

ЭКОНОМИКО-
МАТЕМАТИЧЕСКАЯ
БИБЛИОТЕКА

В. С. ТАНАЕВ
Ю. Н. СОТСКОВ
В. А. СТРУСЕВИЧ

ТЕОРИЯ
РАСПИСАНИЙ
МНОГОСТАДИЙНЫЕ
СИСТЕМЫ

В.С. ТАНАЕВ, Ю.Н. СОТСКОВ, В.А. СТРУСЕВИЧ

ТЕОРИЯ РАСПИСАНИЙ

МНОГОСТАДИЙНЫЕ СИСТЕМЫ



МОСКВА "НАУКА"
ГЛАВНАЯ РЕДАКЦИЯ
ФИЗИКО-МАТЕМАТИЧЕСКОЙ ЛИТЕРАТУРЫ
1989

ББК 22.18
Т18
УДК 519.8

Танаев В.С., Сотсков Ю.Н., Струсеви́ч В.А. Теория расписаний. Многостадийные системы. — М.: Наука. Гл. ред. физ.-мат. лит., 1989. — 328 с. — (Экон.-мат. б-ка). — ISBN 5-02-013984-X.

Рассматриваются вопросы построения оптимальных расписаний для детерминированных обслуживающих систем с одинаковыми, различными и нефиксированными маршрутами прохождения приборов. Значительное внимание уделяется вопросам полиномиальной сводимости задач теории расписаний и оценкам сложности алгоритмов их решения. Развивается теоретико-графовый подход к анализу детерминированных обслуживающих систем.

Для специалистов в области прикладной математики и информатики, студентов и преподавателей математических специальностей вузов, а также специалистов в области экономики, занимающихся вопросами календарного планирования и применяющих в своей работе математические методы.

Табл. 30. Ил. 67. Библиогр. 839 назв.

Рецензент доктор физико-математических наук *В.К. Леонтьев*

Научное издание

Танаев Вячеслав Сергеевич
Сотсков Юрий Назарович
Струсеви́ч Виталий Александрович

ТЕОРИЯ РАСПИСАНИЙ
Многостадийные системы
Серия "Экономико-математическая библиотека"

Заведующий редакцией *Е.Ю. Ходан*
Редактор *А.Д. Вайнштейн*
Художественный редактор *Т.Н. Кольченко*
Технические редакторы: *С.В. Геворкян, С.Н. Баронина*
Корректоры: *Т.С. Родионова, Т.В. Обод, Т.А. Печко*

Набор осуществлен в издательстве
на наборно-печатающих автоматах
ИБ № 32302

Сдано в набор 10.05.88. Подписано к печати 31.08.88. Т—15862
Формат 60×90/16. Бумага писчая
Гарнитура Пресс-Роман. Печать офсетная
Усл.печ.л. 20,5. Усл.кр.отт. 20,5. Уч.-изд.л. 25,24
Тираж 3000 экз. Тип. зак. 234 Цена 5 р. 40 к.

Ордена Трудового Красного Знамени
издательство "Наука"
Главная редакция физико-математической литературы
117071 Москва В-71, Ленинский проспект, 15
Четвертая типография издательства "Наука"
630077 г. Новосибирск-77, ул. Станиславского, 25

1602110000—009
Т — 053 (02)—89

ISBN 5-02-013984-X

© Издательство "Наука".
Главная редакция
физико-математической литературы,
1989

Предисловие	4
Введение	5
Г л а в а 1. Одинаковые маршруты	26
§ 1. Общее время обслуживания. Два прибора	27
§ 2. Общее время обслуживания. Три и более приборов	35
§ 3. Общее время обслуживания требований без задержек	43
§ 4. Максимальное временное смещение	56
§ 5. Суммарное время обслуживания	60
§ 6. Упорядоченные матрицы длительностей	66
§ 7. Доминантные матрицы длительностей	79
§ 8. Приближенные алгоритмы	93
§ 9. Библиографическая справка	101
Г л а в а 2. Различные маршруты.	110
§ 1. Оптимальное обслуживание двух требований	111
§ 2. Максимальное временное смещение	121
§ 3. Общее время обслуживания. Одинаковые длительности	127
§ 4. Общее время обслуживания. Различные длительности	137
§ 5. Общее время обслуживания требований без задержек	146
§ 6. Библиографическая справка	152
Г л а в а 3. Нефиксированные маршруты.	157
§ 1. Общее время обслуживания. Два прибора	158
§ 2. Общее время обслуживания. Три и более приборов	179
§ 3. Общее время обслуживания. Прерывания	187
§ 4. Общее время обслуживания. Упорядоченность	200
§ 5. Директивные сроки	213
§ 6. Суммарное время обслуживания. Одинаковые длительности	224
§ 7. Суммарное время обслуживания. Различные длительности	233
§ 8. Библиографическая справка	242
Г л а в а 4. Задачи на смешанных графах.	248
§ 1. Сетевое представление обслуживающих систем	249
§ 2. Смешанные графы	258
§ 3. Последовательный анализ вариантов	266
§ 4. Оптимизация обслуживающей системы	277
§ 5. Устойчивость оптимальных расписаний	282
§ 6. Библиографическая справка	289
Список литературы	296

Предлагаемая вниманию читателя книга является естественным продолжением и дополнением опубликованных в издательстве "Наука" книг В.С. Танаева, В.В. Шкурбы "Введение в теорию расписаний" (1975 г.) и В.С. Танаева, В.С. Гордона, Я.М. Шафранского "Теория расписаний. Одностадийные системы" (1984 г.). В ней приводится систематизированное изложение современного состояния исследований многостадийных детерминированных обслуживающих систем.

Заметное в последнее время возрастание интереса к вопросам построения оптимальных расписаний для различных обслуживающих систем обусловлено существенным повышением уровня автоматизации всех видов человеческой деятельности, в том числе и управления этой деятельностью. Качество функционирования современного производства во многом определяется решениями, принимаемыми на этапах календарного планирования и оперативного управления. Наряду с улучшением качества плановых решений все более жесткими становятся требования к сокращению сроков их выработки, повышению оперативности и гибкости управления.

В книге рассматриваются детерминированные обслуживающие системы как с фиксированными (одинаковыми и различными), так и с нефиксированными маршрутами прохождения приборов. Все главы книги, а подчас и параграфы, относительно независимы. Используемые обозначения и терминология соответствуют упомянутым выше книгам. Каждая глава сопровождается библиографической справкой и иллюстрируется многочисленными примерами.

При подготовке книги к изданию были учтены замечания и пожелания участников семинара по теории управления в дискретных системах Института технической кибернетики АН БССР. Соответствующие курсы лекций читались авторами в Белорусском государственном университете им. В.И. Ленина. Ряд замечаний по рукописи сделали С.А. Бородич, В.С. Гордон, М.Я. Ковалев, А.В. Тузиков и Я.М. Шафранский. Большую помощь в оформлении рукописи оказали Р.А. Бенардская, С.Н. Воронкова и С.И. Крючкова.

Авторы признательны П.С. Краснощекову, В.С. Михалевичу, Н.Н. Моисеву, Г.С. Пospelову и Д.А. Супруненко за постоянное внимание к работе.

В.С. Танаев, Ю.Н. Сотсков, В.А. Струсович

Многие практические ситуации приводят к необходимости изучения *многостадийных обслуживающих систем*, т.е. таких систем, в которых процесс обслуживания каждого требования состоит из нескольких последовательных *стадий*, на каждой из которых это требование обслуживается тем или иным прибором, либо, в более общем случае, той или иной совокупностью приборов. Так, изготовление детали обычно включает несколько последовательных *операций*, каждая из которых выполняется на некотором из имеющихся в цехе станков. Преподаватель последовательно проводит занятия с несколькими вполне определенными группами студентов. Процесс подготовки книги к изданию включает такие стадии, как написание рукописи, рецензирование, редактирование, набор и т.д.

В любом случае имеется конечное множество $N = \{1, 2, \dots, n\}$ *требований* (деталей, преподавателей, книг и т.п.) и конечное множество $\mathcal{M} = \{1, 2, \dots, M\}$ *приборов* (станков, групп студентов, авторов, редакторов, наборщиков и т.п.). Процесс обслуживания требования $i \in N$ включает r_i стадий. При этом каждому требованию $i \in N$ и каждой стадии q , $1 \leq q \leq r_i$, его обслуживания сопоставляется некоторое множество приборов $\mathcal{M}_q^i \subseteq \mathcal{M}$. В зависимости от типа обслуживающей системы требование i на стадии q либо может быть обслужено любым из приборов $L \in \mathcal{M}_q^i$ (но не более чем одним одновременно), либо должно одновременно обслуживаться всеми приборами из множества $\mathcal{M}_q^i$. Обычно предполагается, что каждый прибор одновременно может обслуживать не более одного требования.

Если $r_i = r' \geq 2$, $\mathcal{M}_q^i = \mathcal{M}_q$, $i = 1, \dots, n$, $\mathcal{M}_{q_u}^i \cap \mathcal{M}_{q_v}^i = \emptyset$, $1 \leq q_u \neq q_v \leq r'$, то обслуживающая система называется *системой поточного типа*. В этой системе множество приборов разбито на r' попарно непересекающихся групп, обычно пронумерованных числами $1, 2, \dots, r'$. Каждое требование $i \in N$ сначала обслуживается приборами 1-й группы, затем 2-й и т.д. пока оно не будет обслужено приборами r' -й группы.

Наиболее изученной системой поточного типа является система, в которой $r' = M$ и, следовательно, $|\mathcal{M}_q| = 1$, $q = 1, \dots, M$. В этой системе каждое требование $i \in N$ сначала обслуживается прибором 1, затем прибором 2 и т.д., пока оно не будет обслужено прибором M . Такая система получила название системы с последовательными приборами и одинаковым поряд-

ком (*маршрутом*) их прохождения всеми требованиями. В зарубежной литературе она известна под названием *flow shop*.

В системе поточного типа с последовательно-параллельным характером обслуживания предполагается, что хотя бы при одном $q, 1 \leq q \leq M$, величина $|M_q| \geq 2$ и все приборы в каждом из множеств M_q одинаковы, либо отличаются друг от друга только производительностью. В этой системе каждое требование $i \in N$ на стадии q может обслуживаться одним прибором множества M_q , но не более чем одним одновременно.

Среди систем с *различными* порядками (*маршрутами*) прохождения приборов требованиями наиболее изученными являются системы с последовательными приборами. В этих системах для каждого требования $i \in N$ задается своя, специфическая для этого требования последовательность $L^i = (L^i_1, L^i_2, \dots, L^i_{r_i})$ его обслуживания приборами. Требование i сначала обслуживается прибором L^i_1 , затем прибором L^i_2 и т.д., пока оно не будет обслужено прибором $L^i_{r_i}$. Последовательности обслуживания могут быть различными для разных требований и могут содержать повторения приборов. В зарубежной литературе такие системы получили название *job shop*.

Определенный интерес представляют системы с последовательными приборами и *нефиксированным* порядком их прохождения для всех или некоторых требований. В наиболее изученной системе такого рода (известной под названием *open shop*) каждое требование $i \in N$ должно быть обслужено всеми M приборами при условии, что каждый прибор одновременно не может обслуживать более одного требования, а каждое требование одновременно не может обслуживаться более чем одним прибором.

В любом случае, если требование i на стадии q должно или может быть обслужено прибором $L = L^i_q$, то предполагается заданной *длительность* $t_{iL} \geq 0$ его обслуживания этим прибором. Запись $t_{iL} = 0$ означает, что требование i на стадии q должно быть обслужено прибором $L = L^i_q$, но длительностью этого обслуживания можно пренебречь. Зачастую запись $t_{iL} = 0$ употребляют и в том случае, когда необходимо отметить, что по условию задачи требование i на стадии q прибором L не обслуживается. Наряду с величинами t_{iL} могут быть заданы также *момент* $d_i \geq 0$ *поступления требования i в систему*, *директивный срок* $D_i > 0$, к которому необходимо или желательно завершить обслуживание требования i , и некоторые другие величины, обусловленные особенностями рассматриваемой задачи. В теории расписаний все эти величины предполагаются детерминированными.

Весьма распространенными являются ситуации, в которых накладываются некоторые ограничения на возможную последовательность обслуживания требований. Ситуации такого рода обычно описываются заданием на множестве N требований некоторого *отношения $\rightarrow$ строгого порядка*. В теории многостадийных обслуживающих систем запись $i \rightarrow j$ может интерпретироваться двояко: 1) обслуживание требования j может начаться только после завершения обслуживания требования i всеми обслуживающими его приборами либо 2) обслуживание требования j каждым обслуживающим его прибором L может начаться только после завершения обслуживания требования i этим прибором (при условии, что требование i обслуживается прибором L).

В процессе обслуживания требования прибором могут допускаться *прерывания*. Если допускаются прерывания обслуживания требования i прибором L , то это означает, что прибор L может обслуживать требование i "по частям". При этом обычно предполагается, что число "частей" конечно, суммарная длительность обслуживания всех "частей" равна t_{iL} и прерывания не сопряжены с дополнительными затратами. Процесс обслуживания требования i прибором L без прерываний удовлетворяет следующему условию. Если обслуживание требования i прибором L начинается в момент времени t_{iL}^0 , то оно протекает непрерывно и завершается в момент времени $\bar{t}_{iL} = t_{iL}^0 + t_{iL}$.

Определенный практический интерес представляют ситуации, в которых необходимо соблюдать одно из двух дополнительных условий: каждый прибор должен обслуживать требования непрерывно одно за другим либо каждое требование должно обслуживаться непрерывно одним прибором за другим. В первом случае не допускаются *простои* обслуживающих приборов, во втором — *задержки* в обслуживании требований.

