\}$.

Очевидно, что $F(t_s) = \int_0^{t_f} f(\tau)d\tau$, где $f(\tau)$ – плотность распределения времени ожидания в очереди.

Предположим, что в момент времени τ_0 в систему поступает требование y_0 (рис. 4.10). Спустя некоторое время $\Delta\tau$, требование проходит очередь и попадает в цех на обслуживание.

Время ожидания в очереди зависит от того, сколько требований уже было в очереди в момент времени τ_0.

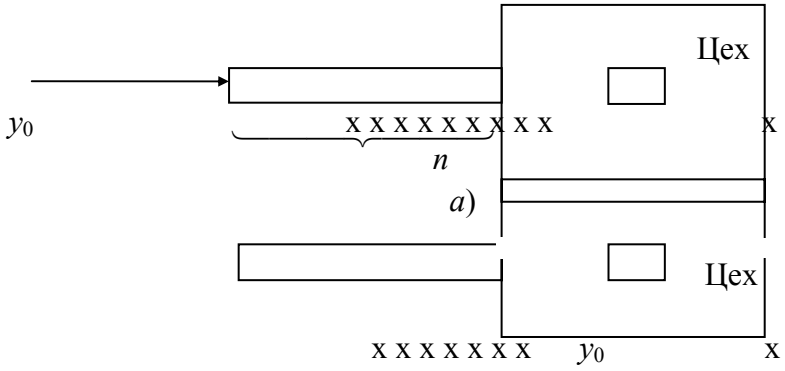


Рис. 4.10 Ожидание в очереди:

а – момент времени τ_0; *б* – момент времени $\tau = \tau_0 + t_f$

Если обозначить $\underline{P}(t_f)$ – вероятность того, что время ожидания лежит в пределах $[t_f, t_f + dt]$; т.е. $t_f \le \Delta\tau \le t_f + dt$; $\underline{P}(n, t_f)$ – вероятность того, что время ожидания лежит в пределах $[t_f, t_f + dt]$ и в момент времени τ_0 в системе было n требований, то, очевидно, по теореме сложения вероятностей можно записать

$$P(t_f) = \sum_{n=0}^{\infty} P(n, t_f).$$

При отсутствии очереди ($n = 0$) требования сразу поступают на обслуживание, поэтому вероятность того, что $\Delta\tau$ лежит в пределах $[t_f, t_f + dt]$, равна нулю, и $P(t_f)$ определяется по формуле

$$P(t_f) = \sum_{n=1}^{\infty} P(n, t_f). \qquad (4.33)$$

Вероятность $P(n, t_f)$ определяется по формуле произведения вероятностей

$$P(n, t_f) = P_n(\tau_0)P(n-1/n)P(n/n-1, n),$$

где $P_n(\tau_0)$ – вероятность того, что в момент τ_0 в системе было n требований; $P(n - 1/n)$ – условная вероятность того, что к моменту времени $\tau_0 + t_f$ (т.е. за время t_f) $(n - 1)$-е требование уже обслужено (т.е. начато обслуживание n-го требования) при условии, что в момент времени τ_0 в системе было n требований; $P(n/n - 1, n)$ – вероятность того, что за время dt n-е требование закончит обслуживаться при усло-

вии, что в момент времени τ_0 в системе было n требований и к моменту времени $\tau_0 + t_f$ обслуживание $(n - 1)$-го требования закончили.

Так как рассматривается установившийся процесс, то

$$P_n(\tau_0) = P_n = \rho^n(1-\rho). \tag{4.34}$$

Вследствие стационарности процесса вероятность обслуживания $(n - 1)$-го требования за время t_f не зависит от числа требований в системе и определяется по формуле (4.12)

$$P(n-1/n) = \frac{(\mu t_f)^{n-1} e^{-\mu t_f}}{(n-1)!}. \tag{4.35}$$

Вероятность того, что за время Δt будет обслужено одно n-е требование, пропорциональна интервалу dt, поэтому согласно (4.15)

$$P(n/n-1,\ n) = G^1(dt) = \mu dt. \tag{4.36}$$

Из (4.33) – (4.36) следует, что

$$P(t_f) = \sum_{n=1}^{\infty} \rho^n (1-\rho) \frac{(\mu t_f)^{n-1} e^{-\mu t_f}}{(n-1)!} \mu dt =$$
$$= (1-\rho)\rho e^{-\mu t_f} \mu dt \sum_{n-1}^{\infty} \frac{(\rho \mu t_f)^{n-1}}{} \tag{4.37}$$

Обозначив $P(t_f) = f(t_f)dt$, где $f(t_f)$ – плотность распределения вероятности того, что время ожидания лежит в пределах $[t_f, t_f + dt]$, и приняв $k = n - 1$, можно записать

$$f(t_s)dt = (1-\rho)\rho e^{-\mu t_f} \mu dt \sum_{k=0}^{\infty} \frac{(\rho \mu t_f)^k}{k!}.$$

С учетом

$$\sum_{k=0}^{\infty} \frac{(\rho \mu t_f)^k}{k!} = e^{\rho \mu t_f}$$

окончательно получают

$$f(t_s)dt = (1-\rho)\rho\mu e^{-\mu t_f (1-\rho)} dt. \tag{4.38}$$

Таким образом, распределение времени ожидания в очереди будет

$$F(t_s) = \int_0^t (1-\rho)\rho\mu e^{-\mu\tau(1-\rho)} d\tau = 1 - \rho e^{-\mu t_f (1-\rho)}.$$

Вероятность $\theta(t_f) = \text{Вер}\{\tau > t_f\}$ определяется по формуле

$$\theta(t_f) = 1 - F(t_f) = \rho e^{-\mu t_f (1-\rho)}.$$

Вероятность того, что время ожидания t_f будет равно нулю, т.е. ждать не придется совсем, будет равна $F(0) = 1 - \rho$. Это же соотношение можно получить и из (4.29) при $n = 0$, как вероятность того, что в системе не находится ни одно требование.

4.4.4 РАСЧЕТ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ С ОЧЕРЕДЬЮ БЕЗ ПОТЕРЬ

4.4.4.1 Система $\frac{|M|M|1|}{|\infty|\lambda|\mu|T|1|0|}$

1 Трафик-интенсивность

$$\rho = \lambda / \mu \quad \text{или} \quad \rho = \overline{n} /(1 - \overline{n}).$$

2 Вероятность нахождения в системе точно n требований

$$P_n = \rho^n (1 - \rho), \ n = 0, \ 1, \ 2, \ \ldots$$

или

$$P_0 = 1 - \rho, \ P_n = \rho P_{n-1}, \ n = 1, \ 2, \ ...$$

3 Вероятность, что в системе находится не более n требований

$$F(n) = 1 - \rho^{n+1}.$$

4 Среднее число требований в системе

$$\overline{n} = \rho /(1 - \rho) \ .$$

5 Среднее число требований в очереди

$$\overline{m} = \rho^2 /(1 - \rho) = \overline{n}\rho = \overline{n} - \rho.$$

6 Вероятность, что время ожидания в очереди меньше t_f

$$F(t_f) = 1 - \rho e^{-\mu t_f (1-\rho)} .$$

7 Среднее время ожидания в очереди

$$\bar{t}_f = \frac{1}{\lambda} \frac{\rho^2}{1 - \rho} = \bar{t}_s \rho = \bar{t}_s - 1/\mu.$$

8 Среднее время ожидания в системе

$$\bar{t}_s = \frac{1}{\mu} \frac{1}{1 - \rho} = \bar{t}_s + 1/\mu.$$

9 Коэффициент простоя (вероятность простоя) обслуживающего устройства (прибора)

$$k^{\text{пр}} = P_0 = 1 - \rho.$$

10 Вероятность существования очереди

$$\underline{P} = 1 - P_0 = \rho.$$

4.4.4.2 Система $\frac{|M|D|1|}{|\infty|\lambda|\mu|T|1|0|}$

Рассматриваемая система отличается от предыдущей тем, что время обслуживания t_0 является постоянной величиной: $t_0 = 1/\mu = \text{const}$. Такими системами могут быть аэропорты, железнодорожные вокзалы, морские и речные порты, отпуск товаров с заводов оптовым покупателям и др.

Основными расчетными соотношениями для этой системы являются следующие:

1 Трафик-интенсивность

$$\rho = \lambda / \mu.$$

2 Среднее время нахождения в очереди

$$\bar{t}_f = \frac{1}{\mu}\frac{\rho}{2(1-\rho)}.$$

3 Средняя длина очереди

$$\bar{n} = \rho + \frac{\rho^2}{2(1-\rho)} = \frac{\rho}{1-\rho} - \frac{\rho^2}{2(1-\rho)}.$$

Таким образом, при равномерном обслуживании среднее время ожидания в очереди сокращается в два раза, средняя длина очереди уменьшается. Можно показать, что среднее время ожидания в очереди и средняя длина очереди в этом случае минимальны.

4.4.4.3 Система $\frac{|M|G|1|}{|\infty|\lambda|\mu|T|1|0|}$

В этой системе время обслуживания распределено не по экспоненциальному, а по произвольному закону с математическим ожиданием $\bar{t} = 1/\mu$ и дисперсией D_t.

Расчетные соотношения следующие:

1 Средняя длина очереди

$$\bar{n} = \rho + \frac{\rho + \lambda^2 D_t}{2(1-\rho)} = \frac{\rho}{1-\rho} - \frac{\rho^2 - \lambda^2 D_t}{2(1-\rho)}.$$

При $D_t = 0$, т.е. когда обслуживание постоянно

$$\bar{n} = \rho + \frac{\rho^2}{2(1-\rho)} = \frac{\rho}{1-\rho} - \frac{\rho^2}{2(1-\rho)}.$$

Таким образом, с увеличением дисперсии (разброса) времен обслуживания очередь увеличивается.

При $D_t = 1/\mu^2$ имеет место экспоненциальное распределение и следовательно

$$\bar{n} = \frac{\rho}{1-\rho} - \frac{\rho^2 - \lambda^2 \frac{1}{\mu^2}}{2(1-\rho)} = \frac{\rho}{1-\rho},$$

ЧТО СОВПАДАЕТ С (4.30).

2 Время ожидания в очереди

$$\bar{t}_f = \frac{1}{\lambda}\frac{\rho^2 + \lambda^2 D_t}{2(1-\rho)}.$$

В случае пуассоновского закона $D_t = 1/\mu^2$ и тогда

$$\bar{t}_f = \frac{1}{\lambda}\frac{\rho^2 + \lambda^2(\frac{1}{\mu^2})}{2(1-\rho)} = \frac{1}{\lambda}\frac{\rho^2}{1-\rho},$$

ЧТО СОВПАДАЕТ С (4.32Б).

3 Среднее время ожидания в системе

$$\bar{t}_s = \bar{t}_f + 1/\mu.$$

4.4.4.4 Система $\frac{|M|M|1|}{|\infty|\lambda|\mu_n|T|1|0|}$

Данная система характеризуется тем, что параметр потока μ не является постоянной величиной, а пропорционален числу требований в системе: $\mu = \mu_0 n$, где $\mu_0 = \text{const}$. Примером подобной системы может быть столовая, где скорость обслуживания обычно пропорциональна числу ожидающих клиентов. Вообще говоря, в системах массового обслуживания процесс обслуживания интенсифицируется при росте
очереди.

Основные расчетные соотношения имеют вид.

1 Вероятность нахождения точно n требований в системе

$$P_n = \frac{(\lambda/\mu_0)^n e^{-\lambda/\mu_0}}{n!}.$$

2 Среднее число требований в системе

$$\bar{n} = \lambda/\mu.$$

3 Дисперсия $D_n = \lambda/\mu$.

4 В переходном процессе среднее число требований изменяется по закону

$$\bar{n}(t) = \frac{\lambda}{\mu}(1 - e^{-\mu t}).$$

Таким образом, среднее число требований увеличивается от нуля до числа $\bar{n}(t) = \lambda/\mu$. При этом, чем меньше значение параметра μ, тем дольше длится переходный процесс.

4.4.4.5 Система $\frac{|M|M|1|}{|N|\lambda|\mu|T|1|0|}$

Система отличается от предшествующей тем, что число требований в системе ограничено. Схема этой системы представлена на рис. 4.11.

Пусть максимальное число требований, которые могут находиться в системе массового обслуживания, равно N. Эта ситуация имеет место, например, при ремонте станков в цехе. Общее число станков равно N. Поэтому число N и есть максимально возможное число станков, которые потенциально могут

сломаться и находиться в системе массового обслуживания. Ремонт станков производится одной бригадой.

Другим примером могут быть студенты, обслуживающиеся в студенческой поликлинике, спортсмены, тренируемые одним тренером и т.д.

Если n – число станков в системе массового обслуживания, то в очереди будет станков

$$m = \begin{cases} 0, & \text{если} \quad n = 0; \\ n-1, & \text{если} \quad n = 1, 2, \ldots, N, \end{cases}$$

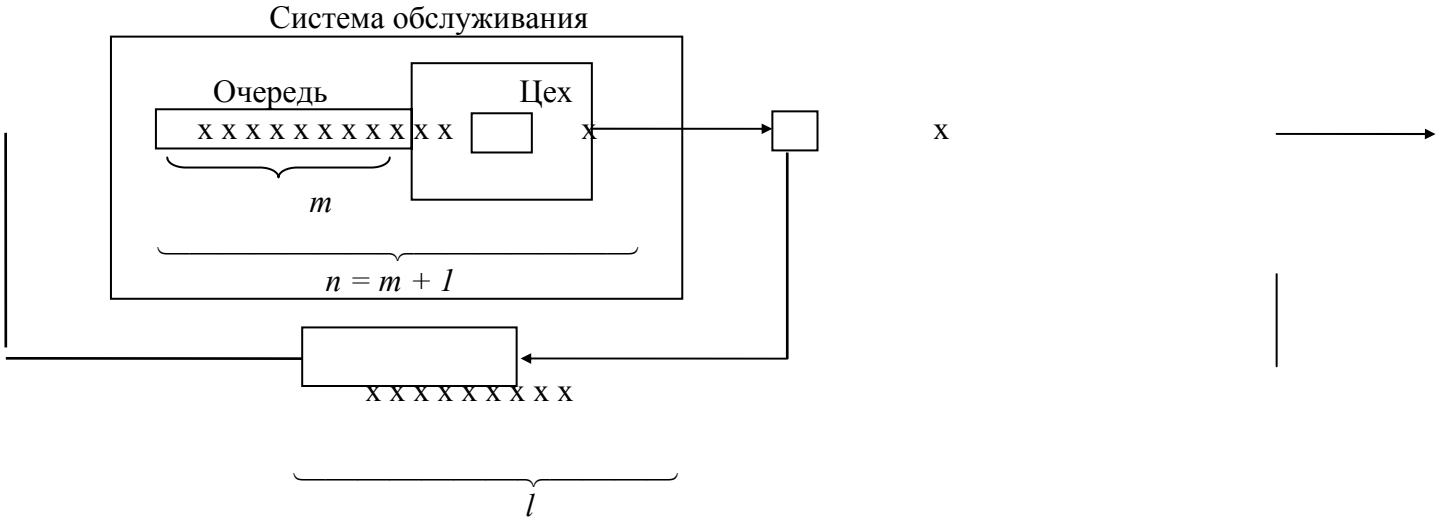


Рис. 4.11 Система с ограниченным числом требований

предельное число которых не может быть равно n и тем более m:

$$n \le m, \qquad m \le N - 1.$$

Очевидно, что реальная интенсивность появления в очереди требований зависит от того, сколько требований уже находится в очереди и сколько осталось "свободных" (не находящихся в системе). Если все требования уже стоят в очереди, то нового требования поступить уже не может, т.е. $\lambda_N = 0$. В общем случае можно считать, что

$$\lambda_n = (N - n)\lambda, \quad 0 \le n \le N,$$

интенсивность обслуживания

$$; \qquad \mu_n = \mu, \ \text{для} \ \ 0 < n \le N$$
$$\mu_n = 0, \ \text{для} \ \ n = 0.$$

Расчетные формулы для данной системы имеют вид:

1 Трафик-интенсивность

$$\rho = \lambda / \mu.$$

2 Вероятность нахождения в системе точно n требований

$$P_n = (N - n + 1)\rho P_{n-1}, \ n = 1, 2, \ldots, N$$

или

$$P_n = \frac{N!}{(N-n)!}\rho^n P_0,\ \ n = 1,\ 2,\ \ldots,\ N;$$

$$P_0 = \frac{1}{1+\sum_{n=1}^{\infty}\frac{N!\rho^n}{(N-n)!}}.$$

3 Среднее число требований в системе

$$\overline{n} = \sum_{n=0}^{\infty} nP_n = N!P_0\sum_{n=0}^{\infty}\frac{n\rho^n}{(N-n)!} = N - \frac{1}{\rho}(1-P_0) = 1-\bar{l}.$$

4 Среднее число требований в очереди

$$\overline{m} = \sum_{n=2}^{N}(n-1)P_n = N!P_0\sum_{n=2}^{N}\frac{n-1}{(N-n)!}\rho^n =$$

$$= N - \frac{1+\rho}{\rho}(1-P_0) = \lambda(N-\overline{n})\bar{t}_f.$$

5 Среднее число требований, не находящихся в системе

$$\bar{l} = N - \overline{n}.$$

6 Среднее время нахождения в очереди

$$\bar{t}_f = \frac{1}{\lambda(N-\overline{n})}\sum_{n=2}^{N}(n-1)P_n = \frac{1}{\mu}\left[\frac{N}{1-P_0} - \frac{1+\rho}{\rho}\right].$$

7 Среднее время нахождения в системе

$$\bar{t}_s = \frac{n}{\lambda(N-\overline{n})} = \frac{1}{\mu}\left[\frac{N}{1-P_0} - \frac{\lambda}{\rho}\right].$$

8 Вероятность существования очереди

$$\underline{P} = 1 - P_0.$$

9 Вероятность отсутствия очереди

$$\text{Вер}\{n = 0\} = P_0.$$

4.4.4.6 Система $\dfrac{|M|M|L|}{|\infty|\lambda|\mu|T|1|0|}$

В отличие от других рассмотренных ранее систем в данном случае число каналов (приборов) не равно единице. Пусть в системе функционирует L параллельно работающих приборов. В этом случае, если число требований в системе $n \le L$, то очереди нет – все требования обслуживаются. Очередь возникает, если $n > L$.