Процесс функционирования обслуживающей системы может быть описан путем задания *расписания* (календарного плана, временного графика и т.п.), т.е. некоторой совокупности указаний относительно того, какие именно требования какими именно приборами обслуживаются в каждый момент времени. Обычно предполагается, что каждое требование не может одновременно обслуживаться двумя и более приборами и каждый прибор не может одновременно обслуживать более одного требования. При этих предположениях расписание можно рассматривать как совокупность $\{s_1(t), s_2(t), \dots, s_M(t)\}$ кусочно-постоянных непрерывных слева функций, каждая из которых задана на интервале $0 \leq t < \infty$ и принимает значения $0, 1, \dots, n$. Если $s_L(t') = i \neq 0$, то в момент времени t' прибор $L \in \mathcal{M}$ обслуживает требование $i \in N$. Если $s_L(t') = 0$, то в момент времени t' прибор L простаивает.

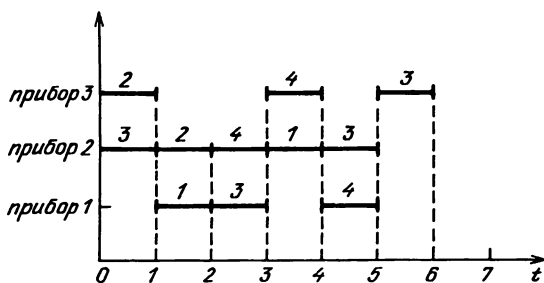


Рис. В1

Иногда вместо функций $s_L(t)$, $L \in \mathcal{M}$, описывающих функционирование каждого прибора, используют аналогичные функции $s_i(t)$, $i \in N$, описывающие процесс обслуживания каждого требования ($s_i(t') = 0$, если в момент времени t' требование i не обслуживается, и $s_i(t') = L$, если в момент времени t' требование i обслуживается прибором L).

При задании расписания должны соблюдаться все условия и ограничения, вытекающие из постановки рассматриваемой задачи, т.е. расписание должно

быть *допустимым*. Следует отметить, что построение допустимого расписания и даже выяснение того факта, существует ли оно вообще, является зачастую далеко не тривиальной задачей.

На рис. В.1 приведен график расписания $s_i(t)$, $i \in N$, обслуживания требований $N = \{1, 2, 3, 4\}$ приборами $\mathcal{M} = \{1, 2, 3\}$ при различных маршрутах обслуживания требований. Здесь все длительности обслуживания равны единице, $L^1 = (1, 2)$, $L^2 = (3, 2)$, $L^3 = (2, 1, 2, 3)$, $L^4 = (2, 3, 1)$; $d_1 = 1, d_2 = d_3 = 0, d_4 = 2$; $D_1 = D_4 = 5, D_2 = 2, D_3 = 6$.

Прибор 1 во временном интервале $(1, 2]$ обслуживает требование 1, в интервале $(2, 3]$ — требование 3, в интервале $(4, 5]$ — требование 4. Прибор 2 в интервале $(0, 5]$ обслуживает без простоев требования в порядке 3, 2, 4, 1 и снова 3. Прибор 3 в интервале $(0, 1]$ обслуживает требование 2, в интервале $(3, 4]$ — требование 4, в интервале $(5, 6]$ — требование 3. Нетрудно убедиться в том, что это расписание является допустимым.

На рис. В.2 приведены графики $s_L(t)$, $L \in \mathcal{M}$, описывающие функционирование каждого прибора. Над графиками — соответствующие значения L .

Если существует несколько допустимых расписаний, то естественным является стремление построить наилучшее из них. В связи с этим возникает весьма сложный и многоплановый вопрос оценки *качества расписания*. В теории расписаний наибольшее распространение получил следующий способ оценки. Каждое (допустимое) расписание s однозначно определяет вектор $\bar{t}(s) = (\bar{t}_1(s), \bar{t}_2(s), \dots, \bar{t}_n(s))$ моментов завершения обслуживания требований. Задается некоторая действительная неубывающая по каждой из переменных функция $F(x) = F(x_1, x_2, \dots, x_n)$, и качество расписания s оценивается значением этой функции при $x = \bar{t}(s)$. Из двух расписаний лучшим

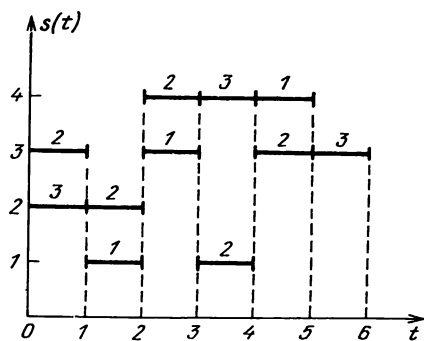


Рис. В.2

считается то, которому соответствует меньшее значение $F(x)$. Расписание, которому соответствует наименьшее значение $F(x)$ (среди всех допустимых расписаний), называется *оптимальным*.

В частности, при построении *оптимального по быстродействию расписания* $F(x) = \max_{1 \leq i \leq n} \{x_i\}$. В этом

случае $F(\bar{t}(s)) = \bar{t}_{\max}(s)$, где $\bar{t}_{\max}(s) = \max_{1 \leq i \leq n} \{\bar{t}_i(s)\}$.

При построении расписания с наименьшим *суммарным временем об-*

служивания $F(x) = \sum_{i=1}^n x_i$. В этом случае $F(\bar{t}(s)) = \sum_{i=1}^n \bar{t}_i(s)$. При построении

расписания с наименьшим *временным смещением* моментов завершения обслуживания требований i относительно директивных сроков D_i функция

$F(x) = \max_{1 \leq i \leq n} \{x_i - D_i\}$. В этом случае $F(\bar{t}(s)) = L_{\max}(s)$, где $L_{\max}(s) = \max_{1 \leq i \leq n} \{\bar{t}_i(s) - D_i\}$.

В большинстве рассматриваемых в этой книге ситуаций оптимальное расписание может быть найдено в результате перебора конечного множества возможных вариантов. Основное затруднение состоит в том, что число таких вариантов обычно оказывается исключительно большим и растет, по меньшей мере, экспоненциально с ростом размерности задачи.

Среди задач теории расписаний можно выделить полиномиально разрешимые и *NP-трудные*. Для каждой *полиномиально разрешимой задачи* известен по крайней мере один *эффективный алгоритм* ее решения, т.е. алгоритм, *трудоемкость* которого (число выполняемых элементарных операций, время решения) ограничена сверху некоторым полиномом от длины записи исходной информации задачи, закодированной в бинарном алфавите. Для *NP-трудных* задач такие алгоритмы неизвестны и, по всей видимости, не существуют. Во всяком случае, если бы существовал полиномиальный алгоритм решения какой-либо одной *NP-трудной* задачи, то существовали бы полиномиальные алгоритмы для очень многих задач, которые традиционно считаются труднорешаемыми.

С вопросами оценки сложности алгоритмов и теорией полиномиальной сводимости дискретных задач можно ознакомиться по монографиям [88, 215, 238, 256, 350, 352, 651, 754] или статьям [133, 158, 174, 190, 480, 532, 616, 621, 652, 657, 659, 736, 765, 793]. В данной книге используется общепринятая техника доказательства *NP-трудности* экстремальных задач. Пусть требуется отыскать в конечном множестве S допустимых расписаний расписание s^* , которому соответствует наименьшее значение $F(\bar{t}(s))$. Сформулируем соответствующую *задачу распознавания*: существует ли во множестве S такое расписание s^0 , что $F(\bar{t}(s^0)) \leq y$ для заданного числа y ? Очевидно, указанное расписание s^0 существует тогда и только тогда, когда $F(\bar{t}(s^*)) \leq y$. Таким образом, экстремальная задача не легче, чем соответствующая ей задача распознавания, и из *NP-трудности* задачи распознавания следует *NP-трудность* исходной экстремальной задачи. Для доказательства *NP-трудности* задачи распознавания достаточно осуществить полиномиальное сведение к ней некоторой (заведомо *NP-трудной*) задачи. Формулировки используемых в каждой главе *эталонных задач* приводятся во введениях к этим главам.

Среди *NP-трудных* задач можно выделить задачи, для которых существуют *псевдополиномиальные алгоритмы* решения, т.е. алгоритмы, сложность которых ограничена сверху некоторым полиномом от длины записи исходной информации задачи, закодированной в унарном алфавите (натуральное число k в этом алфавите представляется последовательностью k единиц). Остальные *NP-трудные* задачи относятся к классу *NP-трудных в сильном смысле*. Если эталонная задача является *NP-трудной в сильном смысле*, то нет необходимости строить полиномиальное ее сведение к рассматриваемой задаче распознавания. Достаточно построить *псевдополиномиальное сведение* и тем самым убедиться в *NP-трудности* (в сильном смысле) исходной задачи.

В этой книге описывается современное состояние исследований многостадийных обслуживающих систем.

В гл. 1 рассматриваются системы с последовательными приборами и одинаковым порядком их прохождения. В § 1 этой главы установлены некоторые свойства расписаний (без прерываний) и описаны полиномиальные

алгоритмы решения задачи построения оптимального по быстродействию расписания для двух приборов и некоторых ее обобщений. *NP*-трудность задачи минимизации общего времени обслуживания (как при запрещении, так и при разрешении прерываний) для трех и более приборов установлена в § 2. В § 3 рассматривается задача построения оптимального по быстродействию расписания без задержек в обслуживании каждого требования. В § 4 и 5 содержатся доказательства *NP*-трудности задач построения расписаний, которым соответствует минимальное временное смещение и суммарное время обслуживания соответственно. В § 6 и 7 приведены условия, выполнение которых позволяет строить оптимальное расписание за полиномиальное число действий. В § 8 основное внимание уделяется приемам построения алгоритмов с априорными оценками качества решения.

В гл. 2 рассматриваются системы с различными маршрутами обслуживания требований. В § 1 этой главы описываются эффективные алгоритмы построения оптимальных расписаний (с прерываниями и без прерываний) обслуживания двух требований при произвольной неубывающей целевой функции. В § 2 рассматривается задача минимизации максимального временного смещения при обслуживании (без прерываний) двумя приборами требований с одинаковыми длительностями обслуживания. В § 3 устанавливается *NP*-трудность некоторых частных случаев задачи построения оптимальных по быстродействию расписаний обслуживания (без прерываний) требований с одинаковыми длительностями обслуживания. В § 4 рассматривается задача построения оптимального по быстродействию расписания при произвольных длительностях обслуживания. В § 5 устанавливается *NP*-трудность задачи построения оптимального по быстродействию расписания без задержек в процессе обслуживания каждого требования.

Гл. 3 посвящена задачам теории расписаний для систем с нефиксированными маршрутами обслуживания требований. В § 1 этой главы приведены полиномиальные алгоритмы решения задачи построения оптимального по быстродействию расписания для двух приборов и ее обобщения. *NP*-трудность задачи минимизации общего времени обслуживания (без прерываний) для трех и более приборов установлена в § 2. Там же описано несколько частных случаев, когда решение указанной задачи можно найти за полиномиальное число операций. В § 3 приведены полиномиальные алгоритмы построения оптимального по быстродействию расписания с прерываниями. В § 4 исследуется ситуация, когда на множестве требований задано отношение строгого порядка. Различные аспекты решения задачи построения расписания с минимаксным временным смещением рассмотрены в § 5. В § 6 приведены полиномиальные алгоритмы построения расписаний, которым соответствуют наименьшие значения суммарного и суммарного взвешенного времени обслуживания, если длительности обслуживания одинаковы. В § 7 установлена *NP*-трудность задачи минимизации суммарного времени обслуживания в случае произвольных длительностей.

В гл. 4 рассматриваются сетевые формы представления обслуживающих систем с использованием смешанных графов. В § 1 описываются алгоритмы построения допустимых относительно заданного графа расписаний. В § 2 исследуются некоторые полезные в дальнейшем свойства смешан-

ных графов. В § 3 рассматриваются основанные на идеях последовательного анализа вариантов алгоритмы построения оптимальных и близких к ним расписаний. В § 4 рассматривается задача оптимального выбора необходимого количества приборов, распределения заданного множества требований по выбранным приборам и составления расписания их обслуживания. Устойчивость оптимальных расписаний исследуется в § 5.

Каждая глава сопровождается библиографической справкой. Дополнительную информацию о результатах, полученных в области теории расписаний и некоторых смежных областях, заинтересованный читатель может найти в монографиях [1, 23, 77, 123, 147, 153, 157, 189, 211, 212, 215, 231, 241–243, 247, 251, 269, 280, 286–288, 350–352, 385, 402–404, 446, 504, 528, 651, 754] и обзорах [45, 49, 50, 61, 76, 80, 98, 233, 263, 337, 345, 365, 430, 432, 501, 502, 514, 535, 546, 548, 568–570, 605, 622, 635, 636, 643, 646, 654–656, 755, 784].

Для облегчения поиска информации о конкретно интересующих читателя задачах ниже приводятся таблицы, содержащие краткие сведения о большинстве рассматриваемых в книге задач.

В табл. В.1 приведены полиномиально разрешимые задачи, в табл. В.2 — NP -трудные задачи, а в табл. В.3 — $\overline{NP}$ -трудные задачи, для которых известны полиномиальные приближенные алгоритмы решения.

В первых шести графах каждой таблицы приводятся описания задач с использованием определенной символики. В последней графе даются ссылки на соответствующие разделы книги.

В первой графе указывается тип обслуживающей системы. Системы с нефиксированными маршрутами обслуживания требований обозначены буквой O , системы с различными фиксированными маршрутами — буквой J , системы с одинаковыми маршрутами — буквой F .

Во второй графе указывается число обслуживающих приборов.

В третьей графе приводятся значения двух параметров — "длительности операций" и "моменты поступления". Под операцией подразумевается процесс обслуживания требования отдельным прибором. При произвольных длительностях операций первый параметр принимает значение t_{iL} , при целочисленных — $[t_{iL}]$, при одинаковых — $t_{iL} = 1$, при двух возможных значениях t_1 и t_2 — $t_{iL} \in \{t_1, t_2\}$.

Параметр "моменты поступления" принимает значение d_i , если требования поступают в очередь на обслуживание неодновременно. Если требования поступают только в целочисленные моменты времени, то используется символ $[d_i]$. Если оба этих символа отсутствуют, то предполагается, что $d_i = 0$ для всех $i \in N$.

В четвертой графе приводятся значения параметров "прерывания", "порядок", "ресурсы" и некоторых других параметров, определяющих условия обслуживания требований.