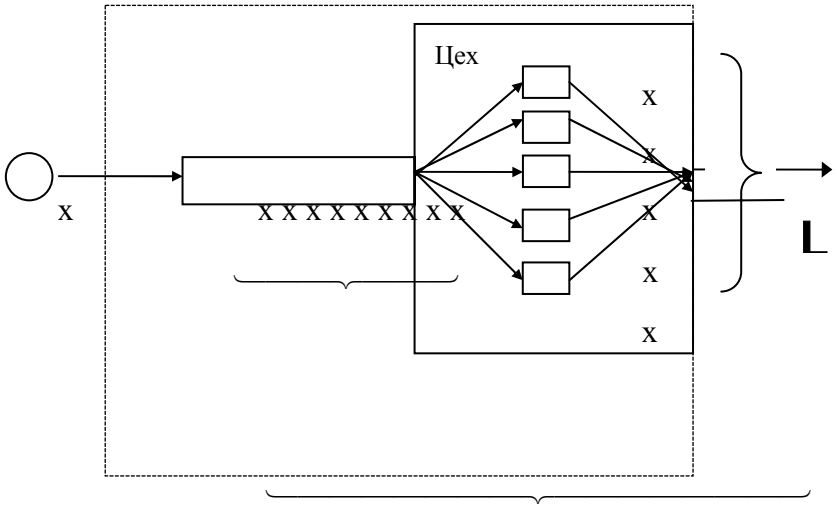
Примерами подобных систем могут быть аэропорты с несколькими посадочными площадками, автозаправочные станции, железнодорожные вокзалы с несколькими кассами, мастерские с несколькими работающими станками, швейные мастерские и др.

Схема такой системы изображена на рис. 4.12.

В таких системах массового обслуживания для того, чтобы наступил статический стационарный режим, должно выполняться соотношение $\frac{\lambda}{\mu L} < 1.$

Если трафик-интенсивность $\rho = \lambda/\mu$, как и раньше, то для рассматриваемых систем с α каналами для наступления статического стационарного режима, следовательно, должно выполняться соотношение

$$\rho/\alpha < 1 \qquad (4.39)$$



m

Рис. 4.12 Система с *L* каналами α

В противном случае очередь будет бесконечно расти. Действительно, так как λ – среднее число требований, поступающих в систему в единицу времени; μ – среднее число требований, которые обслуживаются одним прибором в единицу времени; μL – среднее число требований, обслуживаемых α приборами в единицу времени, то для того, чтобы очередь не росла бесконечно, необходимо, чтобы

$$\lambda < \mu L \text{ или } \lambda/\mu < L.$$

Отсюда и вытекает (4.39).

Расчет системы ведется по следующим формулам.

1 Коэффициент использования каждого канала (трафик-интенсивность)

$$\rho = \lambda / \mu .$$

2 Вероятность нахождения в системе точно *n* требований

$$; \qquad P_n = P_0 \frac{\rho^n}{n!}, \quad 1 \le n < L$$

$$P_n = P_0 \frac{\rho^n}{L! L^{n-1}}, \quad n \ge L$$

или

;

$$P_n = \frac{\rho}{n} P_{n-1}, \quad 1 \le n \le L$$

$$P_n = \frac{\rho}{\alpha} P_{n-1}, \quad n \ge L.$$

3 Вероятность отсутствия требований в системе

$$P_0 = \frac{1}{\dfrac{\rho^L}{L!(1-\rho/L)} + \sum_{n=0}^{L-1} \dfrac{\rho^n}{n!}}$$

или

$$P_0 = \frac{1}{\dfrac{\rho^\alpha}{L!(1-\rho/L)} + 1 + \rho + \dfrac{\rho^2}{2!} + \dfrac{\rho^3}{3!} + \ldots + \dfrac{\rho^{n-1}}{(L-1)!}}.$$

4 Среднее число требований в очереди

$$\overline{m} = \sum_{n=L+1}^{n} (n-L) P_n = \frac{\rho^{L+1}}{LL!(1-\rho/L)^2} P_0 = \frac{P^L \rho}{L(1-\rho/L)^2}.$$

5 Среднее число незанятых каналов

$$\overline{s} = \sum_{n=0}^{L} (L-n) P_n = \overline{m} + L - \overline{n} = M - \rho.$$

6 Среднее число требований в системе

$$\overline{n} = \sum_{n=0}^{\infty} n P_n = \overline{m} + L - \overline{s} = \overline{m} + \rho.$$

7 Вероятность существования очереди

$$\underline{P} = \text{Вер}\{n > L\} = \sum_{n=\alpha}^{\infty} P_n = P_0 \frac{\rho^\alpha}{L!(1-\rho/L)}.$$

Эта вероятность может быть определена также по формуле Эрланга

$$\underline{P} = \frac{\dfrac{\rho^2}{L!(1-\rho/L)}}{\dfrac{\rho^2}{L!(1-\rho/L)} + 1 + \rho + \dfrac{\rho^2}{L!} + \ldots + \dfrac{\rho^{L-1}}{(L-1)!}}.$$

8 Вероятность того, что время ожидания в очереди для одного требования не превышает t_f

$$F(t_f) = 1 - e^{-L\mu t_f (1-\rho/L)} P_0 \frac{\rho^L}{L!(1-\rho/L)^2}.$$

9 Среднее время ожидания в очереди

$$\bar{t}_f = \overline{m} / \lambda = \frac{\rho^L}{LL!\mu(1-\rho/L)^2} P_0.$$

Для определения $\bar{t}_f$ можно воспользоваться номограммой, позволяющей определить $t_f\mu$ по значениям N и ρ/N.

4.4.4.7 Система $\frac{|M|M|L|}{|N|\lambda|\mu|T|1|0|}$

Система характеризуется тем, что число требований, находящихся в ней, ограничено. Условная схема этой системы изображена на рис. 4.13.

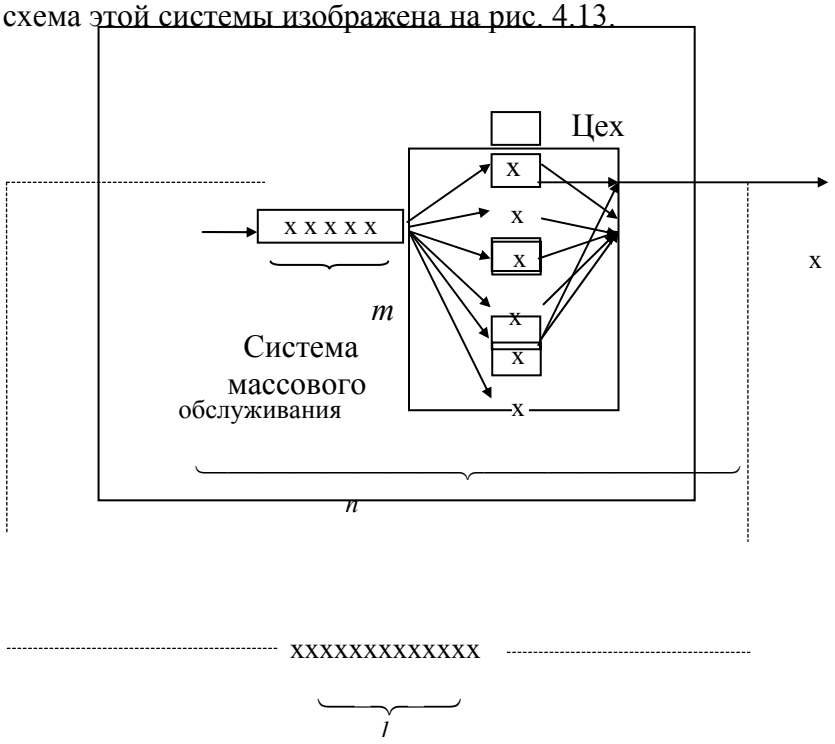


Рис. 4.13 Система с ограниченным максимальным числом требований и с *L* каналами

Здесь обозначено: m – число требований в очереди; n – число требований в системе; l – число требований вне системы массового обслуживания; N – общее число требований; L – общее число приборов (устройств обслуживания); s – число незанятых каналов (простаивающих устройств).

Примерами такой системы могут быть N станков, обслуживаемых L механиками; группа нефтеперерабатывающих заводов, обслуживаемых группой нефтедобывающих предприятий; топливозаправщики, обслуживающие эскадрилью бомбардировщиков и т.п.

Между переменными данной системы, очевидно, справедливы следующие соотношения

$$n - l = N;$$

$$m = \begin{cases} 0, \text{ если } n \le L; \\ n - L, \text{ если } n > L; \end{cases} \qquad s = \begin{cases} 0, \text{ если } n > L; \\ L - n, \text{ если } n \le L. \end{cases}$$

Процесс же массового обслуживания имеет в этом случае следующие реальные значения параметров λ_n, μ_n, зависящие от числа требований в системе

$$\lambda_n = \begin{cases} N\lambda & \text{при} \quad n = 0; \\ (N-n)\lambda & \text{при} \quad 1 \le n \le N; \end{cases} \qquad \mu_n = \begin{cases} 0 & \text{при} \quad n = 0; \\ n\mu & \text{при} \quad 1 \le n < L; \\ L\mu & \text{при} \quad L \le n \le N. \end{cases}$$

Можно показать, что

$$a_n = \begin{cases} 1 \text{ при } n = 0; \\ \dfrac{N-n+1}{n}\rho a_{n-1} \text{ при } n = 1, 2, \ldots, L-1; \\ \dfrac{N-n+1}{N}\rho a_{n-1} \text{ при } n = L, \ldots, N, \end{cases}$$

где

$$a_n \overset{\Delta}{=} P_n / P_0.$$

Расчет системы производится по следующим формулам:

1 Вероятность отсутствия требований в системе

$$P_0 = \frac{1}{1 + \sum_{n=1}^{N} a_n}.$$

2 Вероятность нахождения в системе точно n требований

$$P_n = P_0 a_n.$$

3 Коэффициент интенсивности обслуживания

$$\rho = \lambda / \mu .$$

4 Среднее число требований в очереди

$$\overline{m} = \sum_{n=L+1}^{N} (n-L) P_n.$$

5 Коэффициент простоя требований

$$k_{\text{пр}}^{\text{тр}} = \overline{m} / N.$$

6 Среднее число простоев приборов (оборудования)

$$\overline{s} = \sum_{n=0}^{L} (L-n) P_n.$$

7 Коэффициент простоя приборов (оборудования)

$$k_{\text{об}}^{\text{пр}} = \overline{s} / L.$$

8 Среднее число требований в системе

$$\overline{n} = \sum_{n=0}^{L} n P_n = \overline{m} + L - \overline{s}.$$

9 Среднее число требований, не находящихся в системе

$$\overline{l} = N - \overline{n}.$$

10 Вероятность существования очереди

$$P = \sum_{n=L}^{N} P_n = 1 - \sum_{n=0}^{L-1} P_n.$$

11 Среднее время ожидания в очереди

$$\bar{t}_f = \frac{\overline{m}}{\lambda(N - \bar{n})} = \frac{\overline{m}}{\lambda(L - \bar{s})}.$$

4.5 Системы с нетерпеливыми клиентами (системы без очередей)

4.5.1 СИСТЕМА $\frac{|M|M|L|}{|\infty|\lambda|\mu|H|1|0|}$

Рассматриваемая система массового обслуживания характеризуется тем, что, если число находящихся в цехе клиентов равно L и клиент уходит, то система его теряет. Таким образом, число клиентов всегда не больше обслуживающих устройств $n \le L$, т.е. очереди в такой системе быть не может.

Примером такой системы может служить перехват сообщений по радио, которые осуществляет группа приемников. Если в момент передачи все радиоприемники уже принимают сообщения, то данное сообщение будет потеряно. Другой пример, если на междугородной телефонной станции все каналы линии связи заняты, то соединение не возможно, а также, если в гостинице все места заняты, прием нового гостя не возможен и т.п.

Условная схема системы изображена на рис. 4.14.

В системе массового обслуживания без очередей основным показателем эффективности служит вероятность ухода требования (вероятность, что все устройства обслуживания заняты), которая обозначается V и определяется как $V = P_L$, где L – число обслуживающих устройств.

Эта вероятность характеризует соотношение входного потока требований и потока обслуженных требований. Кроме того, основным характерным показателем для этой системы является среднее число простоя (среднее число свободных устройств) $\bar{s}$, характеризующее загруженность цеха в среднем.

Расчет системы производится по следующим формулам.

1 Трафик-интенсивность

$$\rho = \lambda / \mu.$$

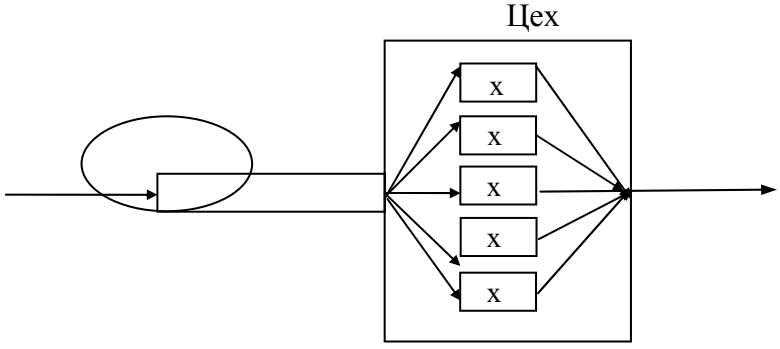
2 Вероятность того, что все обслуживающие устройства свободны

$$P_0 = \frac{1}{\sum_{n=0}^{L} \frac{1}{n!} \rho^n}.$$

3 Вероятность того, что в обслуживающей системе занято n аппаратов

$$P_n = \frac{P_0}{n!} \rho^n, \ n = 1, 2, \ldots, L.$$

Это так называемая формула Эрланга.



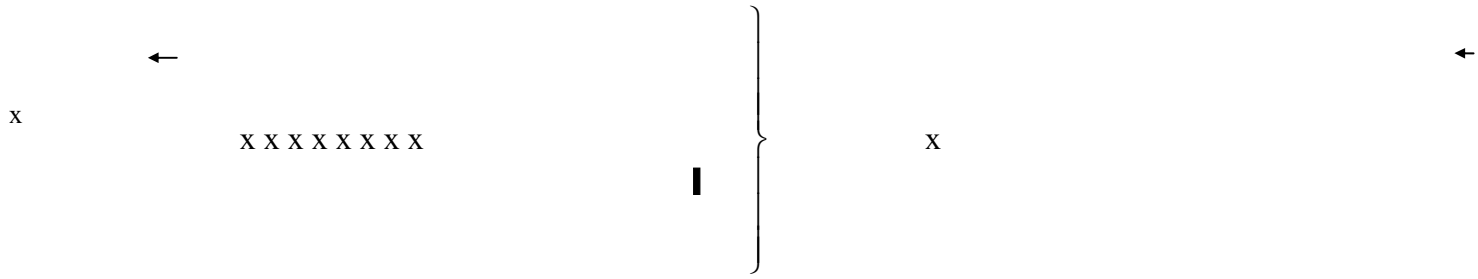


Рис. 4.14 Система без очередей

Вероятности P_n могут быть определены по рекуррентным формулам

$$P_n = \frac{\rho}{n} P_{n-1}, n = 1,2,\ldots, L.$$

4 Вероятность отказа очередному требованию в обслуживании

$$V = P_L = \frac{\rho^L \frac{1}{L!}}{\sum_{n=0}^{L} \frac{1}{n!}\rho^n}.$$

5 Среднее число занятых обслуживающих устройств

$$\overline{n} = \sum_{n=0}^{L} nP_0 = \sum_{n=0}^{L} \frac{1}{(n-1)!}\rho^n P_0.$$

6 Среднее число простоев оборудования

$$\overline{s} = L - \overline{n}.$$

7 Коэффициент простоя оборудования

$$k_{\text{пр}}^{\text{об}} = \overline{s} / L.$$

8 Коэффициент занятости оборудования

$$k_{\text{зан}}^{\text{об}} = \overline{n} / L.$$

4.5.2 СИСТЕМА $\frac{|M|M|\infty|}{|\infty|\lambda|\mu|H|1|0|}$

В данной системе в отличие от предыдущих число приборов бесконечно велико. Такой предельный случай часто используется в расчетах, когда число обслуживающих устройств велико. Подобная идеализация позволяет получить более простые расчетные формулы. При таком варианте, как и в предыдущем случае, очереди нет, но здесь потеря требования невозможна. Число требований в системе всегда равно числу занятых обслуживающих устройств.

Расчет системы ведется по следующим формулам.

1 Неустановившаяся вероятность того, что в момент времени t заняты n приборов при условии, что в начальный момент времени они свободны

$$P_n(t) = \frac{1}{n!}\rho^n (1 - e^{-\mu t})^n e^{-\rho(1-\mu t)}, \ n = 0, 1, 2, \ldots$$

2 Среднее число обслуживающих устройств в момент времени t, если при $t = 0$ они все свободны

$$\overline{n}(t) = \rho(1 - e^{-\mu t}).$$

3 Установившаяся вероятность, что в системе будут работать n аппаратов, каково бы ни было начальное состояние системы

$$P_n = \frac{1}{n!}\rho^n e^{-\rho}.$$

4 Вероятность того, что в установившемся режиме все аппараты свободны

$$P_0 = e^{-\rho}.$$

4.6 Системы с очередями и частичными потерями

4.6.1 СИСТЕМА $\frac{|M|M|L|}{|\infty|\lambda|\mu|H|1|0|}$

Возможными вариантами такой системы могут быть системы с условиями:

– $n \le N$, где N – предельное число требований в системе (требование не встает в очередь, т.е. теряется для обслуживающей системы, если число таких требований, находящихся в системе, равно N);

– $m \le M$, где M – предельное число требований в очереди (здесь требование не встает в очередь, т.е. теряется для обслуживающей системы, если число требований в очереди уже равно M);

– $t_0 \le T_0$, где T_0 – предельное время ожидания в очереди (в этом случае требование выходит из очереди, если его время ожидания начала обслуживания уже равно T_0);

– $t_c \le T_c$, где T_c – предельное время пребывания в системе (требование покидает систему, если оно находится в ней уже время T_c)

Могут быть и любые другие условия, при которых требование покидает систему или вообще не встает в очередь.

Частным случаем рассматриваемой системы является система, где требование встает в очередь только в том случае, если эта очередь меньше предельного значения N:

$$m < M,$$

где m – длина очереди.

Если $m = M$, то требование в очередь не встает. Очевидно, что

$$m = \begin{cases} 0, & \text{если } n \le L; \\ n - L, & \text{если } L < n \le N, \end{cases}$$

где L – число приборов; n – число требований в системе.

Обслуживающая система отказывает в обслуживании при $n = L + M$, т.е. L требований обслуживается и M требований стоит в очереди.

Схема системы массового обслуживания с очередью и потерями представлена на рис. 4.15.

Расчетные формулы системы имеют вид:

1 Трафик-интенсивность

$$\rho = \lambda / \mu.$$

2 Вероятность нахождения в системе n требований

$$P_n = \begin{cases} \dfrac{1}{n!}\rho^n P_0, \ 1 \le n < L; \\ \dfrac{P_0}{L!L^{n-L}}\rho^n, \ L \le n \le L+M. \end{cases}$$

3 Вероятность того, что в системе работают n приборов

$$P_n = \begin{cases} \dfrac{1}{n!}\rho^n P_0, \ 1 \le n < L; \\ \dfrac{P_0}{L!L^{n-L}}\rho^n P_0, \ L \le n \le L+M. \end{cases}$$

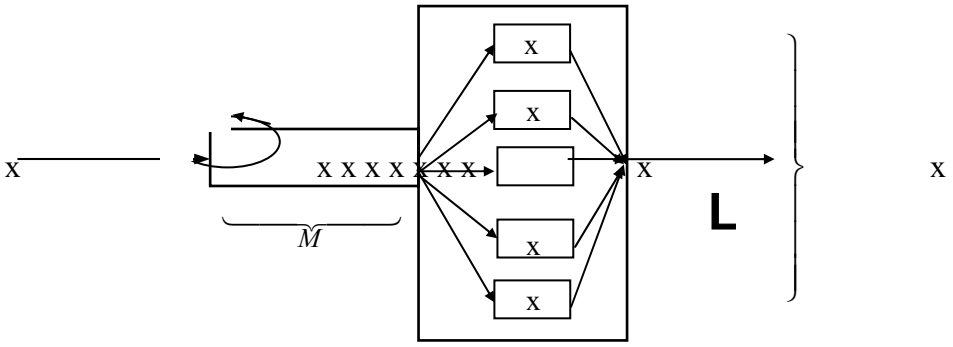


Рис. 4.15 Система с очередью и потерями

4 Вероятность того, что в очереди стоит $m = n - L$ требований

$$P_m = \begin{cases} 0, \ \text{если} \ \ 1 \le n < L; \\ \dfrac{1}{L!L^{n-L}}\rho^n, \ L \le n \le L+M. \end{cases}$$

5 Вероятность, что все обслуживающие аппараты свободны

$$P_0 = \frac{1}{\sum\limits_{n=0}^{L-1}\dfrac{1}{n!}\rho^n + \dfrac{1}{L!(1-\rho/L)}\rho^n\left[1-(\rho/L)^{M+1}\right]},$$

где L – максимальное число аппаратов; M – наибольшая допустимая длина очереди.

6 Вероятность того, что поступившее требование получит отказ, т.е. не будет принято на обслуживание:

$$P_{L+M} = \frac{P_0}{L!L^M}\rho^{L+M}.$$

7 Вероятность того, что все аппараты будут заняты (вероятность образования очереди)

$$\Pi = P_L\frac{1-(\rho/L)^{M+1}}{1-\rho/L}.$$

8 Вероятность того, что время ожидания в очереди будет больше t_f

$$\theta(t_f) = \frac{ne^{-\mu L t_f}}{1-(\rho/L)^{M+1}}\sum_{n=0}^{M-1}\frac{(\mu L t_f)^n}{n!}\left[(\rho/L)^n - (\rho/L)^M\right].$$

9 Средняя длина очереди

$$\overline{m} = \frac{P_L}{(1-\rho/L)^2}\left[\frac{\rho}{L} - (M+1)(\rho/L)^{M+1} + M(\rho/L)^{M+2}\right].$$

10 Среднее число свободных приборов

$$\overline{s} = \sum_{n=0}^{L-1}\frac{L-1}{n!}\rho^n P_0 .$$

4.7 Пример расчета системы массового обслуживания с несколькими каналами и ограниченным числом клиентов

Эта система по принятой классификации $\frac{|M|M|L|}{|N|\lambda|\mu|T|1|0|}$ изображена на рис. 4.12. и описана в п. 4.4.4.7.

Пусть в цехе имеется $N = 16$ ткацких станков и $L = 4$ механиков. Таким образом, максимальное число требований в системе может быть $n = 16$, если же их число $n \ge L$, возникает очередь. Параметр пуассоновского входного потока $\lambda = 10$ 1/сут, а параметр обслуживания $\mu = 100$ 1/сут, следовательно, трафик-интенсивность $\rho = 0{,}1$.

Для последующих расчетов необходимо определить коэффициенты a_n, $n = 0, 1, \ldots, 20$, по формуле

$$a_n =; \begin{cases} 1 \quad \text{при} \qquad n = 0; \\ \dfrac{N-n+1}{n}\rho a_{n-1}, \quad 1 \le n \le 4 \\ \dfrac{N-n+1}{N}\rho a_{n-1}, \quad 5 \le n \le 20 \end{cases}$$

В результате расчетов получено $a_0 = 1$; $a_1 = 2$; $a_2 = 1{,}9$; $a_3 = 1{,}14$; $a_4 = 0{,}48$; $a_5 = 0{,}178$; $a_6 = 0{,}067$; $a_7 = 0{,}023$; $a_8 = 0{,}0074$; $a_9 = 0{,}0022$; $a_{10} = 0{,}000605$; $a_{11} = 0{,}00015$; a_{12}, a_{13}, a_{14}, a_{15}, $a_{16} < 0{,}0001$.

$$\sum_{n=1}^{16} a_n = 2 + 1{,}9 + 1{,}14 + 0{,}48 + 0{,}178 + 0{,}067 + 0{,}023 + 0{,}0074 + 0{,}0022 +$$
$$+ 0{,}000605 + 0{,}00015 + \ldots = 5{,}791.$$

ВЕРОЯТНОСТЬ, ЧТО СТАНКИ НЕ БУДУТ РАБОТАТЬ (ОТСУТСТВИЕ ТРЕБОВАНИЙ В СИСТЕМЕ)

$$P_0 = \frac{1}{1 + \sum_{n=1}^{16} a_n} = \frac{1}{1 + 5{,}791} = 0{,}147.$$

Вероятности работы n станков (нахождения в системе n требований)

$P_n = a_n P_0$:

$P_1 = P_0 a_1 = 0{,}294$; $P_2 = P_0 a_2 = 0{,}279$; $P_3 = P_0 a_3 = 0{,}168$;

$P_4 = P_0 a_4 = 0{,}007$; $P_5 = P a_5 = 0{,}026$; $P_6 = P_0 a_6 = 0{,}0098$;

$P_7 = P_0 a_7 = 0{,}0033$; $P_8 = P_0 a_8 = 0{,}00109$; $P_9 = P_0 a_9 = 0{,}0003, \ldots$

Среднее число станков, ждущих работу (требований в очереди)

$$\overline{m} = \sum (n-4)P_n = P_5 + 2P_6 + 3P_7 + 4P_8 + 5P_9 + 6P_{10} + 7P_{11} +$$
$$+ 8P_{12} + 9P_{13} + 10P_{14} + 11P_{15} + 12P_{16};$$
$$\overline{m} = 0{,}026 + 2 \cdot 0{,}0098 + 3 \cdot 0{,}00339 + 4 \cdot 0{,}00109 + 5 \cdot 0{,}0003 + \ldots = 0{,}057.$$

Среднее число простоев ткацких станков

$$\overline{s} = \sum_{n=0}^{4} (4-n)P_n = 4P_0 + 3P_1 + 2P_2 + P_4; \quad \overline{s} = 2{,}196.$$

Среднее число работающих станков (требований в системе)

$$\overline{n} = L + \overline{m} - \overline{s}; \quad \overline{n} = 4 + 0{,}057 - 2{,}196 = 1{,}861.$$

Коэффициент простоя механиков (приборов)

$$k_{\text{пр}}^{\text{об}} = \overline{s} / L; \quad k_{\text{пр}}^{\text{об}} = \ 0{,}549.$$

Коэффициент простоя станков (требований)

$$k_{\text{пр}}^{\text{тр}} = \overline{m} / N; \quad k_{\text{пр}}^{\text{тр}} \ = 0{,}0036.$$

Вероятность существования очереди

$$P = 1 - \sum_{n=0}^{3} P_n; \quad P = 1 - 0{,}147 - 0{,}279 - 0{,}168 = 0{,}112.$$

Среднее время ожидания в очереди

$$t_f = \frac{\overline{m}}{\lambda(N - \overline{n})}; \quad t_f = \frac{0{,}057}{10(16 - 1{,}861)} 0{,}000403 \text{ сут} = 5{,}8 \text{ мин}.$$

4.8 Моделирование процессов массового обслуживания

Системы массового обслуживания далеко не исчерпываются теми проанализированными случаями, которые были рассмотрены, но, к сожалению, только эти случаи и поддаются пока аналитическому решению. Более сложные случаи могут быть исследованы лишь с помощью методов моделирования с использованием вычислительной техники.

На рис. 4.16 представлена схема моделирования системы массового обслуживания.

Расчет ведется через малый промежуток времени Δt, который задается счетчиком времени 1. Этот счетчик определяет также текущее время t.

Генератор случайных чисел 2 генерирует интервал времени между появлениями очередных требований.

При этом плотность распределения интервалов времени должна соответствовать реальной плотности распределения системы реально функционирующей или гипотетической системы.

Как только прошло время Δt, появляется новый клиент. При этом блок 4 проверяет, есть ли очередь или нет, другими словами есть ли свободный прибор.

Если такой прибор есть, то немедленно начинается обслуживание, если же свободных приборов нет, клиент решает, встать ли ему в очередь или уйти.

Если очередь велика ($n = N$), проверку этого осуществляет блок 5, то клиент попадает в очередь. При этом он решает, возвратится ли ему снова на обслуживание в эту систему или уйти в другую (блок

6). Если клиент покидает систему, то он навсегда теряется для нее, и в этом случае система несет экономические убытки.

Если же очередь не велика, то в блоке 7 проверяются другие возможные причины, препятствующие клиенту встать в очередь. Например, это может быть отсутствие нужного мастера. В блоке 8 снова решается вопрос о повторном в будущем использовании этой же самой системы массового обслуживания для удовлетворения заявки на обслуживание.

Если клиент решил встать в очередь, то последняя увеличивается на единицу (блок 9).

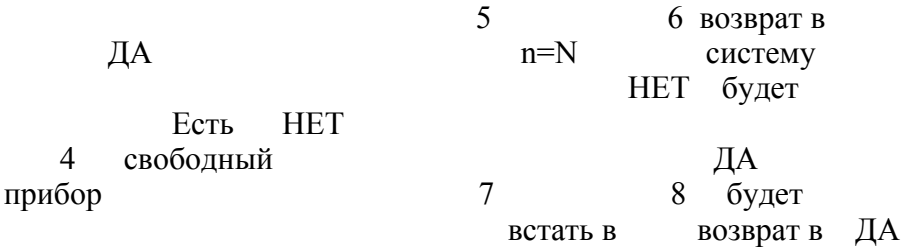
Блок 10 моделирует ожидание в очереди. Здесь для каждого клиента подсчитывается время ожидания в очереди t_0^i. Через каждый маленький интервал времени Δt блок 11 проверяет, не освободился ли прибор. В случае положительного решения этого вопроса очередь уменьшается на единицу (блок 12) и требование поступает на обслуживание в цех. Если же свободного прибора нет, блок 13 проверяет, целесообразно ли продолжать стоять в очереди. Чаще всего это означает, не превышает ли уже ожидание предельного времени. Очевидно, могут быть и любые другие условия.

Если в блоке 13 принимается решение покинуть очередь, то эта очередь уменьшается на одного клиента (блок 14), затем в блоке 15 решается вопрос о целесообразности в будущем снова попасть в данную обслуживающую систему.

Блок 16 уменьшает число свободных приборов на единицу, и клиент попадает в цех на обслуживание.

Счетчик
времени

1
Δt Δt
24
2 Δt Плотность интервалов
между клиентами
НЕТ
3 Время
НЕТ Δt 23 возврата
прошло наступило


5
6 возврат в
ДА
n=N
систему
НЕТ будет
Есть НЕТ
4 свободный
ДА
прибор
7
8 будет
встать в
возврат в ДА

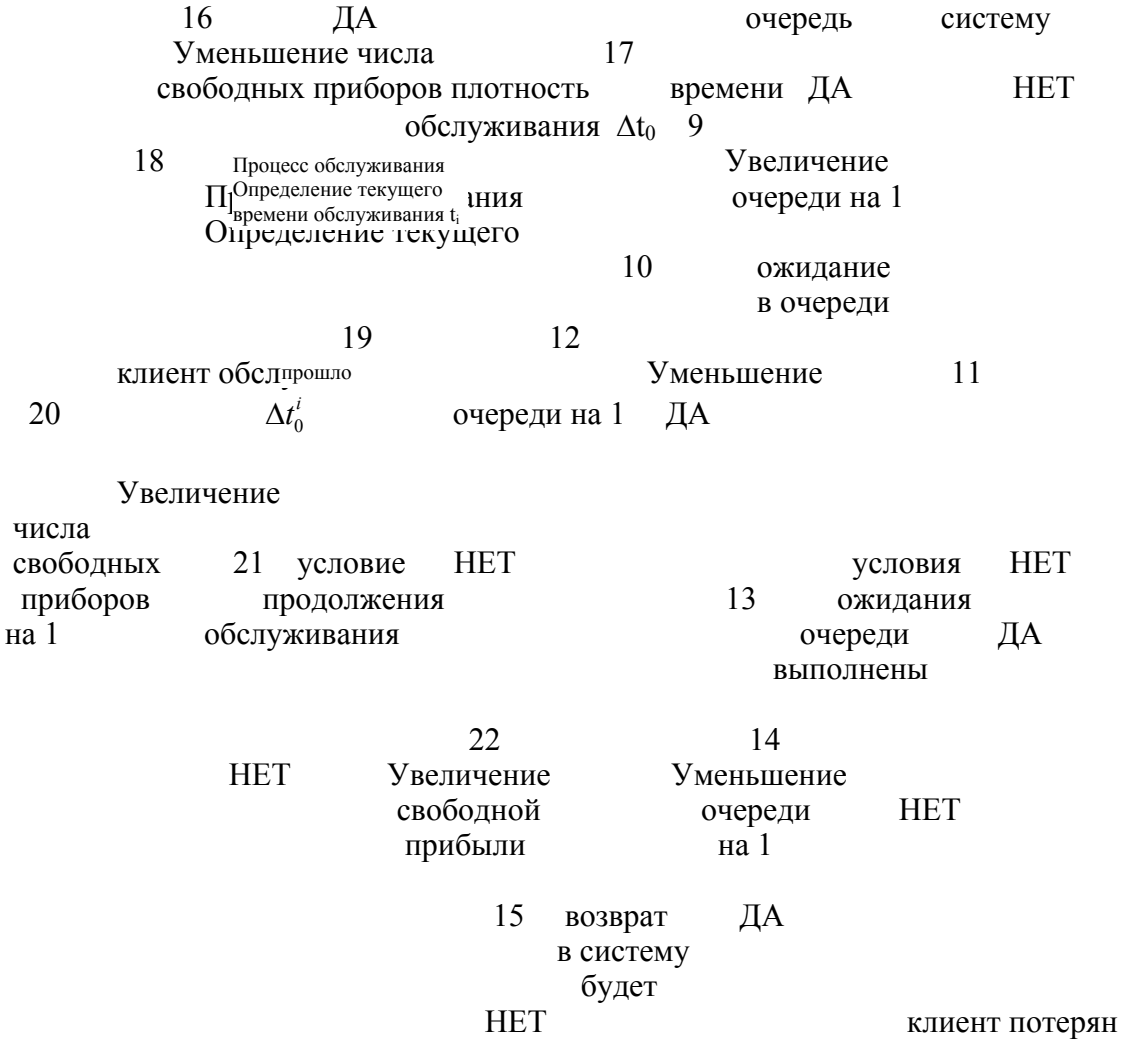


Рис. 4.15 Моделирование системы массового обслуживания

Блок 18 имитирует процесс обслуживания. Сюда поступает время обслуживания очередного требования, которое выдает генератор 17.

Генератор реализует тот закон распределения вероятности, который может быть тождественен плотности распределения времени обслуживания реального (анализируемого) или гипотетического производства.