Параметр "прерывания" может принимать значения Pr и $[Pr]$ в зависимости от того, разрешены ли прерывания в процессе обслуживания требования прибором в произвольные или только целочисленные моменты времени. Символ (Pr) означает, что процесс обслуживания каждого требования любым прибором может быть прерван, а потом возобновлен, но между моментами прерывания и возобновления требование не может

Таблица В.1

Тип системы	Число приборов	Длительности операций; моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Оценка сложности	Раздел книги
F	2	t_{iL}	π	$t_{i2}^2 > d > 0$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 1.3 (*)
F	2	t_{iL}	π	$\delta_{i1} > 0, \delta_{i2} > 0,$ $\tau_{i1} > 0, \tau_{i2} > 0,$ $\xi_i > 0,$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 1.3, 9.1
F	2	t_{iL}	π	$\delta_{i1} > 0, \delta_{i2} > 0,$ $\tau_{i1} > 0, \tau_{i2} > 0,$ $\xi_i > 0, \eta_i > 0$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 9.1
F	2	t_{iL}	$G_0 = SP, \pi$		$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 9.2
F	2	t_{iL}	π , без задержек		$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 3.5
F	2	t_{iL}	без простоев		$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 1.2, 9.3 (*)
F	2	$t_{iL} = 1$	$Rs(1)$		$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 9.4 (*)
F	2	$t_{iL} = 1$	$Rs(1)$, без задержек		$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 9.4
F	2	$t_{iL} = 1$	$Rs(1), R_k = 1,$ $\rho_{ik} \in \{0, 1\}$		$\bar{t}_{\max}$	n	гл. 1, п. 9.4
F	2	t_{iL}	π	$t_{i1} > t_{i2}, i \in N$	$\sum \bar{t}_i$	$n \log n$	гл. 1, п. 6.2, 3)
F	3	t_{iL}	π	$t_{i2} \leq \min\{t_{i1}, t_{i3}\},$ $i \in N$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 9.9
F	3	t_{iL}	π	$t_{i1} \leq t_{i2} \leq t_{i3}, i \in N$	$\bar{t}_{\max}$	n^7	гл. 1, п. 9.9
F	3	t_{iL}	π	$t_{i1} > t_{i2} > t_{i3}, i \in N$	$\bar{t}_{\max}$	n^7	гл. 1, п. 9.9
F	M	t_{iL}	π , без задержек	$\parallel t_{iL} \parallel$ — упорядочена по строкам,	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 6.3, 5)

F	M	t_{iL}	π , без за- держек	$t_{iM} \geq t_{iL}, i \in N,$ $L = 1, \dots, M - 1$ $\ t_{iL}\ $ — упорядо- чена по строкам, $t_{i1} \geq t_{iL}, i \in N,$ $L = 2, \dots, M$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 6.3, 5)
F	M	t_{iL}	π , без за- держек	$\ t_{iL}\ $ — упорядо- чена по строкам,	$\sum \bar{t}_i$	$n \log n$	гл. 1, п. 6.4
F	M	t_{iL}	π , без за- держек	$\ t_{iL}\ $ — упо- рядочена по строкам	$\bar{t}_{\max}$	$n^2 M$	гл. 1, п. 6.3, 3)
F	M	t_{iL}	π	$\ t_{iL}\ $ — упорядо- чена, $t_{i1} \geq t_{iL},$ $i \in N, L = 2, \dots, M$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 6.1, 2)
F	M	t_{iL}	π	$\ t_{iL}\ $ — упорядо- чена, $t_{iM} \geq t_{iL},$ $i \in N, L =$ $= 1, \dots, M - 1$	$\bar{t}_{\max}$	$n \log n$	гл. 1, п. 6.1, 2)
F	M	t_{iL}	π	$\ t_{iL}\ $ — упорядо- чена	$\sum \bar{t}_i$	$n \log n$	гл. 1, п. 6.2, 2)
F	M	t_{iL}	π	$\ t_{iL}\ $ — доми- нантная	$\bar{t}_{\max}$	nM	гл. 1, п. 7.1
F	M	t_{iL}	π	$\ t_{iL}\ $ — (7.6)	$\bar{t}_{\max}$	$nM + n \log n$	гл. 1, п. 7.2, 2)
F	M	t_{iL}	π	$\ t_{iL}\ $ — (7.8)	$\bar{t}_{\max}$	$nM + n \log n$	гл. 1, п. 7.2, 3)
F	M	t_{iL}	π	$\ t_{iL}\ $ — (7.11)	$\bar{t}_{\max}$	$nM + n^2$	гл. 1, п. 7.3
F	M	t_{iL}	π	$\ t_{iL}\ $ — доми- нантная	$\sum w_i \bar{t}_i$	$nM + n^2$	гл. 1, п. 7.5
F	M	t_{iL}	π , без за- держек	$\ t_{iL}\ $ — доми- нантная	$\bar{t}_{\max}$	nM	гл. 1, п. 7.7

Т а б л и ц а В.1 (продолжение)

Тип системы	Число приборов	Длительности операций; моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Оценка сложности	Раздел книги
F	M	t_{iL}	π , без задержек	$\ t_{iL}\ - (7.19)$	$\bar{t}_{\max}$	$nM + n \log n$	гл. 1, п. 7.8, 1)
F	M	t_{iL}	π , без задержек	$\ t_{iL}\ - (7.20)$	$\bar{t}_{\max}$	$nM + n \log n$	гл. 1, п. 7.8, 2)
F	M	t_{iL}	π , без задержек	$\ t_{iL}\ - \text{доминантная}$	$\sum w_i \bar{t}_i$	$nM + n^2$	гл. 1, п. 7.9
J	2	$t_{iL} = 1$			$L_{\max}$	$r(N) \log n$	гл. 2, § 2
J	2	$t_{iL} = 1$		$[D_i]$	$L_{\max}$	$r(N)$	гл. 2, § 2
J	2	$t_{iL} = 1, d_i$			$\bar{t}_{\max}$	$r(N) \log n$	гл. 2, п. 3.2
J	2	$t_{iL} = 1, [d_i]$			$\bar{t}_{\max}$	$r(N)$	гл. 2, п. 3.2
J	2	t_{iL}			$\bar{t}_{\max}$	$m \log m$	гл. 2, п. 4.2
J	2	t_{iL}	без простоев	$r_i \leq 2$	$\bar{t}_{\max}$	$m \log m$	гл. 2, п. 4.3
J	M	t_{iL}		$r_i \leq 2$	F	$r^2 \log r$	гл. 2, п. 1.8-1.12
J	M	t_{iL}	Pr	$n = 2$	F	r^3	гл. 2, п. 1.3-1.6, 1.12
J	M	t_{iL}		$n = 2, r_i \leq M$	F	$M \log M$	гл. 2, п. 1.12
J	M	t_{iL}	Pr	$n = 2, r_i \leq M$	F	M^2	гл. 2, п. 1.12
O	2	t_{iL}	$Rs(q)$		$\bar{t}_{\max}$	n	гл. 3, п. 1.2 (*)
O	2	t_{iL}	$[Pr]$		$\bar{t}_{\max}$	n^3	гл. 3, п. 8.4
O	2	$t_{iL} = 1$		$\varphi(t) - \text{неубывающая функция}$	$\bar{t}_{\max}$	$n \log n$	гл. 3, п. 8.4
		$t_{i1} = a_i + \varphi(t_i^0)$					
		$t_{i2} = b_i + \varphi(t_i^0)$					
O	2	t_{iL}	$Pr, G = C \cup \Gamma$		$\bar{t}_{\max}$	n	гл. 3, п. 4.3
O	2	t_{iL}	$G_0 = C$		$\bar{t}_{\max}$	$n \log n$	гл. 3, п. 4.6

O	2	t_{iL}	$Pr, G_0 = C$	$\bar{t}_{\max}$	n^2	гл. 3, п. 4.6
O	2	t_{iL}	$Pr, G_0 =$ $= C_1 \cup C_2 \cup \dots$ $\dots \cup C_p \cup \Gamma$	$\bar{t}_{\max}$	n^2	гл. 3, п. 4.7
O	2	t_{iL}	Pr	$L_{\max}$	$n \log n$	гл. 3, п. 5.4
O	2	t_{iL}	Pr	$L_{\max}$	n	гл. 3, п. 5.4
O	2	t_{iL}	Pr на одном приборе	$L_{\max}$	n^3	гл. 3, п. 8.5
O	M	t_{iL}	$G = C$	$\bar{t}_{\max}$	nM	гл. 3, п. 4.1 (*)
O	M	t_{iL}, d_i	$Pr, G = C \cup \Gamma$	$\bar{t}_{\max}$	$ЛП$	гл. 3, п. 4.2
O	M	t_{iL}		$\bar{t}_{\max}$	nM	гл. 3, п. 2.2
O	M	t_{iL}	доминирующий прибор, $n > M$	$\bar{t}_{\max}$	nM	гл. 3, п. 2.3
O	M	t_{iL}	два доминирую- щих прибора, $n > M$	$\bar{t}_{\max}$	$n^2 M^3$	гл. 3, п. 2.5
O	M	t_{iL}	$t^*(N) > (M^2 +$ $+ 2M - 1)t^*$	$\bar{t}_{\max}$	$n^2 M^3$	гл. 3, п. 8.2
O	M	t_{iL}	$t^*(N) >$ $> (16M^0 \log M^0 +$ $+ 5M^0)t^*, M^0 = 2^k,$ $2^{k-1} < M \leq 2^k$	$\bar{t}_{\max}$	$nM^3 + nM^2 \log n$	гл. 3, п. 8.2
O	M	t_{iL}	$t^*(N) >$ $> (8M^0 \log M^0 +$ $+ 5M^0)t^*, M^0 = 2^k,$ $2^{k-1} < M \leq 2^k$	$\bar{t}_{\max}$	$H \log^2(n + M)$	гл. 3, п. 3.2, 8.3

Тип системы	Число приборов	Длительности операций; моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Оценка сложности	Раздел книги
O	M	$[t_{iL}]$	$[Pr]$		$\bar{t}_{\max}$	$(n + M)^2 \times \log(n + M)$	гл. 3, п. 3.2, 8.3
O	M	$[t_{iL}]$	$[Pr]$		$\bar{t}_{\max}$	$(n + M)H \log t^*$	гл. 3, п. 3.2, 8.3
O	M	t_{iL}	Pr		$\bar{t}_{\max}$	H^2	гл. 3, п. 3.4, 8.3
O	M	t_{iL}	Pr		$\bar{t}_{\max}$	$H(\min\{H, M^2\} + M \log n)$	гл. 3, п. 3.4, 8.3
O	M	t_{iL}	Pr		$\bar{t}_{\max}$	$H + \min\{M^4, n^4, H^2\}$	гл. 3, п. 3.4, 8.3
O	M	t_{iL}	Pr	параллельные приборы на каждой стадии	$\bar{t}_{\max}$	ЛП	гл. 3, п. 3.6
O	M	t_{iL}, d_i	Pr		$\bar{t}_i \leq D_i$	ЛП	гл. 3, п. 5.1
O	M	$t_{iL}, d_i \in \{0, d\}$	Pr	$D_i = D, 0 < d < D$	$\bar{t}_i \leq D_i$	$n^3 + M^3$	гл. 3, п. 5.2, 2)
O	M	t_{iL}	Pr	$D_i \in \{D', D''\}, 0 < D' < D''$	$\bar{t}_i \leq D_i$	$n^3 + M^3$	гл. 3, п. 5.2, 3)
O	M	t_{iL}, d_i	Pr		$\bar{t}_{\max}$	ЛП	гл. 3, п. 5.3
O	M	$t_{iL} = 1$	Pr		$\bar{t}_i$	nM	гл. 3, п. 6.1 (*)
O	M	$t_{iL} = 1$	Pr		$\{\bar{t}_{\max}, \Sigma \bar{t}_i\}$	nM	гл. 3, п. 6.1 (*)
O	M	$t_{iL} = 1$	Pr		$\Sigma w_i \bar{t}_i$	$nM + n \log n$	гл. 3, п. 6.3
O, F	2	t_{iL}	Pr		$\bar{t}_{\max}$	$n \log n$	гл. 3, п. 8.1 (*)
O, J	2	t_{iL}	Pr		$\bar{t}_{\max}$	$n + m \log m$	гл. 3, п. 1.4
O, J	2	t_{iL}	Pr		$\bar{t}_{\max}$	$n + m \log m$	гл. 3, п. 3.8

Тип системы	Число прифоров	Длительности операций, моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Раздел книги
F	2	t_{iL}	$G = C \cup \Gamma, \pi$		$\bar{t}_{\max}$	гл. 1, п. 9.2
F	2	t_{iL}	G_0, π		$\bar{t}_{\max}$	гл. 1, п. 9.2
F	2	t_{iL}	$Rs(1), R_k = 1,$ $\rho_{jk} \in \{0, 1\}$		$\bar{t}_{\max}$	гл. 1, п. 9.4
F	2	$t_{iL} \geq 0$	без задержек		$\bar{t}_{\max}$	гл. 1, п. 3.3
F	2	t_{iL}	$\pi, 0 < b < n - 1$		$\bar{t}_{\max}$	гл. 1, п. 9.5.
F	2	t_{iL}	π , без задержек		$\sum \bar{t}_i$	гл. 1, п. 5.2
F	2	$t_{iL} = 1$	$G = C \cup \Gamma$		$\sum w_i \bar{t}_i$	гл. 1, п. 9.8
F	2	t_{iL}			$L_{\max}$	гл. 1, п. 4.1
F	2	t_{iL}	Pr		$L_{\max}$	гл. 1, п. 4.1
F	2	t_{iL}	π , без задержек		$L_{\max}$	гл. 1, п. 4.2
F	2	$t_{iL} \geq 0$	без задержек	$D_i = D, i \in N$	$L_{\max}$	гл. 1, п. 4.2
F	2	t_{iL}	π		$\sum \bar{t}_i$	гл. 1, п. 5.1
F	3	t_{iL}	Pr		$\sum \bar{t}_i$	гл. 1, п. 9.8
F	3	$t_{iL} = 1$	$Rs(1)$		$\bar{t}_{\max}$	гл. 1, п. 9.4
F	3	t_{iL}			$\bar{t}_{\max}$	гл. 1, п. 2.1
F	3	t_{iL}	без простоев		$\bar{t}_{\max}$	гл. 1, п. 2.1, 9.3
F	3	t_{iL}	Pr		$\bar{t}_{\max}$	гл. 1, п. 2.3
F	3	t_{iL}	Pr	$r_i = 2, i \in N$	$\bar{t}_{\max}$	гл. 1, п. 2.4 (*)
F	3	t_{iL}	Pr , без простоев		$\bar{t}_{\max}$	гл. 1, п. 2.3, 9.3
F	3	t_{iL}	π , без задержек		$\bar{t}_{\max}$	гл. 1, п. 3.7
F	M	$t_{iL} \in \{0, 1\}$			$\bar{t}_{\max}$	гл. 1, п. 9.8
F	M	$t_{iL} \in \{0, 1\}$	Pr		$\bar{t}_{\max}$	гл. 1, п. 9.8

Таблица В.2 (окончание)

Тип системы	Число приборов	Длительности операций, моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Раздел книги
F	M	$t_{iL} \in \{0, 1\}$	без задержек		$\bar{t}_{\max}$	гл. 1, п. 9.8
F	M	$t_{iL} \in \{0, 1\}$			$\Sigma \bar{t}_i$	гл. 1, п. 9.8
F	M	$t_{iL} \in \{0, 1\}$	P_T		$\Sigma \bar{t}_i$	гл. 1, п. 9.8
F	M	$t_{iL} \in \{0, 1\}$	без задержек		$\Sigma \bar{t}_i$	гл. 1, п. 9.8
J	2	$t_{iL} = 1$	$G = C_1 \cup C_2 \cup \dots$ $\dots \cup C_p \cup \Gamma$		$\bar{t}_{\max}$	гл. 2, п. 3.5
J	2	$t_{iL} \in \{1, 2\}$			$\bar{t}_{\max}$	гл. 2, п. 3.6
J	2	t_{iL}		$r_1 = 3, r_i = 1,$ $i = 2, \dots, n$	$\bar{t}_{\max}$	гл. 2, п. 4.6 (*)
J	2	t_{iL}		$r_1 = 2(n-1), r_i = 2,$ $i = 2, \dots, n$	$\bar{t}_{\max}$	гл. 2, п. 4.7
J	2	t_{iL}		$r_1 = r_2 = 3, r_i = 2,$ $i = 3, \dots, n$	$\bar{t}_{\max}$	гл. 2, п. 4.5 (*)
J	2	t_{iL}	P_T		$\bar{t}_{\max}$	гл. 2, п. 6.4
J	2	t_{iL}	P_T	$r_1 = \lceil n/3 \rceil, r_i = 2,$ $i = 2, \dots, n$	$\bar{t}_{\max}$	гл. 2, п. 5.1–5.4
J	2	t_{iL}	без задержек	$r_i = 2$	$\bar{t}_{\max}$	гл. 2, п. 5.5 (*)
J	2	$t_{iL} = 1$	без задержек		$\bar{t}_{\max}$	гл. 2, п. 3.7
J	2	$t_{iL} = 1$	$G = C_1 \cup C_2 \cup \dots$ $\dots \cup C_p \cup \Gamma$		$L_{\max}$	гл. 2, п. 3.7
J	2	$t_{iL} \in \{1, 2\}$			$L_{\max}$	гл. 2, п. 3.7
J	3	$t_{iL} = 1$			$L_{\max}$	гл. 2, п. 3.4
J	3	$t_{iL} = 1$			$\bar{t}_{\max}$	гл. 3, п. 1.5
O	2	t_{iL}	без задержек		$\bar{t}_{\max}$	

2*	0	2	t_{iL}	Pr , без задержек $G = C \cup \Gamma$	$\bar{t}_{\max}$	гл. 3, п. 3.10(*)
	0	2	t_{iL}		$\bar{t}_{\max}$	гл. 3, п. 4.1
	0	2	t_{iL}	$Pr, G = C_1 \cup C_2 \cup \dots$ $\dots \cup C_p \cup \Gamma$ $G_0 = C \cup \Gamma$	$\bar{t}_{\max}$	гл. 3, п. 4.5
	0	2	t_{iL}		$\bar{t}_{\max}$	гл. 3, п. 4.6
	0	2	t_{iL}	$\xi_i = \tau > 0$	$\bar{t}_{\max}$	гл. 3, п. 8.4
	0	2	t_{iL}	$\xi_i = \tau > 0$	$\bar{t}_{\max}$	гл. 3, п. 8.4
	0	2	t_{iL} зависит от маршрута		$\bar{t}_{\max}$	гл. 3, п. 8.4(*)
	0	2	$t_{i1} = a_i + \varphi(t_i^0)$ $t_{i2} = b_i + \varphi(t_i^0)$	$\varphi(t)$ — кусочно- постоянная не- убывающая функция	$\bar{t}_{\max}$	гл. 3, п. 8.4
	0	2	t_{iL}		$L_{\max}$	гл. 3, п. 5.6
	0	2	t_{iL}	$D_i = D, i \in N$	$\sum \text{sign}(\max\{\bar{t}_i - D, 0\})$	гл. 3, п. 5.7
	0	2	t_{iL}	$D_i = D, i \in N$	$\sum \text{sign}(\max\{\bar{t}_i - D, 0\})$	гл. 3, п. 5.7
	0	2	t_{iL}		$\sum \bar{t}_i$	гл. 3, п. 7.1, 7.2
	0	2	t_{iL}		$\sum \bar{t}_i, \bar{t}_i < D_i$	гл. 3, п. 8.7
	0	3	t_{iL}	Pr	$\bar{t}_{\max}$	гл. 3, п. 3.9(*)
	0	3	t_{iL}	(Pr)	$\sum \bar{t}_i$	гл. 3, п. 7.3
	0	3	t_{iL}	Pr	$\bar{t}_{\max}$	гл. 3, п. 2.1(*)
	0	4	t_{iL}		$\bar{t}_{\max}$	гл. 3, п. 8.1(*)
	0	0	$t_{iL} \in \{0, 1\}$	$r_i < 2$	$\sum \bar{t}_i$	гл. 3, п. 6.4
	0	0	$t_{iL} \in \{0, 1\}$	Pr	$\sum \bar{t}_i$	гл. 3, п. 6.4
	0	0	$t_{iL} \in \{0, 1\}$	без задержек	$\sum \bar{t}_i$	гл. 3, п. 6.4

Таблица В.3

Тип системы	Число приборов	Длительности операций, моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Оценка сложности	Оценка точности	Раздел книги
F	2	t_{iL}, d_i	π		$\bar{t}_{\max}$	$n \log n$	$\Delta < 1$	гл. 1, п. 9.12
F	2	t_{iL}, d_i	π		$\bar{t}_{\max}$	$n^3 \log n$	$\Delta < 2/3$	гл. 1, п. 9.12
F	2	t_{iL}	$\pi, 0 < b < n - 1$		$\bar{t}_{\max}$	$n \log n$	$\Delta < b/(b + 1)$	гл. 1, п. 9.12
F	2	t_{iL}	π		$L_{\max}$	$n \log n$	$(L_{\max}(s^0) - L_{\max}(s^*)) : (L_{\max}(s^*) + \max\{D_i i \in N\}) < 1$	гл. 1, п. 9.12
F	3	t_{iL}	π		$\bar{t}_{\max}$	n	$\delta < 4t^*$	гл. 1, п. 9.11
F	3	t_{iL}	π		$\bar{t}_{\max}$	$n \log n$	$\delta < 3t^*$	гл. 1, п. 9.11
F	4	t_{iL}	π		$\bar{t}_{\max}$	n^2	$\delta < 9t^*$	гл. 1, п. 9.11
F	4	t_{iL}	π		$\bar{t}_{\max}$	n^2	$\delta < 9t^*$	гл. 1, п. 9.11
F	4	t_{iL}	$P\pi$		$\bar{t}_{\max}$	n^2	$\delta < 9t^*$	гл. 1, п. 9.11
F	M	t_{iL}	π		$\bar{t}_{\max}$	$n^2 M^3$	$\delta < M(M - 3)t^*$	гл. 1, п. 8.2
F	M	t_{iL}	π		$\bar{t}_{\max}$	$n^2 M^3$	$\delta < M(M - 1)t^*$	гл. 1, п. 8.2
F	M	t_{iL}	$P\pi$		$\bar{t}_{\max}$	$n^2 M^3$	$\delta < M(M - 1)t^*$	гл. 1, п. 8.2
F	M	t_{iL}	$P\pi$		$\bar{t}_{\max}$	nM	$\Delta < M - 1$	гл. 1, п. 8.3, 1)
F	M	t_{iL}	$P\pi$		$\bar{t}_{\max}$	$nM \log n$	$\Delta < M' - 1$	гл. 1, п. 8.4, 1)
F	M	t_{iL}	$P\pi$		$\bar{t}_{\max}$	$nM \log n$	$\Delta < M' - 1$	гл. 1, п. 8.4, 1)

F	M	t_{iL}	π	$\bar{t}_{\max}$	$nM + n \log n$	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
F	M	t_{iL}		$\bar{t}_{\max}$	$nM + n \log n$	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
F	M	t_{iL}	P_r	$\bar{t}_{\max}$	$nM + n \log n$	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
F	M	t_{iL}	π , без задержек	$\bar{t}_{\max}$	$nM + n \log n$	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
F	M	t_{iL}	$G_0 = SP, \pi$	$\bar{t}_{\max}$	$nM + n \log n$	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
F	M	t_{iL}	π , без простоев	$\bar{t}_{\max}$	$nM \log n$	$(\bar{t}_{\max}(s^0) -$ $-\bar{t}_{\max}(s^*)) :$ $:(\bar{t}_{\max}(s^*) +$ $+ \sum_{i=1}^n \sum_{L=2}^{M-1} t_{iL}) <$ $< 1 - \frac{1}{M-1}$	гл. 1, п. 9.12
F	M	t_{iL}		$\sum \bar{t}_i$	nM	$\Delta \leq n - 1$	гл. 1, п. 8.3, 2)
F	M	t_{iL}		$\sum \bar{t}_i$	$nM + n \log n$	$\Delta \leq M - 1$	гл. 1, п. 8.5
J	3	t_{iL}	$r_i \leq 3$	$\bar{t}_{\max}$	n^2	$\delta \leq 6t^*$	гл. 2, п. 6.3
J	M	t_{iL}		$\bar{t}_{\max}$	$n^2 M^2 r^2$	$\delta \leq r(M^2 r^2 +$ $+ r - 2)t^*$	гл. 2, п. 6.3
J	M	t_{iL}	идентичные приборы на каждой стадии	$\bar{t}_{\max}$	$M^2 n^2 r^2 +$ $+ (rn +$ $+ M)^2 \log rn$	$\delta \leq (r^2 M^2 +$ $+ r^2 - 2r +$ $+ 2)t^*$	гл. 2, п. 6.3
J	M	t_{iL}	$L^i = (1, 2, \dots, M)$ $i = 1, \dots, k$ $L^j = (M, M - 1, \dots, 1),$ $j = k + 1, \dots, n$	$\bar{t}_{\max}$	$n^2 M^2$	$\delta \leq (2M^2 -$ $-(M + 3)/2)t^*$	гл. 2, п. 6.3

Таблица В.3 (окончание)

Тип системы	Число приборов	Длительности операций, моменты поступления	Условия обслуживания	Дополнительные условия	Критерий	Оценка сложности	Оценка точности	Раздел книги
J	M	t_{iL}			$\bar{t}_{\max}$	nM	$\Delta \leq M - 1$	гл. 2, п. 6.3
J	M	t_{iL}			$\sum \bar{t}_i$	nM	$\Delta \leq n - 1$	гл. 2, п. 6.3
J	M	t_{iL}			$\sum \bar{t}_i$	$n \log n$	$\Delta \leq M - 1$	гл. 2, п. 6.3
J	M	t_{iL}			$\bar{t}_{\max}$	$r(N) + Mr$	$\Delta \leq \min\{M, r\}$	гл. 2, п. 6.3
O	M	t_{iL}			$\bar{t}_{\max}$	nM	$\delta \leq t^*(M - 1)$	гл. 1, п. 8.1
O	M	t_{iL}			$\bar{t}_{\max}$	M	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 1)
							$\Delta \leq M' - 1$	гл. 3, п. 8.1
O	M	t_{iL}			$\bar{t}_{\max}$	nM	$\Delta \leq M' - 1$	гл. 1, п. 8.4, 2)
							$\Delta \leq M' - 1$	гл. 3, п. 8.1
O	M	t_{iL}			$\sum \bar{t}_i$	$nM + n \log n$	$\Delta \leq M - 1$	гл. 3, п. 8.7
O	M	t_{iL}			$\sum \bar{t}_i$	nM	$\Delta \leq n - 1$	гл. 3, п. 8.7

обслуживаться другими приборами. Если не указано ни одно из этих значений, то прерывания запрещены.

Если на множестве N требований задано отношение $\rightarrow$ строгого порядка и запись $i \rightarrow j$ ($i, j \in N, i \neq j$) означает, что должно выполняться неравенство $\min \{t_{jL}^0 \mid L \in \mathcal{M}\} \geq \max \{t_{iL} \mid L \in \mathcal{M}\}$, то граф редукции такого отношения обозначается символом G . Если же запись $i \rightarrow j$ ($i, j \in N, i \neq j$) означает, что должно выполняться неравенство $t_{jL}^0 \geq \bar{t}_{iL}$ для каждого $L \in \mathcal{M}$, то для обозначения графа редукции употребляется символ G_0 .

В тех случаях, когда требуется уточнить структуру графа G (или G_0), могут использоваться следующие обозначения:

$G = SP$ — если граф редукции представляет собой последовательно-параллельный граф;

$G = C$ — если граф редукции — цепь, иными словами, если множество N линейно упорядочено;

$G = C_1 \cup C_2 \cup \dots \cup C_p \cup \Gamma$ — если каждая компонента связности графа редукции представляет собой цепь C_i (и число цепей равно $p \geq 1$) либо одновершинный граф.