Через каждый интервал времени Δt_0 блок 19 проверяет, не окончилось ли обслуживание одного из клиентов. Если обслуживание окончилось, то число свободных приборов увеличивается и это делает блок 20. Если же обслуживание не окончилось, то в блоке 21 решается вопрос о продолжении нахождения в системе обслуживания. Чаще всего проверяется, не превосходит ли суммарное время нахождения в системе (в очереди и в цехе) предельно допустимого для клиента времени. Если клиент уходит, то число свободных приборов увеличивается на единицу (блок 22).

Генератор 24 генерирует продолжительность времени (случайную величину), спустя которое очередное требование возвращается в систему на обслуживание. Наступление момента возвращения каждого требования контролируется блоком 23.

Рассмотренная блок-схема (рис. 4.16) является блок-схемой имитационного функционирования системы массового обслуживания. Имитация осуществляется следующим образом. Датчик случайных чисел 2 через одинаковые промежутки времени генерирует либо 0, либо 1, что означает, появилось ли за данный промежуток времени новое требование или нет. Образующийся поток требований такой же, как и на реальном объекте, естественно, в статическом смысле, т.е. вероятность появления того или иного числа требований в единицу времени одинакова, что на реальном объекте, что на имитационной модели.

Датчик случайных чисел 17 каждому требованию, "поступающему" в цех, ставит в соответствие время, которое он будет обслуживаться в цеху. При этом значения этих времен в статистическом смысле такие же, как и на реальном объекте.

Чем точнее заданы распределения вероятностей датчиков 2 и 17 с соответствующими реальными законами распределения вероятностей появления новых требований и времен обслуживания, полученными на имитационной модели, тем больше совпадают с реальными процессами обслуживания на исследуемой системе.

Теперь рассмотрим вопросы расчета операционных показателей по данным имитационного эксперимента. Пусть этот эксперимент длится время $T_{\text{общ}}$. Далее замеряются время T_0, когда в системе не было ни одного требования, и времена T_n, $n = 1, 2, \ldots$, когда в системе было соответственно 1, 2, … требований. Тогда вероятность того, что в системе будет n требований $P_n = T_n/T_{\text{общ}}$.

Вероятность $F(n)$, что в системе будет не больше n требований, определяется формулой

$$F(n) = \sum_{i=1}^{n} P_i.$$

Вероятность $\theta(n)$, что в системе будет более n требований

$$\theta(n) = 1 - F(n).$$

Вероятность того, что в системе не будет ни одного требования

$$P_0 = T_0 / T_{\text{общ}}.$$

Среднее число требований в системе равно

$$\overline{n} = \sum_{n=0}^{\infty} nP_n.$$

Пусть L – число обслуживающих клиентов. Тогда среднее число клиентов в очереди

$$\overline{m} = \sum_{n=L+1}^{N} (n-L)P_n,$$

где N – предельное число клиентов в системе, при превышении которого клиент не встает в очередь.

Среднее число простоев приборов: $\overline{s} = \sum_{n=0}^{L} (L-n)P_n.$

Коэффициент простоя требования: $k_{\text{пр}}^{\text{тр}} = \overline{m} / N$.

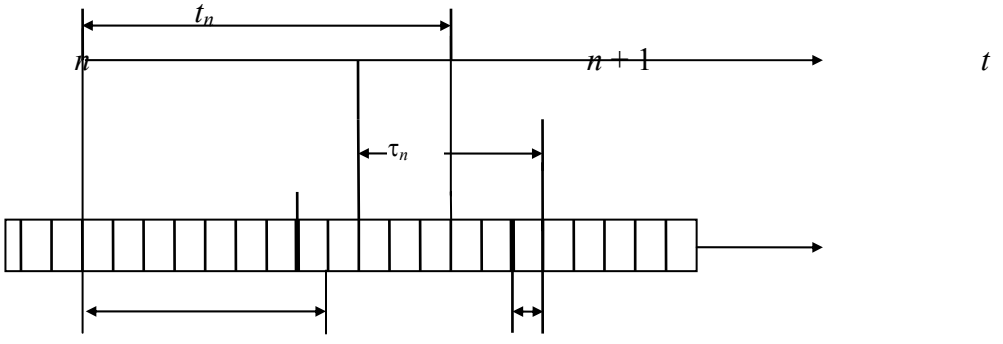
Коэффициент простоя приборов (оборудования): $k_{\text{пр}}^{\text{об}} = \overline{s} / L.$

Вероятность существования очереди: $P = 1 - \sum_{n=0}^{L-1} P_n.$

Время ожидания в очереди определяется по данным имитационного эксперимента. На рис. 4.17 представлены два варианта обслуживания требований в системе массового обслуживания.

В обоих вариантах рис. 4.17, *а* и рис. 4.17, *б* в момент 1 поступает требование n, в момент 3 поступает требование $(n + 1)$. В это время прибор все еще обслуживает требование $n - 1$, обслуживание которого заканчивается в момент 2.

Начиная с момента 2, требование n поступает в систему и начинает обслуживаться. Таким образом ω_n – это время ожидания требования n в очереди.



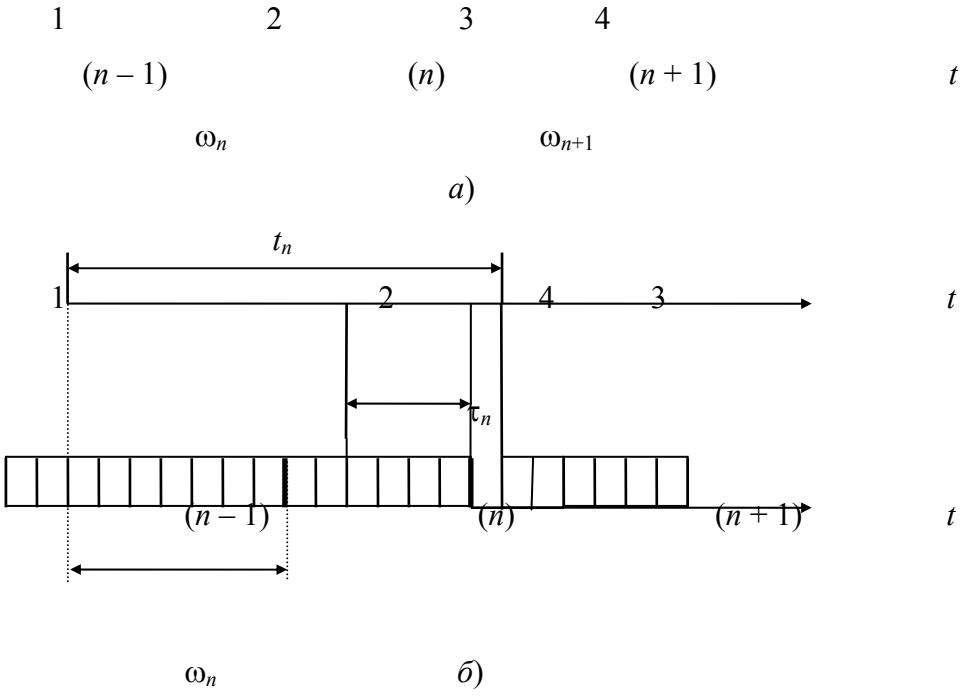


Рис. 4.17 К расчету времени ожидания:
а – время обслуживания велико; *б* – время обслуживания мало

В момент 3 поступает следующее требование $n + 1$. На рис. 4.17, *а* время обслуживания τ_n велико, и требование $n + 1$ также ожидает в очереди время ω_n прежде, чем начнет обслуживаться. Это время рассчитывается по формуле $\omega_{n+1} = \omega_n + \tau_n - t_n$, где t_n – время между последовательными поступлениями требований.

На рис. 4.17, *б* время обслуживания τ_n мало, и время ожидания $(n + 1)$-го требования в очереди равно нулю. Требование начинает обслуживаться сразу. Таким образом

$$\omega_{n+1} = \begin{cases} \omega_n + \tau_n - t_n, & \text{если} \quad \omega_n + \tau_n > t_n; \\ 0, & \text{если} \quad \omega_n + \tau_n \le 0, \quad n = 1, 2, \ldots \end{cases}$$

Начальные условия для этой рекуррентной формулы $\omega_1 \equiv 0$, т.е. первое требование начинает обслуживаться без задержки.

Время нахождения в системе определяется формулой

$$t_n^{\text{c}} = t_n + \omega_n.$$

Среднее время ожидания в очереди

$$\bar{t}_f = \sum_{n=1}^{R} \omega_n / R.$$

Среднее время пребывания в системе

$$\bar{t}_s = \sum_{n=1}^{R} t_n^{\text{c}} / R.$$

4.9 Оптимизация процессов массового обслуживания

Оптимизация систем массового обслуживания, как и всякая оптимизация, заключается в выборе варьируемых параметров (называемых управлением), при которых целевая функция принимает минимальное (максимальное) значение и удовлетворяется система ограничений и условий.

В системах массового обслуживания в качестве варьируемых параметров обычно принимают: L – число обслуживаемых приборов; $1/\mu$ – среднее время обслуживания клиента одним прибором.

Выбор более дорогого прибора с меньшим временем обслуживания удорожает простой прибор, но уменьшает время ожидания клиента в очереди.

Иногда в качестве варьируемого параметра может быть использована структура s системы (число очередей в системе, последовательное или параллельное расположение приборов и др.).

Также в качестве варьируемого параметра иногда используется число M – максимально допустимое число клиентов, после которого клиент получает отказ в обслуживании и др.

Пусть U – вектор варьируемых параметров, $U = (L, \mu, s, M)$. Очевидно, каждая из варьируемых переменных вектора U изменяется в определенных пределах:

$$\begin{aligned} &L_{\min} \le L \le L_{\max};\\ &\mu_{\min} \le \mu \le \mu_{\max};\\ &s \in A;\\ &M_{\min} \le M \le M_{\max}. \end{aligned} \qquad (4.40)$$

Причем $M_{\max}$ может быть и бесконечностью.

Теория массового обслуживания устанавливает связь (математическую модель) между варьируемыми переменными и вектором операционных показателей y. Эта связь в операторной форме может быть представлена в виде

$$y = A(U). \qquad (4.41)$$

Оператор A представляет собой как систему аналитических формул, так и может быть задан в алгоритмической форме (в виде имитационных алгоритмов моделирования).

Целевая функция $Q(y)$ оценивает численно, насколько операционные показатели хороши.

Задача оптимизации ставится как задача отыскания такого вектора U^*, при котором целевая функция $Q(y)$ примет минимальное (максимальное) значение, при этом y определяется по (4.41), а варьируемые переменные удовлетворяют (4.40).

Иногда накладываются дополнительные технологические ограничения, которые также должны удовлетворяться, что сужает область возможных изменений вектора U.

В качестве целевой функции часто выбирается некоторый экономический критерий, который учитывает средние потери от ожидания в очереди клиентами и средние потери от простоя оборудования:

$$Q(y) = (C_1\overline{m} + C_2\overline{s})T,$$

где $\overline{m}T$ – среднее время, потерянное в очереди клиентами за время имитации T; $\overline{s}T$ – среднее время, потерянное из-за простоев приборов; C_1, C_2 – стоимости единицы времени из-за ожидания клиентов в очереди и простоя приборов соответственно.

Если C_1 и C_2 постоянны, то обычно рассматривают целевую функцию в виде

$$Q(y) = C_1\overline{m} + C_2\overline{s}.$$

Величины C_1 и C_2 могут вычисляться довольно сложно, зависеть от многих факторов, быть нелинейными функциями многих переменных (времени непрерывной работы, субъективных факторов усталости обслуживающего персонала и т.д.).

Особенно трудно адекватно учесть величину C_1, так как из-за простоев в очереди клиенты могут теряться (заходить в другую систему обслуживания, не обращаться в будущем к данной системе обслуживания, получать меньше, чем они намеривались и т.п.).

5 УПРАВЛЕНИЕ ЗАПАСАМИ

Запас обозначает ресурсы, готовые к употреблению, но временно не используемые.

Если количество ресурсов можно регулировать, то возникает проблема управления запасами.

5.1 Задачи управления запасами

Основная задача управления запасами заключается в том, что необходимо принять некоторое управляющее воздействие относительно запасов, заключающееся в решении вопроса, сколько и когда надо иметь этих запасов. Имеющиеся запасы, как правило, хранятся на складах, поэтому задачи управления запасами называют еще задачами работы и управления складов. Решение этих задач должно быть таковым, чтобы минимизировалась некоторая целевая функция, которая характеризует эффективность работы склада.

Обычно целевой функцией является экономическая функция, включающая следующие составляющие:

1 Затраты на хранение запаса. Эти затраты пропорциональны объему запаса и времени хранения. Здесь, как правило, учитываются:

а) затраты на складские операции (погрузка, разгрузка, портальные краны и т.п.);

б) стоимость хранения (плата за аренду помещения);

в) страхование;

г) потери, порчи продукции;

д) затраты на учет.

2 Затраты на компенсацию потерь от недостатка запасов (потери от дефицита).

Учет этих потерь ведется одним из двух способов:

а) потери пропорциональны объему недостающих ресурсов и времени, в течение которого существует дефицит;

б) потери оцениваются постоянным штрафом, например, в размере потери прибыли и компенсации морального ущерба, нанесенного предприятию.

В принципе могут быть и другие составляющие работы склада. Так, можно учитывать затраты на размещение заказа на найм и обучение рабочих при изменении характера производства, от объема закупок может зависеть цена ресурсов и т.п.

В качестве управляющих воздействий (варьируемых переменных) выбирают либо объем поступающих ресурсов, либо периодичность или моменты времени поступления ресурсов.

Задачи управления запасами возникают всегда и повсеместно, при любом производстве и коммерческой организации. Так, энергетики должны решить, сколько ввести в систему электростанций, какой мощности включить генератор; университет должен решить, сколько готовить тех или иных специалистов; тренер должен решить, сколько готовить запасных игроков; режиссер – сколько нужно дублеров на спектакль; военные – сколько необходимо военных запасов; финансист – каков должен быть резервный капитал в банке и т.п.

Задачи теории управления запасами подразделяются на следующие категории: детерминированные; случайные, статически устойчивые (стационарные); случайные, но статически неустойчивые (нестационарные); с неизвестным характером спроса.

С другой точки зрения, задачи теории управления запасами можно подразделить по:

а) характеру пополнения запасов (отсрочка, непрерывный, периодический, осуществляется через некоторые интервалы времени);

б) характеру ограничений (максимальный или минимальный объемы, вес, время изготовления, наличие финансовых средств, стоимость перевозок и др.).

5.2 Алгоритмы (характер) пополнения запасов

Пополнение и расходование запасов часто изображают графически (рис. 5.1).

На рис. 5.1 изображен процесс расходования запасов на производство со склада. Начальный запас товара в момент t_0 составляет $S_н$. Рассматривается промежуток времени T, в течение которого в моменты t_1, t_2, t_3, t_4 запасы отпускаются со склада. Ломаная линия показывает изменение запасов на складе. В конечный момент времени запасы равны $S_к$. Соединив $S_к$ и $S_н$, получают линию, которая показана пунктиром. Чаще всего считается, что именно эта идеализированная линия отражает отпуск товара со склада. Этот прием отвечает математическому описанию задачи складирования при достаточно малых потребностях.

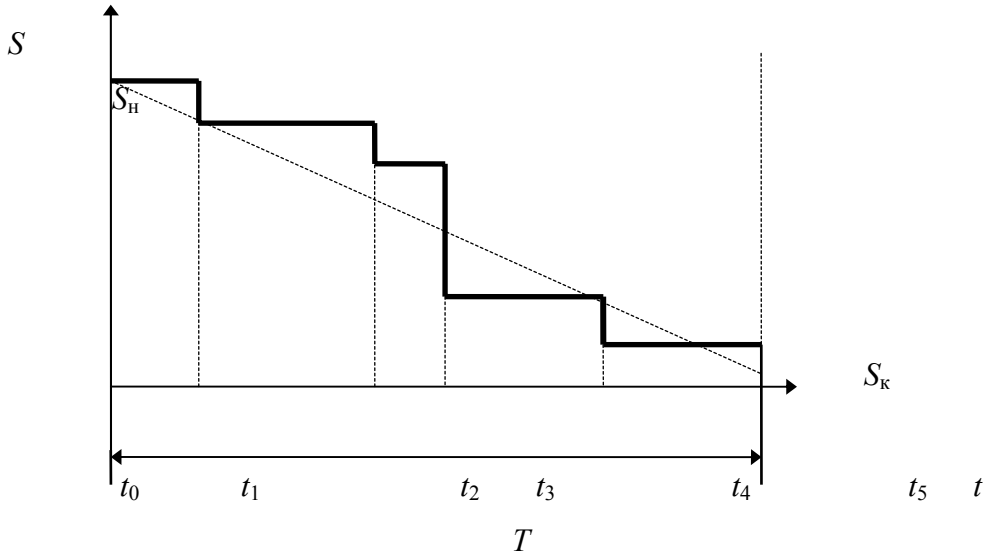
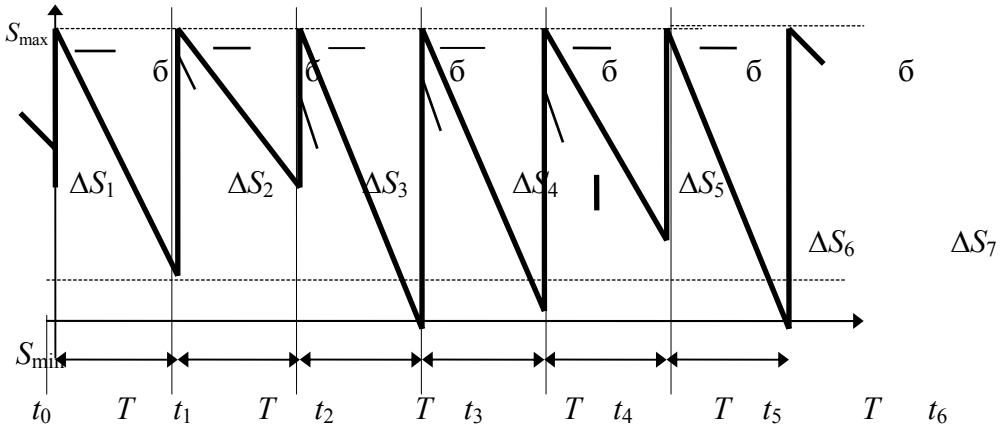
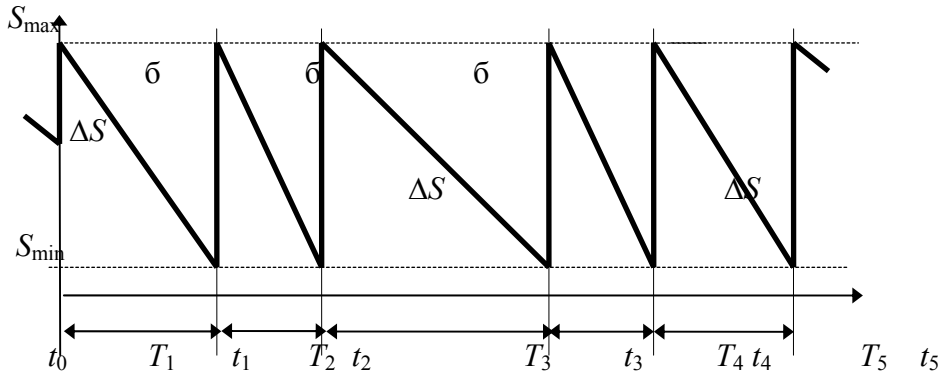


Рис. 5.1 Процесс расходования запасов



а)



б)

Рис. 5.2 Мгновенное (дискретное) пополнение запасов:
а – тип I (периодический), T = const; *б* – тип II (релаксационный), ΔS = const

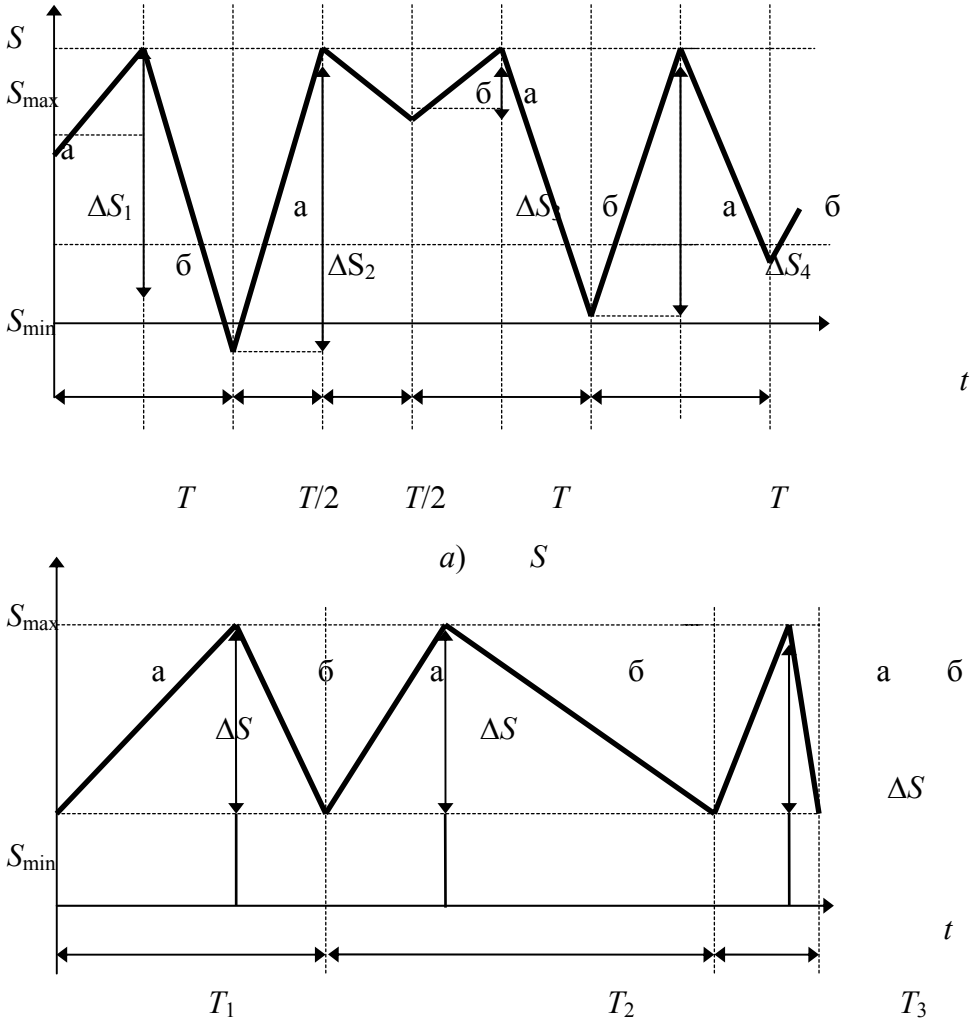
Два основных алгоритма управления запасами при дискретном характере пополнения этих запасов изображены на рис. 5.2.

Первый тип (рис. 5.2, *а*) называется периодическим и обозначается тип I, второй (рис. 5.2, *б*) – релаксационным, тип II. Первый тип характеризуется периодическим пополнением запасов. В этом случае склад пополняется запасами до предельной величины S_{max} через равные промежутки времени, интервалы времени T одинаковы, пополнения ΔS_i разные.

Во втором случае склад пополняется каждый раз на величину ΔS, когда запасы на нем достигают нижнего критического уровня S_{min}. Здесь интервалы времени T_i разные, величина пополнения одинакова ($\Delta S = S_{max} - S_{min}$).

Непрерывная система пополнения запасов показана на рис. 5.3, она также может быть двух типов: тип I – периодический (рис. 5.3, *а*), тип II – релаксационный (рис. 5.3, *б*).

Первый тип соответствует случаю, когда весь диапазон изменения времени делится на равные участки $T/2$. При этом ветвь "б" кривой изменения запасов (рис. 5.3, *а*) соответствует потреблению запасов, а ветвь "а" на участке $T/2$ – пополнению запасов на складе, причем может одновременно происходить расходование запасов на производство. В данном случае интервалы $T/2$ постоянны, а изменения запасов на складе ΔS_i разные.



а) *S*

б)

Рис. 5.3 Непрерывное пополнение запасов:
а – тип I (периодический), T = const;
б – тип II (релаксационный), ΔS = const

При релаксационном характере пополнения склада (рис. 5.3, *б*, участок "а") непосредственное его пополнение начинается, если запас на нем достигает минимальной величины. При этом $\Delta S = S_{max} - S_{min}$ является постоянной величиной, а интервалы времени T_i разные.

Очевидно, движение товаров на складе при дискретном характере пополнения запасов является идеализированным, так как мгновенное пополнение склада невозможно. Задержка в пополнении товара может привести к серьезным последствиям в производстве. В связи с этим приказ о пополнении запасов на складе должен отдаваться заранее, когда еще не известно точно, какое именно количество товара необходимо будет приобрести (для управления типа I) или через какое время запас точно достигнет мини-

мального уровня S_{min} (для управления типа II). Все это приводит к тому, что необходимо построить "опережающий" алгоритм, т.е. алгоритм пополнения склада с прогнозом.

Пусть τ – интервал времени между решениями о приобретении ресурсов (заключение договора) до момента прихода товара на склад, являющийся постоянной величиной, не зависящей от объема поставленного товара.

Характер периодического алгоритма управления типа I с прогнозом изображен на рис. 5.4.

В рассматриваемом алгоритме интервалы времени T между поставками известны. Задача заключается в том, что в момент времени t_k, называемый контрольным моментом времени, отстоящем на интервал τ от момента привоза товара, необходимо спрогнозировать величину ΔS_i, т.е. сколько товара требуется привезти. При прогнозе надо учитывать скорость уменьшения товара на складе.

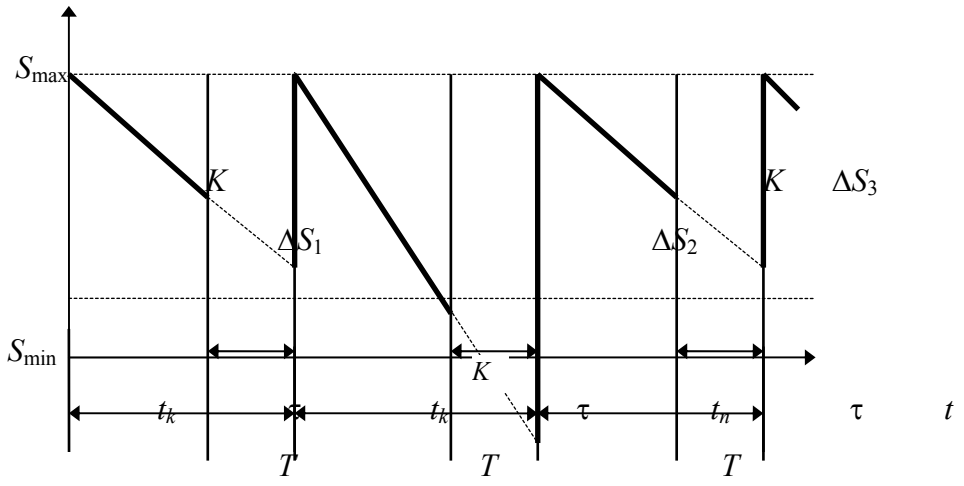


Рис. 5.4 Периодический алгоритм с прогнозом

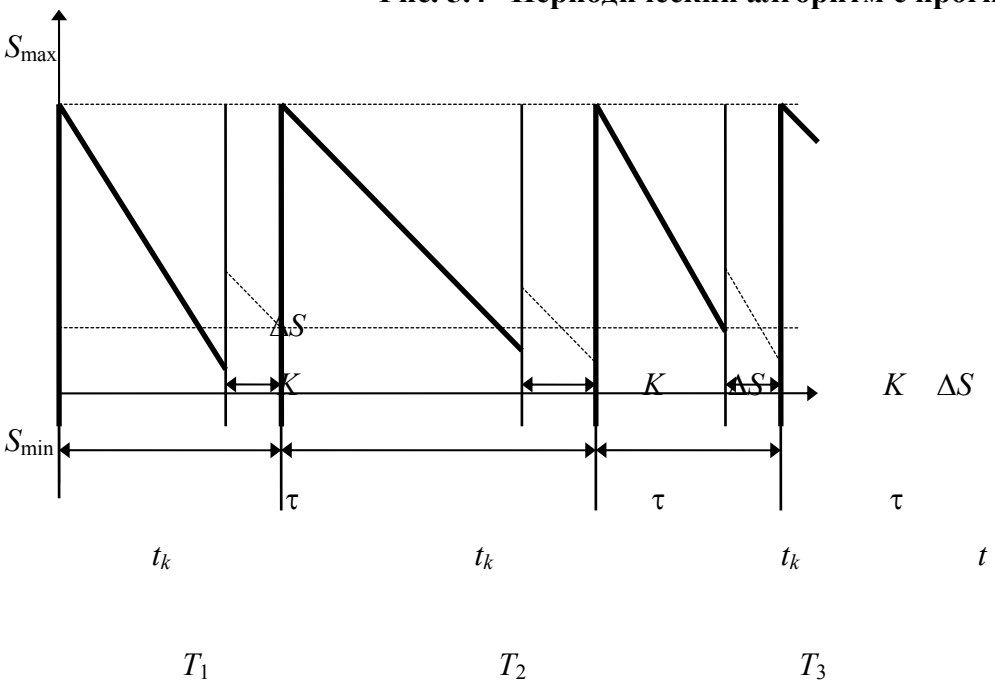


Рис. 5.5 Релаксационный алгоритм прогноза

Релаксационный алгоритм управления с прогнозом запасами на складе (тип II) изображен на рис. 5.5.

В рассматриваемом алгоритме необходимо, наблюдая за скоростью изменения запасов на складе, спрогнозировать контрольный момент времени такой, чтобы через интервал τ запас на складе достиг минимального уровня S_{min}. Именно в этот момент нужно заключить договор на поставку товара в одном и том же количестве ΔS.

Таким образом, в периодическом алгоритме прогнозируется количество товара ΔS_i, в релаксационном – интервалы времени T_i (моменты закупки товара).

На практике довольно часто применяется алгоритм, при котором запасы пополняются, если уровень запасов достиг некоторого критического значения $S_{кр}$. Характер этого алгоритма изображен на рис. 5.6.

Согласно рассматриваемому алгоритму, решение о пополнении запасов (договор на поставку ресурсов) принимается в момент времени $t_{кр}$, когда уровень запасов на складе достиг критического значения $S_{кр}$. Через некоторое время – время задержки τ товар поступает на склад. Пополнение ΔS всегда одинаково, поэтому уровень товара на складе может после пополнения оказаться как выше, так и ниже предельного уровня S_{max}.

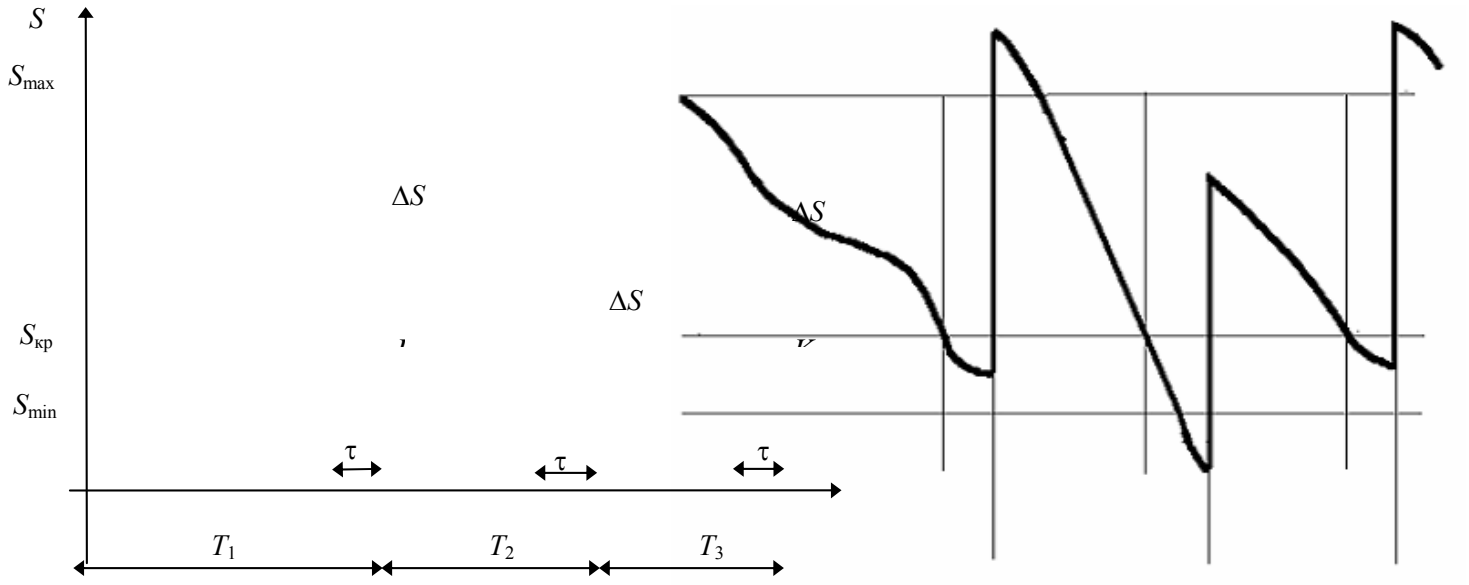


Рис. 5.6 Алгоритм по достижению уровня пополнения

Расположение склада

Склад резервов может иметь двойное назначение. В одном случае (рис. 5.7) склад служит хранилищем сырья, ресурсов, полуфабрикатов, которые идут дальше на производство (или перепродажу). Такой склад называется складом на входе. При этом вертикальный участок (рис. 5.2) соответствует закупке товаров, участок "б" – отпуску ресурсов со склада на производство.

На рис. 5.3 участок "а" соответствует закупке, а участок "б" – отпуску на производство. На участке "а" может одновременно быть и закупка и отпуск на производство.

Под производством понимается перепродажа, оптовая или розничная торговля и т.д.

Затраты на хранение включают в этом случае в себя постоянную составляющую (собственно закупку) и переменную (затраты на охрану, аренду, рабочую силу), обычно пропорциональную времени и количеству хранящихся ресурсов.

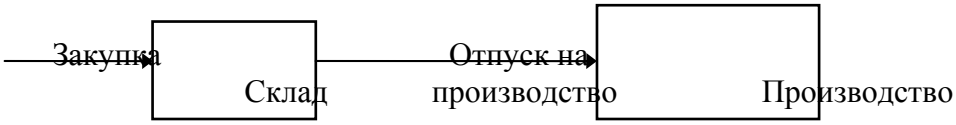


Рис. 5.7 Блок-схема склада на входе

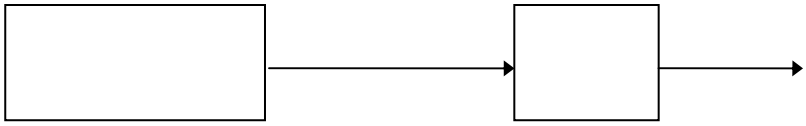


Рис. 5.8 Блок-схема склада на выходе

Склад готовой продукции располагается на выходе производства (рис. 5.8).

На таком складе поступления – это выход продукции с завода, выход со склада – отпуск товара потребителям.