Эти же обозначения могут употребляться и для графа G_0 . Если для графа G или G_0 не указано ни одно из перечисленных обозначений, то граф редукции отношения $\rightarrow$ является произвольным бесконтурным. В случае, когда не указаны ни символ G , ни символ G_0 , множество N требований не упорядочено.

Параметр "ресурсы" принимает значение $Rs(q)$, когда имеются ограничения на используемые при обслуживании требований ресурсы (отличные от обслуживающих приборов) и число видов ресурсов равно q . Через R_k и ρ_{ik} обозначается соответственно запас ресурса вида k и его потребление при обслуживании требования i в любой момент времени.

Запись "без задержек" означает, что каждое требование обслуживается без задержек: начавшись в момент времени t_i^0 , обслуживание требования i завершается в момент времени $\bar{t}_i = t_i^0 + \sum_{L \in \mathcal{M}} t_{iL}$.

Запись "без простоев" означает, что каждый прибор $L \in \mathcal{M}$, начиная с некоторого момента времени t_L^0 , функционирует без простоев до момента времени $\bar{t}_L = t_L^0 + \sum_{i \in N} t_{iL}$.

Если для систем с одинаковыми маршрутами указан символ π , то оптимальное расписание ищется среди расписаний, при которых все приборы обслуживают требования в одной и той же последовательности.

Запись $0 < b < n - 1$ также употребляется для систем с одинаковыми маршрутами и означает, что в любой момент времени число требований, обслуживание которых предыдущим прибором закончилось, а последующим прибором не началось, не может превосходить значения b .

В пятой графе приводятся различные ограничения на число требований n , число стадий r_i обслуживания требования $i \in N$, маршруты L^i обслуживания требований $i \in N$ и некоторые другие условия.

Запись $[D_i]$ означает, что все директивные сроки целочисленные, а запись $D_i = D$ — что все они одинаковые.

В тех случаях, когда необходимо указать вид матрицы длительностей обслуживания, приводится либо номер соотношения из главы 1, которому должны удовлетворять элементы этой матрицы, либо наименование типа матрицы в соответствии с терминологией, принятой в § 6 и 7 гл. 1. Наличие в системе одного или нескольких доминирующих приборов указывается явно (определение доминирующего прибора приводится в § 2 гл. 3).

Для систем с одинаковыми маршрутами при $M = 2$ могут быть заданы длительности $\delta_{iL} \geq 0$ и $\tau_{iL} \geq 0$ пуско-наладочных и переналадочных работ, выполняемых для прибора L перед обслуживанием и после обслуживания требования i прибором $L \in \{1, 2\}$ соответственно. Могут также быть заданы величины $\zeta_i \geq 0$, $\eta_i \geq 0$ и $\xi_i \geq 0$ такие, что при любом расписании должны выполняться неравенства $t_{iL}^0 - t_{i1}^0 \geq \zeta_i$, $t_{i2} - t_{i1} \geq \eta_i$, $t_{i2}^0 - t_{i1}^0 \geq \xi_i$, $i \in N$.

Большинство приведенных в таблицах задач состоит в минимизации функции $F(\bar{t}(s))$, вид которой указан в шестой графе с использованием принятых обозначений.

В седьмой графе табл. В.1 приводятся асимптотические оценки сложности (с точностью до постоянного множителя) алгоритмов решения задач. Запись ЛП в этой графе означает, что рассматриваемая задача сводится к задаче линейного программирования. Задачи, в которых разрешение прерываний не приводит к уменьшению целевого функционала, отмечены в этой таблице знаком (*).

В табл. В.2 знаком (*) отмечены NP -трудные задачи, для которых псевдополиномиальные алгоритмы не известны, а NP -трудность в сильном смысле не установлена. Остальные задачи являются NP -трудными в сильном смысле.

В табл. В.3 приводятся сведения о полиномиальных приближенных алгоритмах: в графе 7 — асимптотические оценки сложности алгоритмов, а в графе 8 — оценки точности получаемых решений. Как правило, в качестве оценки точности получаемого решения s^0 используется оценка его относительной погрешности $\Delta = (F(\bar{t}(s^0)) - F(\bar{t}(s^*))) / F(\bar{t}(s^*))$ или абсолютной погрешности $\delta = F(\bar{t}(s^0)) - F(\bar{t}(s^*))$, где s^* — оптимальное расписание.

В таблицах наряду с уже введенными обозначениями используются следующие: $r = \max\{r_i | i \in N\}$, $r(N) = \sum_{i \in N} r_i$, $t^* = \max\{t_{iL} | i \in N, L = L_i^q$,

$q = 1, \dots, r_i\}$, $M' = \left\lceil \frac{M}{2} \right\rceil$, $t^*(N) = \max\{\sum_{i \in N} t_{iL} | L \in \mathcal{M}\}$, $H = |\{t_{iL} >$

$> 0 | i \in N, L \in \mathcal{M}\}|$. Через m обозначено наибольшее число требований из множества N с одинаковым маршрутом обслуживания, а через $\lceil a \rceil$ — наименьшее целое число, которое не меньше a .

В табл. В.1, В.3 и далее в книге все логарифмы берутся по основанию 2.

В табл. В.1, как правило, не приводятся специальные случаи рассматриваемых задач, если для них не известны более простые алгоритмы решения. Под специальным случаем задачи A понимается такая задача B , что множество всех наборов входных данных задачи A содержит в качестве подмножества все наборы входных данных задачи B .

В табл. В.2 приведены в основном "минимальные" *NP*-трудные задачи, т.е. задачи, специальные случаи которых являются либо полиномиально разрешимыми, либо для них не установлена принадлежность множеству *NP*-трудных. Очевидно, если задача имеет *NP*-трудные специальные случаи, то и сама она является *NP*-трудной. Так, все результаты относительно *NP*-трудности задач с одинаковыми маршрутами обслуживания требований справедливы и для задач с различными маршрутами. Часть задач не включена в табл. В.1 и В.2, поскольку их полиномиальная разрешимость или *NP*-трудность следует из приведенных в таблицах результатов для задач с другими критериями оптимальности. Сравнительная сложность решения задач с различными критериями оптимальности рассматривается в [350, 569, 646].

В табл. В.2 не приведены также задачи, которые являются *NP*-трудными в случае одного или нескольких параллельных (одинаковых) приборов, иными словами, для одностадийных систем обслуживания. Эти задачи, как и результаты по трудоемкости алгоритмов их точного и приближенного решения, представлены в соответствующих таблицах [350].

Множество $N = \{1, 2, \dots, n\}$ требований обслуживается в системе, состоящей из M последовательных приборов. Требование $i \in N$ поступает в систему в момент времени $d_i \geq 0$. Для всех требований множества N маршруты обслуживания (порядки прохождения приборов) одинаковы и определяются последовательностью приборов $1, 2, \dots, M$. Длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$, предполагаются заданными. Одновременное обслуживание любого требования несколькими приборами не допускается. Каждый прибор обслуживает не более одного требования одновременно. Такие обслуживающие системы в этой главе называются *системами поточного типа* (с последовательными приборами и одинаковым порядком их прохождения).

Для доказательства NP -трудности задач теории расписаний в данной главе используются следующие эталонные задачи.

Задача о разбиении. Дано множество $N^0 = \{1, 2, \dots, n_0\}$, каждому элементу i которого сопоставлено натуральное число e_i , и $\sum_{i \in N^0} e_i = 2E$. Существует ли разбиение множества N^0 на два подмножества N_1^0 и N_2^0 таких, что $\sum_{i \in N_1^0} e_i = \sum_{i \in N_2^0} e_i = E$?

Задача о 3-разбиении. Дано множество $N^0 = \{1, 2, \dots, 3n_0\}$ и натуральное число E . Каждому элементу $i \in N^0$ сопоставлено натуральное число e_i такое, что $E/4 < e_i < E/2$ и $\sum_{i \in N^0} e_i = n_0 E$. Существует ли такое разбиение множества N^0 на n_0 трехэлементных подмножеств N_j^0 , что $\sum_{i \in N_j^0} e_i = E$, $j = 1, \dots, n_0$?

Задача о трехмерном паросочетании. Даны три попарно непересекающихся множества X_1, X_2 и X_3 натуральных чисел, $|X_1| = |X_2| = |X_3| = n_0$, и множество $E \subseteq X_1 \times X_2 \times X_3$ упорядоченных троек элементов этих множеств. Существует ли такое подмножество $E' \subseteq E$, называемое *трехмерным паросочетанием*, что $|E'| = n_0$ и любые две тройки, вошедшие в E' , не совпадают ни по одной компоненте?

Задача о гамильтоновом цикле. Дан неориентированный граф $\tilde{G} = (V, \tilde{U})$ без петель. Существует ли в графе $\tilde{G}$ гамильтонов цикл?

Задача о 3-разбиении NP -полна в сильном смысле [531]. Остальные перечисленные задачи являются NP -полными [133].

§ 1. Общее время обслуживания. Два прибора

В этом параграфе устанавливаются некоторые свойства оптимальных по быстродействию расписаний обслуживания n требований в системе с M последовательными приборами и одинаковым порядком их прохождения для всех требований. Особое внимание уделяется рассмотрению ситуации, когда $M = 2$.

1.1. В обслуживающую систему, состоящую из M последовательных приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Каждое требование сначала обслуживается прибором 1, затем прибором 2 и т.д., пока оно не будет обслужено прибором M . Длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L, L = 1, \dots, M$, предполагаются заданными. Если процесс обслуживания требования i прибором L начинается в момент времени t_{iL}^0 , то он протекает непрерывно и завершается в момент времени $\bar{t}_{iL} = t_{iL}^0 + t_{iL}$. Каждый прибор одновременно может обслуживать не более одного требования, а каждое требование одновременно может обслуживаться не более чем одним прибором.

В рассматриваемом случае расписание s обслуживания требований однозначно определяется заданием матрицы $\|t_{iL}^0\|$ моментов начала или матрицы $\|\bar{t}_{iL}\|$ моментов завершения обслуживания требований приборами. При этом момент завершения обслуживания всех требований всеми приборами равен

$$\bar{t}_{\max}(s) = \max \{ \bar{t}_{iL} \mid i \in N, L = 1, \dots, M \}.$$

Задача состоит в построении расписания s^* , которому соответствует наименьшее значение $\bar{t}_{\max}(s)$. Это расписание будем называть *оптимальным (по быстродействию) расписанием*.

Если при некотором расписании s прибор 1 обслуживает требования в последовательности $\pi_1 = (i_1, i_2, \dots, i_n)$, а некоторый прибор $L, 2 \leq L \leq M$, — в последовательности $\pi_L = (j_1, j_2, \dots, j_n)$, то имеют место следующие очевидные неравенства:

$$t_{i_1 1}^0 \geq 0, \quad t_{i_k 1}^0 \geq \bar{t}_{i_{k-1} 1}; \quad t_{j_1 L}^0 \geq \bar{t}_{j_1, L-1};$$

$$t_{j_k L}^0 \geq \max \{ \bar{t}_{j_k, L-1}, \bar{t}_{j_{k-1} L} \}$$

для всех $k = 2, \dots, n$.

Если $t_{i_1 1}^0 = 0$, то $\bar{t}_{\max}(s)$ — общее время обслуживания требований. При построении расписания s^* можно ограничиться рассмотрением класса S расписаний, при которых все приведенные нестрогие неравенства выполняются как равенства. Каждое расписание $s \in S$, очевидно, однозначно определяется заданием последовательностей (перестановок) π_L обслуживания требований каждым прибором $L, L = 1, \dots, M$. В дальнейшем рассматривается именно этот класс расписаний.

1.2. Рассмотрим некоторые свойства оптимальных (по быстродействию) расписаний.

1) Покажем, что расписание s^* можно искать в классе расписаний, при которых приборы 1 и 2 обслуживают требования в одной и той же последовательности.

Пусть $s \in S$. Предположим, что при этом расписании прибор 1 обслуживает требования в последовательности $\pi_1 = (i_1, i_2, \dots, i_n)$, а прибор 2 — в последовательности $\pi_2 = (j_1 = i_1, j_2 = i_2, \dots, j_p = i_p, j_{p+1} = i_{r+1}, \dots, j_{p+q} = i_r, \dots, j_n)$, $q > 1, r > p \geq 0$.

Для расписания s справедливы соотношения

$$t_{i_{r+1}2}^0(s) = \max \left\{ \sum_{k=1}^{r+1} t_{i_k1}, \bar{t}_{j_p2}(s) \right\},$$

$$t_{i_r2}^0(s) = \max \left\{ \sum_{k=1}^r t_{i_k1}, \bar{t}_{j_{p+q-1}2}(s) \right\} = \bar{t}_{j_{p+q-1}2}(s) \geq$$

$$\geq \bar{t}_{i_{r+1}2}(s) \geq \bar{t}_{i_{r+1}1}(s) = \sum_{k=1}^{r+1} t_{i_k1}.$$

Рассмотрим расписание $s' \in S$, отличающееся от расписания s только тем, что прибор 1 обслуживает требования в последовательности $\pi'_1 = (i_1, \dots, i_{r-1}, i_{r+1}, i_r, i_{r+2}, \dots, i_n)$.

Для расписания s' справедливы соотношения

$$t_{i_{r+1}2}^0(s') = \max \left\{ \sum_{k=1}^{r-1} t_{i_k1} + t_{i_{r+1}1}, \bar{t}_{j_p2}(s') \right\},$$

$$t_{i_r2}^0(s') = \max \left\{ \sum_{k=1}^{r+1} t_{i_k1}, \bar{t}_{j_{p+q-1}2}(s') \right\}.$$

Так как $\bar{t}_{j_p2}(s) = \bar{t}_{j_p2}(s')$, то $t_{i_{r+1}2}^0(s) \geq t_{i_{r+1}2}^0(s')$. Отсюда следует, что $\bar{t}_{j_{p+q-1}2}(s) \geq \bar{t}_{j_{p+q-1}2}(s')$ и, следовательно, $t_{i_r2}^0(s) \geq t_{i_r2}^0(s')$.