На рис. 5.3 участок "а" соответствует поступлению готовой продукции (производству), участок "б" – отпуску товара со склада. Естественно, что и на участке "а" и на участке "б" отпуск товара и поступление на склад могут быть одновременными. Однако, участок "а" в случае варианта склада на выходе соответствует интенсивному производству, а участок "б" – интенсивному отпуску. В случае склада на входе смысл этих участков противоположен.

Рис. 5.2 при рассмотрении склада на выходе соответствует случаю, когда производство очень быстрое (мгновенное). Это весьма идеализированный склад. Обычно для склада на выходе характерно изменение запасов на нем в соответствии с рис. 5.3, для склада на входе – с рис. 5.2.

Задачи управления запасами можно классифицировать по ряду признаков, которые характеризуют их со всех сторон. Основными характеристиками являются следующие:

– сбыт, который может быть детерминированным (Д) и случайным (С);

– поступление товаров – дискретное (Д), непрерывное (Н);

– алгоритм управления запасами – периодический (П), непериодический (Н);

– количество продукции – одна ("1"), много (*n*);

– затраты на хранение – есть (+), нет (–);

– затраты на дефицит – есть (+), нет (–).

В соответствии с принятыми характеристиками задача управления запасами может быть записана как

ν =	сбыт	поступление	алгоритм	количество продукции	затраты	
					хранение	дефицит

5.3 Детерминированная задача управления запасами с единственным видом продукции

Рассмотренная задача имеет несколько вариантов.

5.3.1 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА. ЗАТРАТЫ НА ХРАНЕНИЕ ПРОПОРЦИОНАЛЬНЫ РАЗМЕРУ ПАРТИИ (СЕРИИ). СКЛАД НА ВХОДЕ. ДЕФИЦИТ ОТСУТСТВУЕТ

По принятой классификации задача обозначается как

$$\nu = / Д / Д / П / 1 / + / – /.$$

Данную задачу удобно рассмотреть на примере работы склада по периодическому типу алгоритма на протяжении достаточно большого времени θ.

При этом считают, что а) продукция однородна; б) максимальный уровень запасов фиксирован; в) ресурсы поступают на склад партиями (дискретно) или сериями. Также считают, что а) стоимость доставки партии C_p; б) стоимость хранения единицы продукции в единицу времени (например, сутки) C_s; в) общее потребление товара за время θ равно N.

Задача управления запасами состоит в том, чтобы найти уровень пополнения запасов ΔS и период пополнения T, при которых затраты на создание и хранение N ресурсов были бы минимальны.

Пусть расход ресурса (например, деталей) равномерен. Тогда в единицу времени расходуется $n = N/\theta$ ресурсов.

Не уменьшая общности, можно считать, что $S_{\min} = 0$. В этом случае изменение запаса на периоде T будет выглядеть в соответствии с тем, как изображено на рис. 5.9.

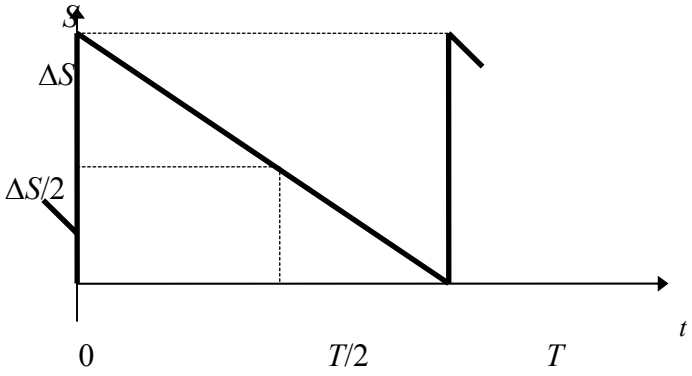


Рис. 5.9 Изменение запаса на периоде

Искомая величина ΔS называется иногда экономической партией или серией. Этот термин особенно удобен при дискретном характере ресурсов, например, если ресурсы – детали.

Число серий (периодов) на интервале θ можно определить либо по формуле

$$N = \theta / T, \tag{5.1}$$

либо по формуле

$$n = N/\Delta S. \tag{5.2}$$

Целевая функция, представляющая собой затраты на серию, для рассматриваемой задачи является аддитивной и включает в себя затраты на доставку партии ΔS – q_1, затраты на хранение – q_2. Таким образом, целевая функция записывается в виде

$$q = q_1 + q_2,$$

где очевидно

$$q_1 = \mathrm{C}_p;\ q_2 = (\Delta S/2) T \mathrm{C}_s.$$

Здесь обозначено: C_p – стоимость доставки партии ΔS; $\Delta S/2$ – средний за период T запас на складе; C_s – стоимость хранения единицы продукции за единицу времени.

Общие затраты на хранение за время θ равны

$$Q = \mathrm{C}_p n + \frac{\Delta S}{2} T \mathrm{C}_s n.$$

Используя (5.1), (5.2), выражение для общих затрат преобразуется к виду

$$Q = \mathrm{C}_p \frac{N}{\Delta S} + \frac{\Delta S}{2} T \mathrm{C}_s \frac{\theta}{T}$$

или

$$Q = \frac{\mathrm{C}_p N}{\Delta S} + \frac{\theta \mathrm{C}_s}{2} \Delta S\,. \tag{5.3}$$

Таким образом, общие затраты (5.3) состоят из суммы двух составляющих, одна из которых обратно пропорциональна уровню пополнения запасов ΔS: $Q_1 = \mathrm{C}_p N / \Delta S$, другая же пропорциональна этому уровню: $Q_2 = \theta \mathrm{C}_s \Delta S / 2.$

Минимум общих затрат на хранение, т.е. минимум целевой функции Q находится из условия $\partial Q / \partial S = 0$:

$$-\frac{\mathrm{C}_p N}{\Delta S^2} + \frac{\theta \mathrm{C}_s}{2} = 0, \tag{5.4}$$

откуда

$$\Delta S = \sqrt{2\frac{C_p}{C_s}\frac{N}{\theta}}. \qquad (5.5)$$

Выражение (5.4) определяет уровень пополнения склада или оптимальное число деталей в экономической партии (серии), из него видно, что $C_p N/\Delta S = \theta C_s = \Delta S/2$, т.е. составляющие общих затрат равны между собой $Q_1 = Q_2$.

Это свидетельствует о том, что минимум затрат при решении задачи управления запасами имеет место тогда, когда общие затраты $Q_1 = C_p N/\Delta S$ равны общим затратам $Q_2 = \theta C_s \Delta S/2$ на хранение этих запасов (рис. 5.10).

Оптимальное число серий (периодов) n^* согласно (5.2) равно

$$n^* = N/\Delta S^*.$$

Оптимальный интервал цикла

$$T^* = \theta/n^* = \frac{\theta}{N}\Delta S^*.$$

Оптимальное значение целевой функции $Q^* = 2Q_2^* = 2\frac{\theta C_s}{2}\Delta S^*$.

С учетом (5.5) $Q^* = \sqrt{2N\theta C_p C_s}$.

Интерес представляет оценка влияния погрешности в определении ΔS на целевую функцию.

Средняя погрешность при ошибке в определении ΔS на 10 % дает следующую погрешность целевой функции

$$\Delta\overline{Q} = \frac{1}{2}\left[Q(\Delta S^* + 0{,}1\Delta S^*) - Q(\Delta S^*) + Q(\Delta S^* - 0{,}1\Delta S^*) - Q(\Delta S^*)\right] =$$

$$= \frac{1}{2}\left[Q(1{,}1\Delta S^*) - 2Q(\Delta S^*) + Q(0{,}9\Delta S^*)\right].$$

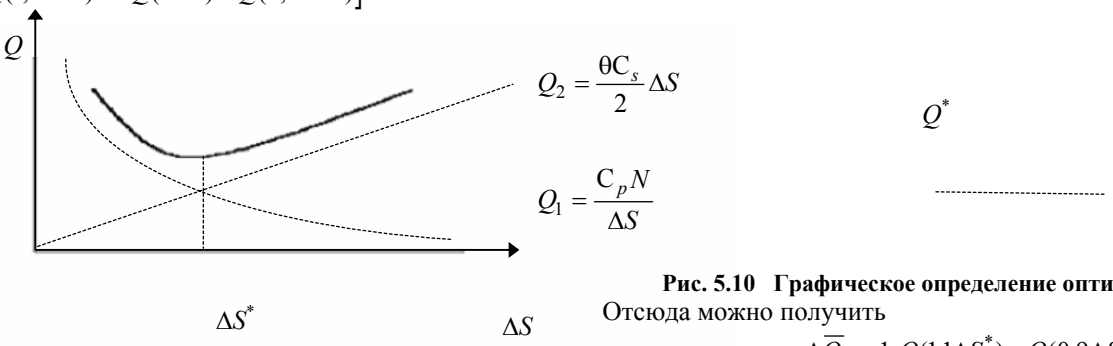


Рис. 5.10 Графическое определение оптимальной партии ΔS

Отсюда можно получить

$$\frac{\Delta\overline{Q}}{Q^*} = \frac{1}{2}\frac{Q(1{,}1\Delta S^*) + Q(0{,}9\Delta S^*)}{Q^*} - 1.$$

5.3.2 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА. СКЛАД НА ВХОДЕ. НАЛИЧИЕ ДЕФИЦИТА.

$\nu = |Д|Д|П|1|+|+|.$

Для данной задачи полагают, что а) продукция однородна; б) максимальный уровень запасов фиксирован; в) ресурсы поступают на склад партиями (или сериями); г) максимальный уровень дефицита фиксирован.

Изменение запаса на складе для периода T представлено на рис. 5.11.

В момент времени $t = 0$ уровень запаса равен S_{max}, в момент $t = t_{кр}$ уровень запаса на складе достигает условного нуля (нуля или минимального уровня), после чего производство невозможно вследствие дефицита ресурсов.

На интервале времени T_2 (от $t_{кр}$ до T) дефицит растет и в момент $t = T$ достигает максимального значения $S_{деф}$. После этого на склад поступают ресурсы в размере ΔS, которые перекрывают дефицит, и уровень ресурсов на складе снова становится S_{max}:

$$\Delta S = S_{\max} - S_{\text{деф}}.$$

Общие затраты в этом случае складываются из Q_1 – затрат на доставку ресурсов, Q_2 – затрат на хранение на интервале времени T_1, а также Q_3 – потерь из-за дефицита на интервале времени T_2:

$$Q = Q_1 + Q_2 + Q_3 .$$

Очевидно $Q_1 = C_p n$, где C_p – стоимость доставки партии, а n – число партий, которые определяются, как и раньше

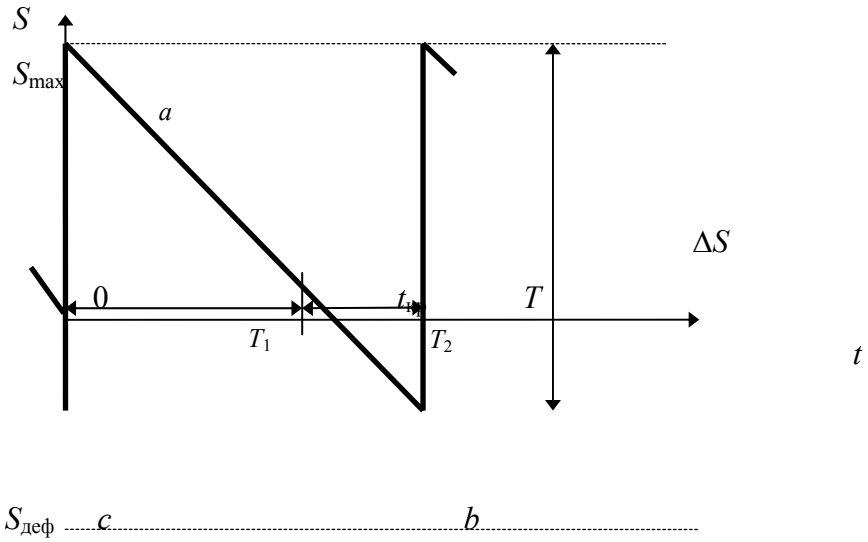


Рис. 5.11 Изменение запасов на складе

$$n = \frac{\theta}{T} = \frac{N}{\Delta S}. \qquad (5.6)$$

Здесь θ – общий интервал времени, в течение которого должно быть израсходовано на производство N ресурсов.

Затраты на хранение, как и раньше, пропорциональны треугольнику "$0at_{\text{кр}}$":

$$Q_2 = \frac{1}{2} S_{\max} T_1 C_s n, \qquad (5.7)$$

где C_s – стоимость хранения единицы продукции в единицу времени.

Убытки из-за дефицита пропорциональны треугольнику "$t_{\text{кр}}bT$", т.е. пропорциональны дефициту и времени дефицита с коэффициентом пропорциональности $C_{\text{д}}$

$$Q_3 = \frac{1}{2} S_{\text{деф}} T_2 C_{\text{д}} n$$

или

$$Q_3 = \frac{1}{2} (\Delta S - S_{\max}) T_2 C_{\text{д}} n,$$

где $C_{\text{д}}$ – цена дефицита единицы продукции в единицу времени.

Из подобия треугольников "$a0t_{\text{кр}}$" и "abc" можно записать

$$\frac{T_1}{T} = \frac{S_{\max}}{\Delta S}$$

или $T_1 = TS_{\max} / \Delta S.$ И тогда затраты на хранение, определяемые по (5.7) с учетом (5.6), можно рассчитать по выражению

$$Q_2 = \frac{1}{2}\frac{S_{\max}^2}{\Delta S}\mathrm{C}_s Tn = \frac{1}{2}\frac{S_{\max}^2 \theta \mathrm{C}_s}{\Delta S}.$$

Из подобия треугольников "*abc*" и "$t_{\text{кр}}Tb$" следует, что

$$\frac{T_2}{T} = \frac{S_{\text{деф}}}{\Delta S}$$

и соответственно

$$T_2 = T\frac{S_{\text{деф}}}{\Delta S} = T\frac{\Delta S - S_{\max}}{\Delta S}. \qquad (5.9)$$

Стоимость убытков (5.8) с учетом (5.9) определяется соотношением

$$Q_3 = \frac{1}{2}\frac{(\Delta S - S_{\max})^2}{\Delta S} T\mathrm{C}_{\text{д}} n = \frac{1}{2}\frac{(\Delta S - S_{\max})^2 \theta \mathrm{C}_{\text{д}}}{\Delta S}.$$

Таким образом, общие затраты определяются целевой функцией

$$Q(\Delta S,\ S_{\max}) = \frac{\mathrm{C}_p N}{\Delta S} + \frac{S_{\max}^2 \theta \mathrm{C}_s}{2\Delta S} + \frac{1}{2}\frac{(\Delta S - S_{\max})^2 \theta \mathrm{C}_{\text{д}}}{\Delta S}.$$

Оптимальные значения ΔS^* и $S_{\max}^*$ определяются из условий

$$\frac{\partial Q}{\partial \Delta S} = 0;\quad \frac{\partial Q}{\partial S_{\max}} = 0.$$

Решение этой системы дает

$$\begin{cases} \Delta S^* = \sqrt{\dfrac{2N}{\theta}\dfrac{\mathrm{C}_p}{\mathrm{C}_s}}\sqrt{\dfrac{\mathrm{C}_s + \mathrm{C}_{\text{д}}}{\mathrm{C}_{\text{д}}}}; \\ S_{\max}^* = \sqrt{\dfrac{2N}{\theta}\dfrac{\mathrm{C}_p}{\mathrm{C}_s}}\sqrt{\dfrac{\mathrm{C}_{\text{д}}}{\mathrm{C}_s + \mathrm{C}_{\text{д}}}} = \Delta S^*\dfrac{\mathrm{C}_{\text{д}}}{\mathrm{C}_s + \mathrm{C}_{\text{д}}}. \end{cases} \qquad (5.10)$$

Отношение

$$\rho = \frac{S_{\max}^*}{\Delta S^*} \qquad (5.11)$$

называется плотностью убытков.

Так как $\frac{S_{\max}^*}{\Delta S^*} = \frac{T_1}{T}$, то $\rho = \frac{T_1}{T}$, а $1 - \rho = \frac{T_2}{T}$,

откуда и вытекает физический смысл плотности убытков, который заключается в том, что в течение (1 – ρ) доли от T будет дефицит товара – $T_2 = (1 - \rho)T$. Из (5.10) плотность убытков можно определить так же, как

$$\rho = \frac{\mathrm{C}_{\text{д}}}{\mathrm{C}_s + \mathrm{C}_{\text{д}}}.$$

Таким образом, плотность убытков зависит от соотношения цены хранения товаров C_p и цены платы за дефицит товаров $C_д$. Она изменяется в пределах $0 \le \rho \le 1$. Если плата за дефицит товаров равна нулю, то $\rho = 0$, если эта плата стремится к бесконечности, то $\rho = 1$.

Используя (5.6), получают выражение для оптимального времени цикла

$$T^* = \theta / n^* = \theta \Delta S^* / N,$$

а, используя (5.10) – следующее выражение

$$T^* = \sqrt{\frac{2\theta}{N}\frac{C_p}{C_s}}\sqrt{\frac{C_s + C_д}{C_д}}.$$

В соответствии с выведенными соотношениями оптимальное значение целевой функции имеет вид

$$Q^* = Q(\Delta S^*, S^*_{\max}) = \sqrt{2N\theta C_s C_p}\sqrt{\frac{C_д}{C_s + C_д}}.$$

В заключении следует еще раз выписать основные расчетные формулы алгоритма:

1 Плотность убытков $\rho = \dfrac{C_д}{C_s + C_д}$.

2 Максимальный уровень ресурсов $S^*_{\max} = \Delta S^* \rho$.

3 Оптимальное поступление на склад $\Delta S^* = \sqrt{\dfrac{2N}{\theta}\dfrac{C_p}{C_s}}\dfrac{1}{\sqrt{\rho}}$.

4 Оптимальное время периода $T^* = \sqrt{\dfrac{2\theta}{N}\dfrac{C_p}{C_s}}\dfrac{1}{\sqrt{\rho}}$.

5 Оптимальное число серий $n^* = \theta / T^* = N / \Delta\rho^*$.

6 Оптимальное значение целевой функции $Q^* = \sqrt{2N\theta C_s C_p}\sqrt{\rho}$.

При отсутствии дефицита, т.е. когда $\rho = 1$ перечисленные формулы совпадают с формулами раздела 5.1, т.е.

$$S^*_{\max} = S^1_{\max}\rho; \quad \Delta S^* = \Delta S^1 1/\sqrt{\rho};$$
$$T^* = T^1 \; 1/\rho; \quad \theta^* = \theta^1\sqrt{\rho},$$

где $S^1_{\max}, \Delta S^1, T^1, \theta^1$ – соответствующие величины при отсутствии дефицита.

Как видно, с увеличением дефицита потери увеличиваются, как и $\sqrt{\rho}$. При заданной величине $\rho = \rho$ в соответствии с (5.11) должно соблюдаться соотношение $S^*_{\max} = \rho\Delta S^*$.

5.3.3 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА. СКЛАД НА ВЫХОДЕ. НАЛИЧИЕ ДЕФИЦИТА

При рассмотрении этой задачи полагают, что а) продукция однородна; б) максимальный уровень запаса и дефицита не фиксирован.

Изменение запасов на складе для периода T изображается графиком, представленным на рис. 5.12.

Уровень запасов в начальный момент времени $t = 0$ равен нулю, затем в течение времени T_1 он возрастает до максимальной величины $S_{\max}$, после чего в течение периода времени T_2 убывает до нуля. Последнее свидетельствует о том, что в течение T_2 производства либо не было, либо его темп был недостаточен и запасы на складе уменьшились. Начиная с момента C (рис. 5.12), запасов на складе нет, поэтому растет дефицит продукции (на рисунке это отображается увеличением отрицательного "запаса" за время

T_3 – на отрезке CF). Производство начинает работать и на склад поступают изделия, что свидетельствует об уменьшении дефицита на участке $FД$ (рис. 5.12). В конце интервала T_4 дефицит исчезает.

Для решения поставленной задачи считают, что а) стоимость производства одной партии C_p; б) стоимость хранения единицы продукции в единицу времени C_s; в) потери от дефицита единицы продукции в единицу времени $C_д$; г) спрос на продукцию из склада постоянен и равен единице продукции в единицу времени: $\omega = N/\theta$.

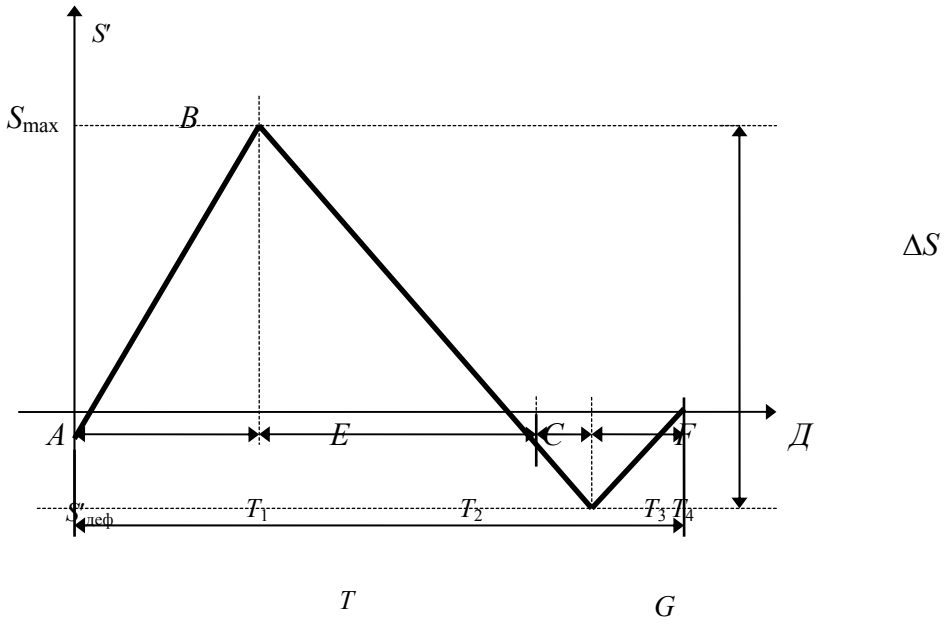


Рис. 5.12 Изменение запасов на складе

Кроме того, считают, что производство продукции осуществляется со скоростью k на участках AE и $FД$ и не производится вообще на участке EF (рис. 5.12).

В этом случае величина запасов S определяется соотношениями:

$$S = \begin{cases} (k-\omega)t \ \text{ при } \ 0 \le t \le T_1; \\ S_{\max} - \omega(t - T_1) \ \text{ при } \ T_1 \le t \le T_1 + T_2; \\ -\omega(t - T_1 - T_2) \ \text{ при } \ T_1 + T_2 \le t \le T_1 + T_2 + T_3; \\ -S_{\text{деф}} + (k-\omega)(t - T_1 - T_2 - T_3) \ \text{ при } \ T_1 + T_2 + T_3 \le t \le T_1 + T_2 + T_3 + T_4. \end{cases}$$

Для временных точек E, C, F, $Д$ соответственно имеют

$$\begin{aligned} &(k-\omega)T_1 = S_{\max}; \\ &S_{\max} - \omega T_2 = 0; \\ &-\omega T_3 = -S_{\text{деф}}; \\ &-S_{\text{деф}} + (k-\omega)T_4 = 0. \end{aligned} \qquad (5.13)$$

Общие затраты на одном цикле, как и в предыдущих случаях, складываются из затрат на изготовление продукции q_1, затрат на хранение q_2, потерь из-за дефицита q_3 и составляют

$$q = q_1 + q_2 + q_3.$$

Затраты на изготовление продукции равны $q_1 = C_p$,

где $n = \theta / T = N / \Delta S, \ \ \Delta S = S_{\max} - S_{\text{деф}}.$

Затраты на хранение $q_2 = \frac{1}{2} S_{\max}(T_1 + T_2)C_s.$

Убытки из-за дефицита пропорциональны треугольнику *СДG* (рис. 5.12) и определяются как $q_3 = \frac{1}{2} S_{\text{деф}}(T_3 + T_4)\text{C}_\text{д}$.

Таким образом, общие затраты составят

$$q = \left[\text{C}_p + \frac{1}{2} S_{\max}(T_1 + T_2)\text{C}_s + \frac{1}{2} S_{\text{деф}}(T_3 + T_4)\text{C}_\text{д}\right]. \qquad (5.14)$$

Неизвестными в (5.14) являются $S_{\max}$, T_1, T_2, $S_{\text{деф}}$, T_3, T_4. Однако условие (5.13) устанавливает четыре связи и, следовательно, степень свободы системы (5.13), (5.14) равна двум.

Рассматриваемую задачу управления запасами удобно решать методом неопределенных множителей Лагранжа. Для этого составляется вспомогательная функция

$$\overline{q}(T_1,\ T_2,\ S_{\text{деф}},\ T_3,\ T_4,\ S_{\max}) = q + \sum_{i=1}^{4} \lambda_i \varphi_i, \qquad (5.15)$$

где $\varphi_i = 0$ – условия (5.13).

Дифференцируя (5.15) по переменным $S_{\max}$, T_1, T_2, $S_{\text{деф}}$, T_3, T_4 и используя (5.13), определяют значения переменных, минимизирующих целевую функцию. Расчетные формулы для определения основных показателей приводятся ниже.

1 Плотность убытков

$$\rho = \frac{\text{C}_\text{д}}{\text{C}_s + \text{C}_\text{д}}. \qquad (5.16а)$$

2 Оптимальные значения $T_i^*,\ i = 1,\ 2,\ 3,\ 4$

$$\begin{aligned} T_2^* &= \sqrt{\frac{2\text{C}_p\text{C}_\text{д}(1 - \omega/k)}{\omega(\text{C}_\text{д} + \text{C}_s)\text{C}_s}} = \sqrt{\frac{2\text{C}_p\rho(1 - \omega/k)}{\omega\text{C}_s}}; \\ T_3^* &= T_2^* \frac{\text{C}_s}{\text{C}_\text{д}}; \\ T_1^* &= T_2^* \frac{\omega}{k - \omega}; \\ T_4^* &= T_3^* \frac{\omega}{k - \omega}. \end{aligned} \qquad (5.16б)$$

3 Оптимальное значение времени цикла

$$T^* = T_1^* + T_2^* + T_3^* + T_4^*. \qquad (5.16в)$$

4 Общий объем производства (равен объему спроса)

$$V = \omega T^*. \qquad (5.16г)$$

5 Максимальный уровень запасов

$$S_{\max} = T_2^* \omega = \sqrt{\frac{2\text{C}_p\rho(1 - \omega/k)}{\text{C}_s}}\,\omega. \qquad (5.16д)$$

6 Максимальный уровень дефицита

$$S_{\text{деф}} = T_3^* \omega = T_2^* \frac{\text{C}_s}{\text{C}_\text{д}}\,\omega. \qquad (5.16е)$$

7 Оптимальное значение целевой функции

$$q^* = \sqrt{\frac{2\omega C_s C_p C_д (1 - \omega/k)}{C_д + C_s}} = \sqrt{2\rho\omega C_p C_s (1 - \omega/k)}. \qquad (5.16ж)$$

8 Оптимальное число циклов

$$n^* = \theta / T^* = N / V. \qquad (5.16з)$$

Примером рассмотренной задачи управления запасами может быть случай, когда производство имеет слишком большой темп. В этом случае приближенно можно считать, что темп заполнения склада бесконечно большой ($k = \infty$), и на рис. 5.12 участки *AB* и *ГД* вертикальны. Расчетные формулы (5.16) при $k \to \infty$ совпадут с формулами (5.12), а рис. 5.12 – с рис. 5.13.

Таким образом, при бесконечно быстром производстве ситуация со складом на выходе тождественна ситуации склада на входе производства при бесконечно быстром заполнении склада сырьем и постоянной скорости потребления продукции производством.

В том случае, если дефицит не допускается, $C_д = \infty$, $\rho = 1$ и $k = \infty$, расчетные формулы (5.16) превращаются в расчетные формулы раздела 5.3.1.

5.3.4 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА. СКЛАД НА ВЫХОДЕ. ПРОИЗВОДСТВО БЕСКОНЕЧНОЙ ИНТЕНСИВНОСТИ. ОТСУТСТВИЕ ДЕФИЦИТА. СТОИМОСТЬ ХРАНЕНИЯ ПРОПОРЦИОНАЛЬНА СЕБЕСТОИМОСТИ ИЗДЕЛИЙ (ДЕТАЛЕЙ)

Так как дефицита нет, а скорость пополнения склада равна бесконечности, то при работе производства склад заполняется мгновенно. Изменение запаса на нем иллюстрируется рис. 5.9, где вертикальные участки характеризуют заполнение склада при работе производства. Уменьшение запасов на складе происходит в результате отпуска готового товара (изделий потребителю).

Целевая функция в данном случае

$$Q = Q_1 + Q_2,$$

где Q_1 – себестоимость изделий, поступивших на склад в результате работы производства за время θ; Q_2 – затраты на хранение изделий, пропорциональные времени и суммарной себестоимости деталей, все еще остающихся на складе.

Себестоимость Q_1 изготовленных деталей определяется по формуле

$$Q_1 = (\Delta S C_a + C_e) n, \qquad (5.17)$$

где C_e – постоянные затраты на серию; C_a – себестоимость единицы продукции; n – число серий, определяемых, как и раньше:

$$n = \theta/T = N/\Delta S.$$

Стоимость хранения продукции на складе пропорциональна количеству этой продукции $S(t)$, меняющейся во времени по закону, изображенному на рис. 5.9.

$$S(t) = \Delta S - \frac{\Delta S}{T} t.$$

Следовательно, можно определить затраты на хранение

$$Q_2 = \alpha \frac{\Delta S \mathrm{C}_a + \mathrm{C}_e}{\Delta S} \int_0^T \left[\Delta S - \frac{\Delta S}{T} t \right] dt,$$

где α – коэффициент пропорциональности.

Интегрирование выражения для Q_2 дает

$$Q_2 = \alpha \left[\mathrm{C}_a + \frac{\mathrm{C}_e}{\Delta S} \right] \left[\Delta S t \Big|_0^T - \frac{\Delta S}{2T} t^2 \Big|_0^T - \frac{\Delta S}{2T} t^2 \Big|_0^T \right] n =$$

$$= \alpha (\mathrm{C}_a + \frac{\mathrm{C}_e}{\Delta S}) \Delta S \frac{T}{2} n = \alpha \left[\frac{\mathrm{C}_a T}{2} \Delta S + \frac{\mathrm{C}_e}{2} T \right] n.$$

Общие затраты составляют

$$Q = \left[\Delta S \mathrm{C}_a + \mathrm{C}_e + \frac{\alpha \mathrm{C}_a T}{2} \Delta S + \alpha \frac{\mathrm{C}_e}{2} T \right] n =$$

$$= \left[\Delta S \mathrm{C}_a + \mathrm{C}_e + \frac{\alpha \mathrm{C}_a \theta}{2N} \Delta S^2 + \alpha \frac{\mathrm{C}_e}{2} \frac{\theta}{N} \Delta S \right] \frac{N}{\Delta S}$$

или

$$Q = \left[N \mathrm{C}_a + \alpha \frac{\mathrm{C}_e}{2} \theta + \frac{\mathrm{C}_e N}{\Delta S} + \frac{\alpha \mathrm{C}_a}{2} \Delta S \right]. \qquad (5.18)$$

Расчетные формулы находятся из соотношения, полученного в результате записи необходимого условия оптимальности целевой функции Q по ΔS, т.е. $\partial Q / \partial \Delta S = 0$. Минимум функции Q достигается при равенстве

$$\Delta S \frac{\alpha \theta}{2} \mathrm{C}_a = \frac{N \mathrm{C}_e}{\Delta S}.$$

Графическая иллюстрация представлена на рис. 5.13.

Основными расчетными формулами являются следующие:

1 Оптимальное значение экономической партии серии

$$\Delta S^* = \sqrt{2 \frac{\mathrm{C}_e}{\mathrm{C}_a} \frac{N}{\alpha \theta}}.$$

2 Оптимальное значение числа серий

$$n^* = N / \Delta S^*.$$

3 Оптимальный интервал цикла

$$T^T = \frac{\theta}{N} \Delta S^*.$$

4 Оптимальные затраты на хранение

$$Q^* = N \mathrm{C}_a + \frac{1}{2} \theta \alpha \mathrm{C}_e + \sqrt{2 N \theta \alpha \mathrm{C}_a \mathrm{C}_e}.$$

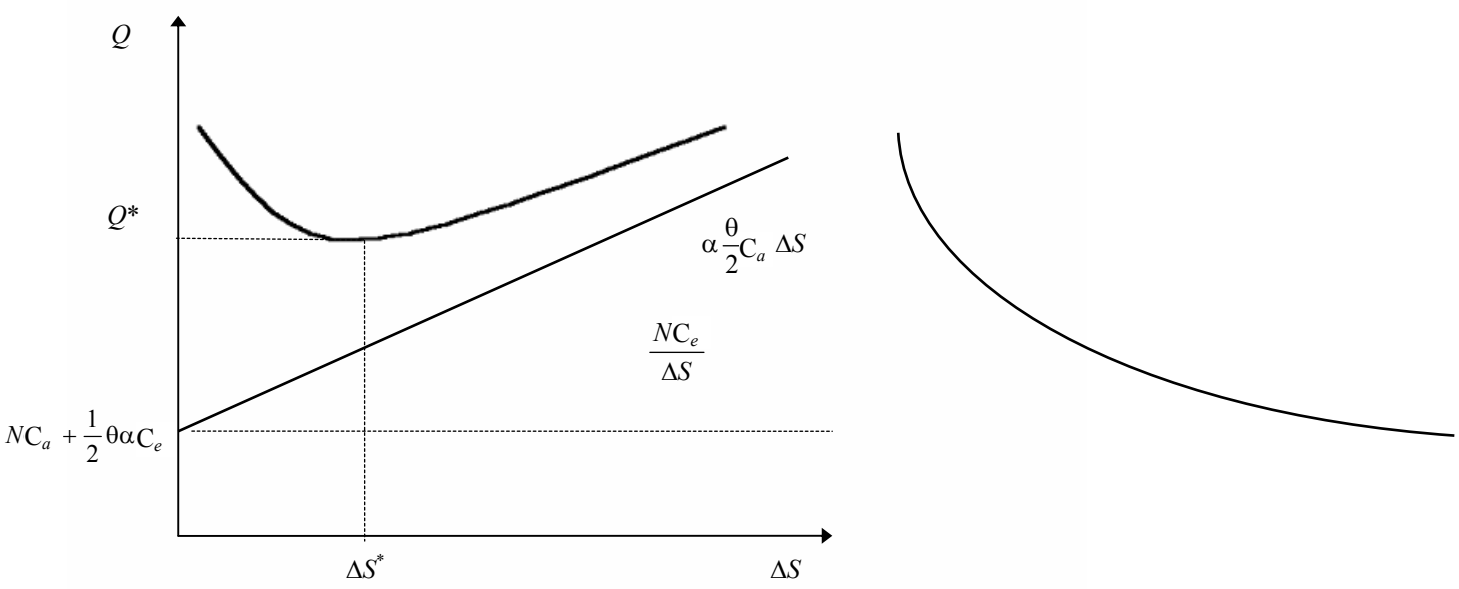


Рис. 5.13 Графическое определение оптимальной партии ΔS^*

5.3.5 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА. СКЛАД НА ВЫХОДЕ. ПРОИЗВОДСТВО БЕСКОНЕЧНОЙ ИНТЕНСИВНОСТИ. ОТСУТСТВИЕ ДЕФИЦИТА. ЗАВИСИМОСТЬ СЕБЕСТОИМОСТИ ОТ РАЗМЕРА ПАРТИИ

Ранее себестоимость S единицы продукции определялась соотношением

$$q_S=\begin{cases}0, & \text{если} \quad S=0;\\ C_e+C_aS, & \text{если} \quad S>0,\end{cases}$$

где C_e – постоянные затраты на серию; C_a – себестоимость единицы продукции.