Повторяя описанные рассуждения не более чем $O(n^2)$ раз, приходим к расписанию $\tilde{s}$, при котором оба прибора 1 и 2 обслуживают требования в одной и той же последовательности, причем выполняется соотношение $t_{i_2}^0(s) \geq t_{i_2}^0(\tilde{s})$, $i \in N$. Тем самым $\bar{t}_{\max}(\tilde{s}) \leq \bar{t}_{\max}(s)$.

2) Покажем, что расписание s^* можно искать в классе расписаний, при которых приборы $M-1$ и M обслуживают требования в одной и той же последовательности.

Пусть $s \in S$. Предположим, что при этом расписании прибор M обслуживает требования в последовательности $\pi_M = (j_1, j_2, \dots, j_{r-1}, j_r, j_{r+1}, j_{r+2}, \dots, j_n)$, а прибор $M-1$ — в последовательности $\pi_{M-1} = (i_1 = j_1, i_2 = j_2, \dots, i_p = j_p, i_{p+1} = j_{r+1}, \dots, i_{p+q} = j_r, \dots, j_n)$, $q > 1, r > p \geq 0$.

Для расписания s справедливы соотношения

$$t_{j_rM}^0(s) = \max \{ \bar{t}_{j_{r-1}M}(s), \bar{t}_{j_r, M-1}(s) \},$$

$$t_{j_{r+1}M}^0(s) = \max \{ \bar{t}_{j_rM}(s), \bar{t}_{j_{r+1}, M-1}(s) \} = \bar{t}_{j_rM}(s).$$

Рассмотрим расписание $s' \in S$, отличающееся от расписания s только тем, что прибор M обслуживает требования в последовательности $\pi'_M = (j_1, j_2, \dots, j_{r-1}, j_{r+1}, j_r, j_{r+2}, \dots, j_n)$.

Для расписания s' справедливы соотношения

$$\begin{aligned} t_{j_r+1M}^0(s') &= \max \{ \bar{t}_{j_r-1M}(s'), \bar{t}_{j_r+1, M-1}(s') \} = \\ &= \max \{ \bar{t}_{j_r-1M}(s), \bar{t}_{j_r+1, M-1}(s') \}, \\ t_{j_rM}^0(s') &= \max \{ \bar{t}_{j_r+1M}(s'), \bar{t}_{j_r, M-1}(s') \} = \\ &= \max \{ \bar{t}_{j_r+1M}(s'), \bar{t}_{j_r, M-1}(s) \}. \end{aligned}$$

Поскольку $\bar{t}_{j_r+1, M-1}(s) \leq \bar{t}_{j_r, M-1}(s)$, то $t_{j_r+1M}^0(s') \leq \max \{ \bar{t}_{j_r-1M}(s), \bar{t}_{j_r, M-1}(s) \} = t_{j_rM}^0(s)$. Тогда

$$\begin{aligned} \bar{t}_{j_rM}(s') &= t_{j_rM}^0(s') + t_{j_rM} \leq \\ &\leq \max \{ t_{j_rM}^0(s) + t_{j_r+1M}, \bar{t}_{j_r, M-1}(s) \} + t_{j_rM} = \\ &= t_{j_rM}^0(s) + t_{j_r+1M} + t_{j_rM} = \bar{t}_{j_r+1M}(s). \end{aligned}$$

Таким образом,

$$\begin{aligned} \max \{ \bar{t}_{j_rM}(s'), \bar{t}_{j_r+1M}(s') \} &= \bar{t}_{j_rM}(s') \leq \\ &\leq \bar{t}_{j_r+1M}(s) = \max \{ \bar{t}_{j_r+1M}(s), \bar{t}_{j_rM}(s) \}. \end{aligned}$$

Повторяя описанные рассуждения не более чем $O(n^2)$ раз, приходим к расписанию $\bar{s}$, при котором оба прибора $M-1$ и M обслуживают требования в одной и той же последовательности, причем $\bar{t}_{\max}(\bar{s}) \leq \bar{t}_{\max}(s)$.

3) Из п. 1) и 2) непосредственно следует, что при $M \leq 3$ оптимальное по быстродействию расписание s^* можно искать среди расписаний, при которых все приборы обслуживают требования в одной и той же последовательности.

Покажем, что при $M > 3$ это, вообще говоря, не имеет места.

Действительно, рассмотрим задачу построения оптимального по быстродействию расписания обслуживания двух требований четырьмя последовательными приборами. Длительности обслуживания приведены в табл. 1.1.

Таблица 1.1

$i \backslash L$				
	1	2	3	4
1	4	1	1	4
2	1	4	4	1

Если все приборы обслуживают требования в одной и той же последовательности (1, 2) или (2, 1), то общее время обслуживания равно 14 единицам времени. Если приборы 1 и 2 обслуживают требования в последовательности (2, 1), а приборы 3 и 4 – в последовательности (1, 2), то получаем расписание, при котором общее время обслуживания равно 12 единицам времени (рис. 1.1).

4) При любом расписании $s \in S$ прибор 1 функционирует без простоев, начиная с момента времени $d = 0$. Покажем, что расписание s может быть преобразовано в некоторое новое расписание $\hat{s}$ (не обязательно из класса S), при котором $\bar{t}_{\max}(\hat{s}) = \bar{t}_{\max}(s)$ и приборы 1 и M функционируют без простоев, начиная с моментов времени 0 и $\bar{t}_{\max}(s) - \sum_{i=1}^n t_{iM}$ соответственно.

Действительно, пусть $s \in S$ и прибор M при этом расписании обслуживает требования в последовательности $\pi_M = (j_1, j_2, \dots, j_M)$. Положим

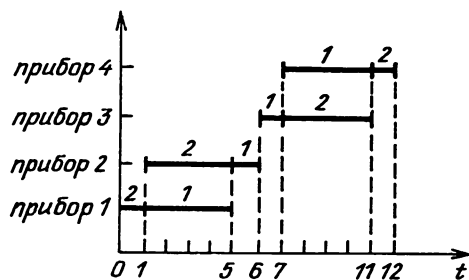


Рис. 1.1

$\bar{t}_{iL}(\hat{s}) = \bar{t}_{iL}(s)$ для всех $i = 1, \dots, n, L = 1, \dots, M-1$; $\bar{t}_{j_n M}(\hat{s}) = \bar{t}_{\max}(s)$, $\bar{t}_{j_k M}(\hat{s}) = \bar{t}_{\max}(s) - \sum_{i=k+1}^n t_{j_i M}, k = 1, \dots, n-1$. Поскольку $t_{j_k M}^0(\hat{s}) \geq \geq t_{j_k M}^0(s), k = 1, \dots, n$, и $t_{j_1 M}^0(\hat{s}) = \bar{t}_{\max}(s) - \sum_{i=1}^n t_{iM}$, то $\hat{s}$ — искомое расписание.

5) Рассмотрим часто встречающуюся ситуацию, которая отличается от описанной в п. 1.1 тем, что заранее оговорено, что все приборы обслуживают требования множества N в одной и той же последовательности.

Пусть $\pi = (i_1, i_2, \dots, i_n)$ — некоторая перестановка элементов множества $N = \{1, 2, \dots, n\}$. Рассмотрим расписание $s \in S$, при котором все приборы обслуживают требования в последовательности π .

Для этого расписания справедливы соотношения

$$t_{i_1 1}^0 = 0, \quad t_{i_k 1}^0 = \bar{t}_{i_{k-1} 1}, \quad k = 2, \dots, n;$$

$$t_{i_1 L}^0 = \bar{t}_{i_1 L-1}, \quad L = 2, \dots, M;$$

$$t_{i_k L}^0 = \max \{ \bar{t}_{i_{k-1} L}, \bar{t}_{i_k L-1} \}, \quad k = 2, \dots, n, L = 2, \dots, M.$$

Нетрудно убедиться, что в этом случае общее время обслуживания равно

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \\ &= \max_{1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n} \left\{ \sum_{k=1}^{u_1} t_{i_k 1} + \sum_{k=u_1}^{u_2} t_{i_k 2} + \dots + \sum_{k=u_{M-1}}^n t_{i_k M} \right\}, \end{aligned} \quad (1.1)$$

где $u_1, u_2, \dots, u_{M-1}$ — целые числа.

1.3. Перейдем к более подробному рассмотрению случая, когда $M = 2$.

Пусть система состоит из двух приборов A и B , длительности обслуживания требования $i \in N$ приборами A и B равны соответственно a_i и b_i .

Как уже отмечалось, в рассматриваемом случае расписание s^* можно искать в классе расписаний, при которых оба прибора функционируют без простоев и обслуживают требования в одной и той же последовательности.

Пусть $\pi = (i_1, i_2, \dots, i_n)$ — произвольная перестановка элементов множества $N = \{1, 2, \dots, n\}$.

Поскольку требования обоими приборами обслуживаются в одной и той же последовательности, то для расписания $s \in S$, определяемого последовательностью π , из (1.1) следует, что

$$\bar{t}_{\max}(\pi) = \max_{u \in N} \left\{ \sum_{k=1}^u a_{i_k} + \sum_{k=u}^n b_{i_k} \right\}.$$

Преобразуя расписание s в расписание $\hat{s}$ в соответствии с п. 1.2, 4), получаем

$$\bar{t}_{\max}(\pi) = t_{i_1, B}^0(\hat{s}) + \sum_{i=1}^n b_i = \max_{u \in N} \left\{ \sum_{k=1}^u a_{i_k} - \sum_{k=1}^{u-1} b_{i_k} \right\} + \sum_{i=1}^n b_i.$$

Таким образом, задача построения оптимального (по быстродействию) расписания s^* сводится к нахождению перестановки π^* , которой соответствует наименьшее значение $\max \{ \Delta_u(\pi) \mid u \in N \}$, где

$$\Delta_u(\pi) = \sum_{k=1}^u a_{i_k} - \sum_{k=1}^{u-1} b_{i_k}, \quad u \in N.$$

Введем в рассмотрение перестановку π' , отличающуюся от π транспозицией элементов i_r и i_{r+1} , т.е. $\pi' = (i_1, i_2, \dots, i_{r-1}, i_{r+1}, i_r, i_{r+2}, \dots, i_n)$. Найдем достаточные условия, при которых

$$\max \{ \Delta_u(\pi) \mid u \in N \} \leq \max \{ \Delta_u(\pi') \mid u \in N \}. \quad (1.2)$$

Очевидно, $\Delta_u(\pi) = \Delta_u(\pi')$ при всех $u \in \{1, 2, \dots, r-1, r+2, \dots, n\}$. Следовательно, для выполнения неравенства (1.2) достаточно, чтобы

$$\max \{ \Delta_r(\pi), \Delta_{r+1}(\pi) \} \leq \max \{ \Delta_r(\pi'), \Delta_{r+1}(\pi') \}.$$

Последнее неравенство равносильно неравенству

$$\max \{ a_{i_r}, a_{i_r} + a_{i_{r+1}} - b_{i_r} \} \leq \max \{ a_{i_{r+1}}, a_{i_r} + a_{i_{r+1}} - b_{i_{r+1}} \},$$

которое в свою очередь равносильно неравенству

$$\min \{ a_{i_r}, b_{i_{r+1}} \} \leq \min \{ a_{i_{r+1}}, b_{i_r} \}. \quad (1.3)$$

Таким образом, из неравенства (1.3) следует неравенство (1.2).

Пусть $i, j, k \in N$. Легко показать, что в случае, когда имеют место неравенства

$$\min \{ a_i, b_j \} \leq \min \{ a_j, b_i \}, \quad \min \{ a_j, b_k \} \leq \min \{ a_k, b_j \},$$

выполняется также неравенство

$$\min \{ a_i, b_k \} \leq \min \{ a_k, b_i \}.$$

Отсюда следует, что если неравенство (1.3) выполняется для всех $r = 1, \dots, n-1$, то $\pi = (i_1, i_2, \dots, i_n)$ является искомой оптимальной перестановкой.

Таким образом, для построения оптимальной перестановки достаточно разбить множество N на два (не обязательно непустых) подмножества $N_1 = \{i | a_i \leq b_i\}$ и $N_2 = \{i | a_i > b_i\}$ и сначала обслужить требования множества N_1 в порядке неубывания значений a_i , а затем — требования множества N_2 в порядке невозрастания значений b_i .

Для построения оптимальной перестановки можно также сопоставить каждому требованию i вес

$$w(i) = (W - \min\{a_i, b_i\}) \operatorname{sign}(a_i - b_i),$$

где W — достаточно большое число ($W > \max\{\min\{a_i, b_i\} | i \in N\}$), и упорядочить требования в порядке неубывания весов.

В любом случае трудоемкость построения оптимальной перестановки не превосходит $O(n \log n)$ операций.

1.4. Рассмотрим некоторые обобщения ситуации, описанной в предыдущем пункте.

Пусть по-прежнему имеется два прибора A и B . Каждое требование $i \in N = \{1, 2, \dots, n\}$ обслуживается начиная с прибора A , известны длительности $a_i \geq 0$ и $b_i \geq 0$ обслуживания требования i приборами A и B соответственно. Если требование i начинает обслуживаться прибором A в момент времени t_{iA}^0 , то момент времени начала его обслуживания прибором B $t_{iB}^0 \geq t_{iA}^0 + \xi_i$, где ξ_i — данное число. Если $\xi_i < a_i$, то требование i может одновременно обслуживаться обоими приборами. Будем предполагать также заданными наименьшие промежутки времени $\delta_{ik}^A \geq 0$ и $\delta_{ik}^B \geq 0$ между возможными моментами начала обслуживания требований i и k приборами A и B соответственно при условии, что требование i начинается обслуживаться первым. Если $\delta_{ik}^A < a_i$, то требования i и k могут обслуживаться прибором A одновременно. Аналогично, если $\delta_{ik}^B < b_i$, то эти требования могут одновременно обслуживаться прибором B . Пусть $\tau_{iA} \geq 0$ и $\tau_{iB} \geq 0$ — заданные моменты времени, раньше которых не может быть начато обслуживание требования i приборами A и B соответственно.