При этом себестоимость единицы продукции была величиной постоянной. Однако, часто себестоимость изготовления (закупки) единицы продукции зависит от размеров изготавливаемой партии. Часто бывает ситуация, когда для маленьких партий себестоимость C'_a велика, а для больших – C''_a мала, т.е. $C'_a > C''_a$. При этом

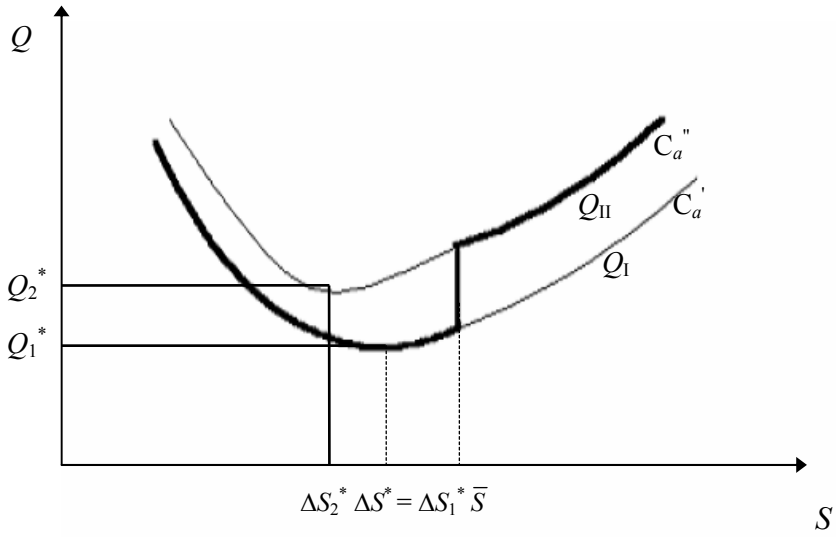
$$q_S=\begin{cases}0, & \text{если } S=0;\\ \vdash C'_aS, & \text{если } 0\le S<\overline{S}\\ C_e+C''_aS, & \text{если } \overline{S}\le S,\end{cases}$$

и целевая функция с учетом (5.18) записывается как

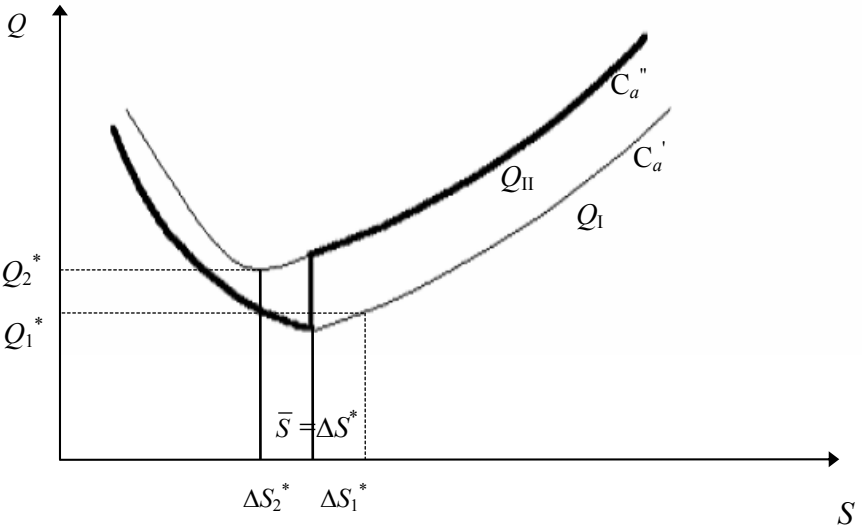
$$Q(\Delta S)=\begin{cases}0, & \text{если } \Delta S=0;\\ NC'_a+\alpha\frac{C_e}{2}\theta+\frac{C_eN}{\Delta S}+\frac{\alpha C'_a}{2}\Delta S, & \text{если } 0\le\Delta S<\overline{S};\\ NC''_a+\alpha\frac{C_e}{2}\theta+\frac{C_eN}{\Delta S}+\frac{\alpha C''_a}{2}\Delta S, & \text{если } \Delta S\ge\overline{S}.\end{cases}$$

Если обозначить кривую Q (ΔS) при $0\le\Delta S\le\overline{S}$ через $Q_{\text{I}}(\Delta S)$ (при этом $C_a=C'_a$), то минимальное значение Q_{I}^* для $Q_{\text{I}}(\Delta S)$ достигается в точке ΔS_{I}^* (рис. 5.14). При $\Delta S\ge\overline{S}$ кривая Q (ΔS) рассчитывается по (5.19) при $C_a=C''_a$ и обозначается как $Q_{\text{II}}(\Delta S)$. Минимальное значение $Q_{\text{II}}(\Delta S)$ будет Q_2^* и оно достигается в точке ΔS_2^* (рис. 5.14).

В соответствии с рис. 5.14 переход с кривой Q_{I} на кривую Q_{II} происходит при $S\ge\overline{S}$ (рис. 5.14).

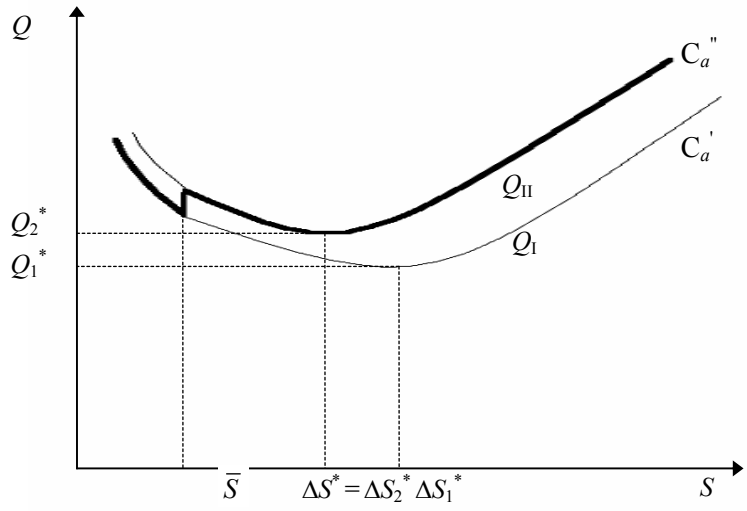


а)



б)

Рис. 5.14 Графическое решение задачи определения оптимальной партии:
а – $\Delta S^* = \Delta S_1^*$; *б* – $\Delta S^* = S$; *в* – $\Delta S^* = \Delta S_2^*$



в)

Рис. 5.14 Окончание

Оптимальное решение в соответствии с графическим решением задачи (рис. 5.14) определяется следующим образом:

а) если $\Delta S_1^* \le \bar{S}$ (рис. 5.14, *а*), то

$$\Delta S^* = \Delta S_1^* = \sqrt{2\frac{C_e}{C_a}\frac{N}{2\theta}};$$

$$Q^* = Q_1^* = NC'_a + \frac{1}{2}\theta\alpha C_e + \sqrt{2N\theta\alpha C'_a C_e};$$

$\Delta S_2^* \le \bar{S} < \Delta S_1^*$ (рис. 5.14, *б*), то $\Delta S^* = \bar{S}$;

б) если $\Delta S_2^* \le \bar{S} \le \Delta S_1^*$ (рис. 5.14, *б*), то $\Delta S^* = \bar{S}$;

$$Q^* = Q_1(\bar{S}) = NC'_a + \alpha\frac{C_e}{2}\theta + \frac{C_e N}{\bar{S}} + \frac{\alpha C''_a}{2}\bar{S};$$

в) если $\bar{S} > \Delta S_2^*$ (рис. 5.14, *в*), то

$$\Delta S^* = \arg\min(Q_1(\bar{S}), Q_2(\Delta S_2^*)),$$

$$Q^* = \min(Q_1(\bar{S}), Q_2(\Delta S_2^*)).$$

Аналогичным образом исследуется задача с конечными разрывами целевой функции при различных условиях в постановке задачи.

5.4 Детерминированная задача управления запасами при различных видах продукции

5.4.1 ПЕРИОДИЧЕСКИЙ ТИП АЛГОРИТМА, ЗАТРАТЫ НА ХРАНЕНИЕ ПРОПОРЦИОНАЛЬНЫ РАЗМЕРУ ПАРТИИ (СЕРИИ). СКЛАД НА ВХОДЕ

Работа склада рассматривается на достаточно большом интервале времени θ. При этом считают, что а) имеется n видов продукции; б) максимальный уровень запасов на складе фиксирован; в) ресурсы поступают на склад одновременно партиями (сериями). Также считают, что а) стоимость доставки i-й партии C_p^i; б) стоимость хранения единицы продукции в единицу времени C_s^i; в) общее потребление товара за время θ равно N_i.

Задача управления состоит в том, чтобы найти уровень пополнения ΔS_i и периоды пополнения T_i, при которых затраты на хранение N_i ресурсов, $i = 1, 2, \ldots, n$, на периоде θ были бы минимальны и при этом

$$\sum_{i=1}^{n} \Delta S_i \le I, \tag{5.19}$$

где $\Delta S_i = S_i^{\max} - S_i^{\min}$ – уровень пополнения i-го ресурса; $S_i^{\max}$ – максимальный запас i-го ресурса; $S_i^{\min}$ – минимальный запас i-го ресурса; I – объем склада, предназначенного для хранения запасов и ресурсов.

Пусть расходы ресурсов равномерны, тогда в единицу времени расходуется i-го ресурса $h_i = N_i/\theta$.

Затраты на хранение за период T i-го сырья составят

$$q^i = q_1^i + q_2^i,$$

где q_1^i – затраты на доставку ΔS i-го сырья; q_2^i – затраты на собственно хранение.

$$q^i = C_p^i + \frac{\Delta S_i}{2} T C_s^i. \tag{5.20}$$

Очевидно

$$h_i = \theta/T, \quad n_i = N_i/\Delta S_i,$$

отсюда

$$T = \frac{\theta}{N_i}\Delta S_i. \tag{5.21}$$

Используя (5.20), можно записать затраты Q_i за время θ

$$Q^i = \frac{q^i}{T}\theta = (\frac{C_p^i}{T} + \frac{\Delta S_i}{2}C_s^i)\theta$$

или с учетом (5.21)

$$Q^i = \frac{C_p^i N_i}{\Delta S_i} + \frac{\theta C_s^i}{2}\Delta S_i, \tag{5.22}$$

что согласуется с (5.5).

Суммарные затраты Q для всех видов продукции определяются как сумма

$$Q = \sum_{i=1}^{n} Q^i(\Delta S_i) = \sum_{i=1}^{n}\left[\frac{C_p^i N_i}{\Delta S_i} + \frac{\theta C_s^i}{2N_i}\Delta S_i\right]. \tag{5.22а}$$

Задача оптимизации управления запасами состоит в том, чтобы минимизировать целевую функцию (5.22) при условии соблюдения неравенства

$$\sum_{i=1}^{n}\Delta S_i \le I. \tag{5.23}$$

Легко показать, что задача оптимальна, если последнее неравенство превращается в равенство, т.е.

$$\sum_{i=1}^{n}\Delta S_i = I. \tag{5.24}$$

Безусловно, оптимальное значение $\Delta\overline{S}_i$, которое минимизирует (5.22), если бы условия (5.23) не было, т.е. склад был бы бесконечен, определяется из необходимого условия экстремума функции многих переменных $\partial Q^i / \partial \Delta S_i = 0$ или конкретно

$$\frac{\partial Q_i}{\partial \Delta S_i} = -\frac{C_p^i N_i}{\Delta S_i^2} + \frac{\theta C_s^i}{2} = 0, \tag{5.25}$$

откуда

$$\Delta\overline{S}_i^* = \sqrt{2\frac{C_p^i}{C_s^i}\frac{N_i}{\theta}}. \tag{5.26}$$

Этот результат уже был получен при рассмотрении склада с одним продуктом.

Если бы теперь оказалось, что

$$\sum_{i=1}^{n}\Delta\overline{S}_i^* < I,$$

то значения (5.26) были бы оптимальны, так как ограничения по величине склада I не играют роли (склад слишком большой и не препятствует завозу оптимальных объемов запасов).

Задача появится, если размеры склада этому препятствуют, т.е.

$$\sum_{i=1}^{n}\Delta\overline{S}_i^* > I.$$

В этом случае $\Delta\overline{S}_i^*$, найденные из (5.26), недопустимы и их нужно как-то уменьшить.

Задачу (5.22), (5.24) решают методом неопределенных множителей Лагранжа. Для этого составляется новая вспомогательная функция

$$L(\Delta S_i, \lambda) = \sum_{i=1}^{n} Q^i(\Delta S_i) + \lambda(\sum_{i=1}^{n} \Delta S_i - I), \qquad (5.27)$$

где λ – неопределенный множитель Лагранжа.

Решение находится дифференцированием $L(\Delta S_i, \lambda)$ по ΔS_i и приравниванием нулю производных $\partial L(\Delta S_i, \lambda) / \partial \Delta S_i = 0.$

Из (5.27) имеют

$$\frac{\partial Q^i(\Delta S_i)}{\partial(\Delta S_i)} + \lambda = 0 \qquad (5.28)$$

или $\partial Q^i / \partial(\Delta S_i) = -\lambda.$

Отсюда видно, что производные $\partial Q^i / \partial \Delta S_i$ в оптимальном режиме не равны нулю, как в (5.25), а равны некоторому числу (–λ). Причем они одинаковы для всех $i = 1, 2, 3, \ldots, n$, что свидетельствует о том, что оптимальные запасы ΔS_i^* выбираются таким образом, чтобы увеличения затрат на единицу уменьшения продукции $\partial Q^i / \partial \Delta S_i$ были бы одинаковы для всех $i = 1, 2, 3, \ldots, n$:

$$\frac{\partial Q^1(\Delta S_1)}{\partial(\Delta S_1)} = \frac{\partial Q^2(\Delta S_2)}{\partial(\Delta S_2)} = \cdots = \frac{\partial Q^n(\Delta S_n)}{\partial(\Delta S_n)} = -\lambda.$$

Если считать, что

$$\frac{\partial Q^i(\Delta S_i)}{\partial(\Delta S_i)} \approx \frac{\Delta Q^i}{\Delta(\Delta S_i)}, \quad \text{то}$$

$$\frac{\Delta Q^1}{\Delta(\Delta S_1)} = \frac{\Delta Q^2}{\Delta(\Delta S_2)} = \cdots = \frac{\Delta Q^n}{\Delta(\Delta S_n)} = -\lambda,$$

где $\Delta(\Delta S_i)$ – уменьшение максимального пополнения запасов ΔS_i; ΔQ^i – увеличение затрат из-за этого.

Соотношения (5.22), (5.24), (5.28) составляют систему уравнений для определения λ и оптимальных значений ΔS_i:

$$-\frac{C_p^i N_i}{\Delta S_i^2} + \frac{\theta C_i}{2} + \lambda = 0;$$

$$\sum_{i=1}^{n} \Delta S_i = I.$$

Решение этой системы дает

$$\Delta S_i^* = \sqrt{\frac{2C_p^i N_i}{C_{s_i}\theta + 2\lambda}}; \qquad (5.29)$$

$$\sum_{i=1}^{n} \Delta S_i^* = I\ . \qquad (5.30)$$

Для вычисления конкретных значений ΔS_i^* по (5.29) необходимо задать λ, после проведения этих вычислений проверяется правильность задания λ по (5.30), в случае необходимости оно корректируется (если условие (5.30) не выполняется).

5.5 Вероятностная задача управления запасами

Во многих случаях предсказать спрос нельзя, часто производство останавливается, если запас сырья мал, не всегда можно быстро восстановить производство, если уровень запасов был снижен. Поэтому

используется алгоритм с прогнозом либо момента пополнения запаса, либо количества пополнения запаса, т.е. алгоритм прогноза уровня пополнения склада.

Другими словами, прогнозируется, когда нужно покупать (производить товар), либо сколько и какого товара нужно купить (произвести).

Эти два алгоритма прогноза основаны на двух детерминированных основополагающих алгоритмах.

СПИСОК ЛИТЕРАТУРЫ

1 Танаев В.С. Теория расписаний. М.: Знание. 1988. 32 с.

2 Танаев В.С. Теория расписаний. Одностадийные системы. М.: Наука, 1984.

3 Майник Э. Алгоритмы оптимизации на сетях и графах. М.: Мир, 1981. 323 с.

4 Кофман А. Методы и модели исследования операций. М.: Мир, 1977. 432 с.

5 Шкурба В.В. Задача трех станков. М.: Наука, 1976. 95 с.

6 Рыжиков Ю.И. Управление запасами. М.: Наука, 1969. 343 с.

7 Вентцель Е.С. Исследование операций. М.: Наука, 1980. 230 с.

8 Кофман А., Крюон Р. Массовое обслуживание. Теория и приложение. М.: Мир, 1965. 302 с.

9 Кузин Л.Т. Основы кибернетики. Т. 1: Математические основы кибернетики. М.: Энергия, 1973. 504 с.

10 Ивченко Г.И., Каштанов В.А., Коваленко И.Н. Теория массового обслуживания. М.: Высш. шк., 1982. 256 с.