Если положить $\xi_i = \delta_{ik}^A = a_i$, $\delta_{ik}^B = b_i$, $\tau_{iA} = \tau_{iB} = 0$, $i, k \in N$, то получим ситуацию, описанную в предыдущем пункте.

Через $t_{\max}(\pi, \pi')$ обозначим наименьший момент времени, к которому можно завершить обслуживание всех требований обоими приборами при условии, что прибор A обслуживает требования в последовательности $\pi = (i_1, i_2, \dots, i_n)$, а прибор B — в последовательности $\pi' = (j_1, j_2, \dots, j_n)$.

По условию обслуживание требования i_1 прибором A может быть начато в момент времени $t_{i_1 A}^0 = \tau_{i_1 A}$, требования i_2 — в момент времени $t_{i_2 A}^0 = \max\{\tau_{i_2 A}, t_{i_1 A}^0 + \delta_{i_1 i_2}^A\}$ и т.д., требования i_n — в момент времени $t_{i_n A}^0 = \max\{\tau_{i_n A}, t_{i_1 A}^0 + \delta_{i_1 i_2}^A, \dots, t_{i_{n-1} A}^0 + \delta_{i_{n-1} i_n}^A\}$.

Самый ранний момент начала обслуживания требования j_1 прибором B равен $t_{j_1 B}^0 = \max\{\tau_{j_1 B}, t_{i_1 A}^0 + \xi_{j_1}\}$, требования j_2 — $t_{j_2 B}^0 = \max\{\tau_{j_2 B}, t_{j_1 A}^0 + \xi_{j_2}, t_{j_1 B}^0 + \delta_{j_1 j_2}^B\}$ и т.д., требования j_n — $t_{j_n B}^0 = \max\{\tau_{j_n B}, t_{i_n A}^0 + \xi_{j_n}, t_{j_1 B}^0 + \delta_{j_1 j_n}^B, t_{j_2 B}^0 + \delta_{j_2 j_n}^B, \dots, t_{j_{n-1} B}^0 + \delta_{j_{n-1} j_n}^B\}$.

Наконец, $\bar{t}_{\max}(\pi, \pi') = \max\{\bar{t}_{i_{kA}}, \bar{t}_{j_{kB}} \mid k \in N\}$, где $\bar{t}_{i_{kA}} = t_{i_{kA}}^0 + a_{i_{kA}}$, $\bar{t}_{j_{kB}} = t_{j_{kB}}^0 + b_{j_{kB}}$, $k \in N$.

1) Сформулируем достаточные условия, при которых неравенство $\bar{t}_{\max}(\pi, \pi) \leq \bar{t}_{\max}(\pi, \pi')$ выполняется для любых перестановок π и π' . Не нарушая общности рассуждений, будем считать, что $\pi = (1, 2, \dots, n)$ и $\pi' = (j_1, j_2, \dots, j_n)$.

Если для любых требований $i, k, r \in N$ выполняются неравенства

$$\delta_{ir}^A \leq \delta_{ik}^A + \delta_{kr}^A, \quad \tau_{kA} - \tau_{iA} \leq \delta_{ik}^A, \quad (1.4)$$

то для всех $i \in N$ справедливо

$$t_{iA}^0 = \tau_{1A} + \sum_{l=1}^{i-1} \delta_{i,l+1}^A.$$

Если для любых требований $i, k, r \in N$ выполняются неравенства

$$\delta_{ir}^B \leq \delta_{ik}^B + \delta_{kr}^B, \quad \tau_{iB} - \tau_{iA} \leq \xi_i, \quad (1.5)$$

то для всех $k \in N$ справедливо

$$t_{j_{kB}}^0 = \max\{t_{j_{rA}}^0 + \xi_{j_r} + \sum_{l=r}^{k-1} \delta_{j_l j_{l+1}}^B \mid r = 1, \dots, k\}.$$

Если наряду с (1.4) и (1.5) для любой пары требований $i, k \in N$ выполняются неравенства

$$a_i - b_i \leq \xi_i, \quad b_i - a_k \leq \delta_{ik}^B, \quad (1.6)$$

то имеет место соотношение

$$\bar{t}_{\max}(\pi, \pi') = \max_{r \in N} \left\{ \tau_{1A} + \sum_{l=1}^{j_r-1} \delta_{i,l+1}^A + \xi_{j_r} + \sum_{l=r}^{n-1} \delta_{j_l j_{l+1}}^B + b_{j_n} \right\}. \quad (1.7)$$

Покажем, что для того чтобы $\bar{t}_{\max}(\pi, \pi) \leq \bar{t}_{\max}(\pi, \pi')$ при любых π и π' , достаточно, чтобы наряду с неравенствами (1.4) – (1.6) выполнялись соотношения

$$\delta_{ik}^B = b_i, \quad \xi_i - \xi_k \leq \delta_{ik}^A. \quad (1.8)$$

для любой пары требований $i, k \in N$.

По-прежнему полагаем $\pi = (1, 2, \dots, n)$ и пусть $\pi' = (j_1, \dots, j_{p-1}, j_p, \dots, j_{p+u}, \dots, j_n)$, где $j_l = l$, $l = 1, \dots, p-1$, $j_p \neq p$, $j_{p+u} = p$, $u \geq 1$. Рассмотрим перестановку $\tilde{\pi}' = (1, 2, \dots, p-1, p, j_p, \dots, j_{p+u-1}, j_{p+u+1}, \dots, j_n)$.

Покажем, что при сделанных предположениях имеет место неравенство $\bar{t}_{\max}(\pi, \tilde{\pi}') \leq \bar{t}_{\max}(\pi, \pi')$.

Действительно, в силу (1.8) выражение (1.7) может быть записано в виде

$$\bar{t}_{\max}(\pi, \pi') = \max_{r \in N} \left\{ \tau_{1A} + \sum_{l=1}^{j_r-1} \delta_{i,l+1}^A + \xi_{j_r} + \sum_{l=r}^n b_{j_l} \right\}.$$

При этом

$$\bar{t}_{\max}(\pi, \tilde{\pi}') = \max \left\{ \max_{\substack{1 \leq r \leq p-1 \\ p+u+1 \leq r \leq n}} \left\{ \tau_{1A} + \sum_{l=1}^{j_r-1} \delta_{i,l+1}^A + \xi_{j_r} + \sum_{l=r}^n b_{j_l} \right\}, \right.$$

$$\tau_{1A} + \sum_{l=1}^{p-1} \delta_{l,l+1}^A + \xi_p + \sum_{l=p}^n b_{jl}, \quad \max_{p \leq r \leq p+u-1} \{ \tau_{1A} + \sum_{l=1}^{p-1} \delta_{l,l+1}^A + \\ + \sum_{l=p+1}^{j_r-1} \delta_{l,l+1}^A + \xi_{j_r} + \sum_{l=r}^n b_{jl} \} \}.$$

Для того чтобы доказать соотношение $\bar{t}_{\max}(\pi, \tilde{\pi}') \leq \bar{t}_{\max}(\pi, \pi')$, достаточно показать, что

$$\tau_{1A} + \sum_{l=1}^{p-1} \delta_{l,l+1}^A + \xi_p + \sum_{l=p}^n b_{jl} \leq \bar{t}_{\max}(\pi, \pi'). \quad (1.9)$$

Имеем

$$\bar{t}_{\max}(\pi, \pi') \geq \tau_{1A} + \sum_{l=1}^{j_p-1} \delta_{l,l+1}^A + \xi_{j_p} + \sum_{l=p}^n b_{jl}.$$

Из (1.8) следует, что

$$\xi_p \leq \sum_{l=p}^{j_p-1} \delta_{l,l+1}^A + \xi_{j_p}$$

и неравенство (1.9) выполняется.

2) Опишем способ построения оптимального расписания в предположении, что выполняются соотношения (1.4) – (1.6), (1.8) и $\delta_{ik}^A = \delta_i^A \geq 0$ для всех $i, k \in N$.

Как показано выше, достаточно рассматривать такие расписания, при которых оба прибора обслуживают требования в одной и той же последовательности. Пусть $\pi = (i_1, i_2, \dots, i_n)$ – некоторая перестановка элементов множества $N = \{1, 2, \dots, n\}$. Тогда

$$\bar{t}_{\max}(\pi) = \max_{u \in N} \{ \tau_{i_1A} + \sum_{k=1}^{u-1} \delta_{i_k}^A + \xi_{i_u} + \sum_{k=u}^n b_{i_k} \}. \quad (1.10)$$

Пусть известно, что в оптимальной перестановке на первом месте стоит требование u . Тогда задача сводится к нахождению перестановки $\pi_u^* = (i_2^*, i_3^*, \dots, i_n^*)$ из множества всех перестановок вида $\pi_u = (i_2, i_3, \dots, i_n)$ элементов множества $N \setminus \{u\}$, которой соответствует наименьшее значение

$$\begin{aligned} \tilde{t}(\pi_u) &= \max_{2 \leq u \leq n} \{ \sum_{k=2}^{u-1} \delta_{i_k}^A + \xi_{i_u} + \sum_{k=u}^n b_{i_k} \} = \\ &= \max_{2 \leq u \leq n} \{ \sum_{k=2}^u (\delta_{i_k}^A - b_{i_k}) + \xi_{i_u} - \delta_{i_u}^A + b_{i_u} \} - \sum_{k=2}^n b_{i_k}. \end{aligned}$$

Полагая $\alpha_{i_k} = \delta_{i_k}^A - b_{i_k}$ и $\beta_{i_u} = \xi_{i_u} - \delta_{i_u}^A + b_{i_u}$, приходим к задаче минимизации функционала

$$f(\pi_u) = \max_{2 \leq u \leq n} \{ \sum_{k=2}^u \alpha_{i_k} + \beta_{i_u} \}$$

на множестве перестановок вида $\pi_u = (i_2, i_3, \dots, i_n)$ элементов множества $N \setminus \{u\}$.

Пусть $\pi'_v = (i_2, \dots, i_{r-1}, i_{r+1}, i_r, i_{r+2}, \dots, i_n)$. Имеем

$$f(\pi'_v) = \max \left\{ \max_{\substack{2 \leq u \leq r-1, \\ r+2 \leq u \leq n}} \left\{ \sum_{k=2}^u \alpha_{i_k} + \beta_{i_u} \right\}, \sum_{k=2}^{r-1} \alpha_{i_k} + \alpha_{i_{r+1}} + \beta_{i_{r+1}}, \right.$$

$$\left. \sum_{k=2}^{r+1} \alpha_{i_k} + \beta_{i_r} \right\}.$$

Для того чтобы $f(\pi_v) \leq f(\pi'_v)$, достаточно, чтобы

$$\begin{aligned} & \max \{ \alpha_{i_r} + \beta_{i_r}, \alpha_{i_r} + \alpha_{i_{r+1}} + \beta_{i_{r+1}} \} \leq \\ & \leq \max \{ \alpha_{i_{r+1}} + \beta_{i_{r+1}}, \alpha_{i_r} + \alpha_{i_{r+1}} + \beta_{i_r} \}. \end{aligned}$$

Это неравенство равносильно неравенству

$$\min \{ \alpha_{i_r} + \beta_{i_r}, \beta_{i_{r+1}} \} \leq \min \{ \alpha_{i_{r+1}} + \beta_{i_{r+1}}, \beta_{i_r} \},$$

для выполнимости которого достаточно, чтобы

$$\begin{aligned} & (W - \min \{ \alpha_{i_r} + \beta_{i_r}, \beta_{i_r} \}) \operatorname{sign}(\alpha_{i_r}) \leq \\ & \leq (W - \min \{ \alpha_{i_{r+1}} + \beta_{i_{r+1}}, \beta_{i_{r+1}} \}) \operatorname{sign}(\alpha_{i_{r+1}}), \end{aligned}$$

где W — достаточно большое число ($W > \max \{ \min \{ \alpha_i + \beta_i, \beta_i \} \mid i \in N \}$).

Таким образом, для нахождения оптимальной перестановки π^* достаточно

а) каждому требованию $i \in N$ сопоставить вес $w(i) = (W - \min \{ \xi_i, \xi_i - \delta_i^A + b_i \}) \operatorname{sign}(\delta_i^A - b_i)$;

б) построить n перестановок вида (v, π_v^*) , где $v \in N$, а требования множества $N \setminus \{v\}$ упорядочены по неубыванию их весов;

в) среди построенных перестановок вида (v, π_v^*) выбрать ту, которой соответствует наименьшее значение величины $t_{\max}(\pi)$, определяемой соотношением (1.10).

Трудоемкость нахождения π^* , очевидно, не превосходит $O(n \log n)$ операций.

§ 2. Общее время обслуживания.

Три и более приборов

В этом параграфе показано, что задача построения оптимального по быстродействию расписания при $M \geq 3$ является NP -трудной независимо от того, допускаются прерывания в процессе обслуживания требований или нет.

2.1. Пусть множество $N = \{1, 2, \dots, n\}$ требований поступает в момент времени $d = 0$ в систему поточного типа, состоящую из трех последовательных приборов. Покажем, что задача построения оптимального (по быстродействию) расписания является NP -трудной в сильном смысле, если в процессе обслуживания не допускаются прерывания и все $t_{iL} > 0$, $i \in N$, $L = 1, 2, 3$.

Сформулируем соответствующую задачу распознавания: определить, существует ли такое расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче сводится задача о 3-разбиении (см. вводную часть этой главы).

Положим $n = 4n_0 + 1$, $N = \{1, 2, \dots, n\}$. Требования множества N разобьем на две группы: U -требования, обозначаемые U_i , $i = 1, \dots, 3n_0$, и V -требования, обозначаемые V_j , $j = 0, \dots, n_0$. Положим

$$t_{U_{i1}} = t_{U_{i3}} = 1, \quad t_{U_{i2}} = e_i, \quad i = 1, \dots, 3n_0;$$

$$t_{V_{01}} = 1, \quad t_{V_{02}} = E, \quad t_{V_{03}} = 2E - 3;$$

$$t_{V_{j1}} = 2E - 3, \quad t_{V_{j2}} = E, \quad t_{V_{j3}} = 2E - 3, \quad j = 1, \dots, n_0 - 1;$$

$$t_{V_{n_01}} = 2E - 3, \quad t_{V_{n_02}} = E, \quad t_{V_{n_03}} = 1.$$

Покажем, что в построенной задаче расписание s_0 , при котором $\bar{t}_{\max}(s^0) \leq y = (2n_0 + 1)E + 2$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

1) Пусть задача о 3-разбиении имеет решение и N_j^0 , $j = 1, \dots, n_0$, — найденные трехэлементные подмножества элементов множества N^0 . Через $\pi_j(U)$ обозначим произвольную перестановку U -требований с номерами из множества N_j^0 , $j = 1, \dots, n_0$.

Расписание s^0 можно построить, например, следующим образом. Все три прибора обслуживают требования множества N в одной и той же последовательности $(V_0, \pi_1(U), V_1, \pi_2(U), \dots, V_{n_0-1}, \pi_{n_0}(U), V_{n_0})$ и функционируют без простоев: прибор 1 — во временном интервале $(0, 2n_0E + 1]$, прибор 2 — в интервале $(1, (2n_0 + 1)E + 1]$ и прибор 3 — в интервале $(E + 1, (2n_0 + 1)E + 2]$.

2) Пусть существует расписание s^0 . Будем полагать, что в задаче о 3-разбиении $E > 8$, так как в противном случае задача о 3-разбиении либо не имеет смысла (при $E \in \{1, 2, 4\}$), либо проверка существования ее решения не вызывает затруднений (при $E \in \{3, 5, 6, 7, 8\}$).

Учитывая результаты п.1.2 § 1 этой главы, можно считать, что при расписании s^0 все три прибора обслуживают требования множества N в одной и той же последовательности π^0 , а приборы 1 и 3 функционируют без простоев.

а) Покажем, что в последовательности π^0 первым должно обслуживаться требование V_0 .

Очевидно, прибор 1 функционирует в интервале $(0, 2n_0E + 1]$. Кроме того, $\min\{t_{i3}^0 \mid i \in N\} \leq E + 1$, так как в противном случае $\bar{t}_{\max}(s^0) > y$. Тогда в последовательности π^0 ни одно из требований $V_1, V_2, \dots, V_{n_0}$ не может обслуживаться первым, поскольку обслуживание этого требования прибором 3 начиналось бы в момент времени $3E - 3 > E + 1$ (при $E > 8$).

Предположим, что первым в последовательности π^0 обслуживается некоторое U -требование. Пусть R_U — множество номеров U -требований, обслуживаемых прибором 1 непрерывно одно за другим в интервале $(0, |R_U|]$, $|R_U| \geq 1$. Поскольку прибор 2 начинает функционировать не раньше чем в момент времени 1, он должен функционировать без простоев в интервале $(1, y - 1]$.

Если вслед за требованиями этого множества прибор 1 обслуживает требование V_0 , то для того чтобы приборы 2 и 3 функционировали без простоев, должны выполняться неравенства

$$1 + \sum_{i \in R_U} e_i \geq |R_U| + 1, \quad E + 1 + |R_U| \geq 1 + \sum_{i \in R_U} e_i + E.$$

Эти неравенства, однако, не могут одновременно иметь места при $E > 8$.

Если после требований множества $\{U_i | i \in R_U\}$ прибор 1 обслуживает одно из требований $V_1, V_2, \dots, V_{n_0}$, то для того чтобы приборы 2 и 3 функционировали без простоев, должны выполняться неравенства

$$1 + \sum_{i \in R_U} e_i \geq |R_U| + 2E - 3, \quad E + 1 + |R_U| \geq 1 + \sum_{i \in R_U} e_i + E.$$

Эти неравенства также одновременно не могут иметь места при $E > 8$.

Таким образом, первым в последовательности π^0 обслуживается требование V_0 .

При этом $t_{V_0,3}^0 = E + 1$ и $\bar{t}_{\max}(s^0) = y$. Поскольку $t_{V_0,2}^0 \geq 1$, $\sum_{i \in N} t_{i2} = y - 2$ и $\min\{t_{i3} | i \in N\} = 1$, то прибор 2 функционирует без простоев в интервале $(1, y - 1]$.

б) Покажем, что в последовательности π^0 последним должно обслуживаться требование V_{n_0} .

Очевидно, ни одно из требований $V_1, V_2, \dots, V_{n_0-1}$ не может обслуживаться последним, поскольку $\max\{t_{i2} | i \in N\} = y - 1$ и $t_{V_j,3} = 2E - 3 > 1$, $j = 1, \dots, n_0 - 1$.

Пусть V_q , $1 \leq q \leq n_0$, — последнее из V -требований в последовательности π^0 . Пусть R_U — множество номеров U -требований, обслуживаемых после требования V_q . Для того чтобы прибор 2 функционировал без простоев, должно выполняться неравенство $\bar{t}_{V_q,1} \leq t_{V_q,2}^0$, или, что то же,

$$2n_0 E - |R_U| + 1 \leq 2n_0 E + 1 - \sum_{i \in R_U} e_i.$$

Последнее неравенство означает, что $|R_U| \geq \sum_{i \in R_U} e_i$. Получаем противоречие.

Поскольку требования $V_1, V_2, \dots, V_{n_0-1}$ не отличаются друг от друга по длительностям обслуживания, то они могут обслуживаться в произвольном порядке. Для удобства дальнейших рассуждений будем предполагать, что в последовательности π^0 эти требования обслуживаются в порядке возрастания их номеров.

в) Пусть R_j — множество номеров U -требований, расположенных в последовательности π^0 между требованиями V_{j-1} и V_j , $j = 1, \dots, n_0$. Покажем, что $|R_j| \in \{2, 3, 4\}$, $j = 1, \dots, n_0$.

Рассмотрим пару требований V_0 и V_1 . Так как все приборы функционируют без простоев в соответствующих интервалах, то имеют место соотношения

$$\bar{t}_{V_0,1} + |R_1| = t_{V_1,1}^0,$$

$$\bar{t}_{V_0,2} + \sum_{i \in R_1} e_i = t_{V_1,2}^0 \geq \bar{t}_{V_0,1} + |R_1| + 2E - 3,$$

$$\bar{t}_{V_0,3} + |R_1| = t_{V_1,3}^0 \geq \bar{t}_{V_1,2}.$$

Поскольку $\bar{t}_{V_0 2} = \bar{t}_{V_0 1} + E$ и $\bar{t}_{V_0 3} = \bar{t}_{V_0 1} + 3E - 3$, то $\bar{t}_{V_1 2} = \bar{t}_{V_0 1} + 2E + \sum_{i \in R_1} e_i$. Таким образом, должно выполняться равенство

$$\sum_{i \in R_1} e_i = E - 3 + |R_1|. \quad (2.1)$$

Нетрудно проверить, что при $|R_1| = 1$ равенство (2.1) не имеет места, если $E > 8$. Если $|R_1| > 4$, то $\sum_{i \in R_1} e_i > |R_1|E/4$ и при $E > 8$ значение

$|R_1|E/4 > E - 3 + |R_1|$, т.е. равенство (2.1) не имеет места.

Таким образом, $|R_1| \in \{2, 3, 4\}$, причем в силу (2.1) $t_{V_1 2}^0 = \bar{t}_{V_1 1}$ и $t_{V_1 3}^0 = \bar{t}_{V_1 2}$.

Рассуждая аналогичным образом, можно показать, что для всех $j = 2, \dots, n_0$ значения $|R_j| \in \{2, 3, 4\}$ и

$$\sum_{i \in R_j} e_i = E - 3 + |R_j|, \quad (2.2)$$

причем $t_{V_j 2}^0 = \bar{t}_{V_j 1}$ и $t_{V_j 3}^0 = \bar{t}_{V_j 2}$.

г) Покажем, что задача о 3-разбиении имеет решение.

Если $|R_j| = 3$, $j = 1, \dots, n_0$, то в силу (2.1), (2.2), полагая $N_j^0 = R_j$, получаем решение задачи о 3-разбиении.

Пусть число множеств R_j таких, что $|R_j| = 3$, равно $k < n_0$. Тогда найдется ровно $(n_0 - k)/2$ множеств R_j таких, что $|R_j| = 2$, и столько же множеств R_j таких, что $|R_j| = 4$.

Рассмотрим произвольное множество R_j с $|R_j| = 2$. Пусть $R_j = \{i_1, i_2\}$. Тогда в силу (2.1), (2.2) имеем $e_{i_1} + e_{i_2} = E - 1$. Если E — четное число, то e_{i_1} и e_{i_2} должны быть не больше чем $E/2 - 1$. Но тогда их сумма не превосходит $E - 2$, что невозможно. Таким образом, E — нечетное число, причем $e_{i_1} = e_{i_2} = (E - 1)/2$, так как только в этом случае $e_{i_1} + e_{i_2} = E - 1$.

Рассмотрим произвольное множество R_j , $|R_j| = 4$. Пусть $R_j = \{p_1, p_2, p_3, p_4\}$. Из (2.1), (2.2) следует, что $\sum_{l=1}^4 e_{p_l} = E + 1$. Как уже отмечалось, E — нечетное число. Если $E \equiv 1 \pmod{4}$, то $\sum_{l=1}^4 e_{p_l} \geq 4(E + 3)/4 > E + 1$, что невозможно. Следовательно, $E \equiv 3 \pmod{4}$, причем $e_{p_l} = (E + 1)/4$, $l = 1, 2, 3, 4$ (поскольку только в этом случае $\sum_{l=1}^4 e_{p_l} = E + 1$).

Каждую пару множеств вида $\{i_1, i_2\}$ и $\{p_1, p_2, p_3, p_4\}$ преобразуем в пару множеств $\{i_1, p_1, p_2\}$ и $\{i_2, p_3, p_4\}$. Очевидно, $e_{i_1} + e_{p_1} + e_{p_2} = e_{i_2} + e_{p_3} + e_{p_4} = E$. Построенные множества вместе с теми из множеств R_j , для которых $|R_j| = 3$, образуют решение задачи о 3-разбиении. Таким образом, за $O(n_0)$ операций построено сведение задачи о 3-разбиении к рассматриваемой задаче распознавания, и следовательно, задача построения оптимального (по быстродействию) расписания обслуживания требований в системе, состоящей из трех и более приборов, является NP-трудной в сильном смысле.

2.2. Введем понятие расписания с прерываниями процесса обслуживания и установим основные свойства таких расписаний.

В систему, содержащую M последовательных приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Известны длительности $t_{iL} \geq 0$ обслуживания требования $i \in N$ прибором L , $L = 1, \dots, M$. Каждое требование обслуживается прибором 1, после завершения этого обслуживания — прибором 2 и т.д. до тех пор, пока оно не будет обслужено прибором M . По-прежнему предполагается, что каждый прибор одновременно обслуживает не более одного требования, а каждое требование одновременно обслуживается не более чем одним прибором.

В процессе обслуживания требований допускаются прерывания: каждое требование i может обслуживаться каждым прибором L в несколько "приемов" при сохранении условия, что суммарная длительность такого обслуживания равна t_{iL} . Процесс обслуживания требования i прибором L может быть прерван в любой момент времени и затем возобновлен. Число прерываний может быть произвольным, но конечным. После каждого прерывания прибор L может обслуживать любое требование, в то время как требование i не может обслуживаться прибором $L + 1$ до тех пор, пока полностью не завершится процесс его обслуживания прибором L .

Если $\bar{t}_{iL}$ — момент (полного) завершения обслуживания требования i прибором L при некотором расписании (с прерываниями) s , то, как и прежде, полагаем $\bar{t}_{\max}(s) = \max\{\bar{t}_{iL} \mid i \in N, L = 1, \dots, M\}$. Расписание s^* , минимизирующее значение $\bar{t}_{\max}(s)$, будем называть *оптимальным по быстродействию*.

Сформулируем и докажем некоторые свойства оптимальных (по быстродействию) расписаний с прерываниями.

1) Покажем, что *оптимальное расписание s^* можно искать в классе расписаний, при которых число прерываний не превосходит $Mn(n-1)$* .

Пусть s — произвольное расписание, и при этом расписании прибор Q , $1 \leq Q \leq M$, во временном интервале $(\tau, \tau']$ обслуживает требования $i_1, i_2, \dots, i_k$. Требование i_l обслуживается прибором Q в интервале $(t_l^0, \bar{t}_l]$, $l = 1, \dots, k$, где $\tau \leq t_1^0 < \bar{t}_1 \leq t_2^0 < \bar{t}_2 \leq \dots \leq t_k^0 < \bar{t}_k \leq \tau'$, причем обслуживание каждого из требований $i_1, i_2, \dots, i_k$ не завершается в интервале $(\tau, \tau']$.

Пусть прибор Q во временных интервалах $(t, \tau]$ и $(\tau', t']$ обслуживает требование $i \in N$.

Нетрудно видеть, что расписание s можно преобразовать в расписание s' , которое отличается от s только тем, что прибор Q обслуживает требование i во временном интервале $(t, \tau + \Delta]$, а требование i_l — во временном интервале $(t_l^0 + \Delta, \bar{t}_l + \Delta]$, $l = 1, \dots, k$, $\Delta = t' - \tau'$. При этом, очевидно, $\bar{t}_{\max}(s') = \bar{t}_{\max}(s)$.

Повторяя этот процесс исключения прерываний, приходим к расписанию $\tilde{s}$, при котором между моментами прерывания и последующего возобновления процесса обслуживания каждого требования каждым прибором завершается обслуживание какого-либо требования этим прибором,

т.е. процесс обслуживания каждого требования каждым прибором прерывается не более $n - 1$ раз, причем $\bar{t}_{\max}(\bar{s}) = \bar{t}_{\max}(s)$.

2) Покажем, что оптимальное расписание s^* можно искать в классе таких расписаний, при которых приборы 1 и M обслуживают требования без прерываний.

Действительно, пусть при некотором расписании s прибор 1 во временном интервале $(\tau, \tau']$ обслуживает (возможно, с прерыванием и не полностью) последовательность требований R , в которой не все требования, вообще говоря, различны, а во временных интервалах $(t, \tau]$ и $(\tau', t']$ — некоторое требование $i \in N$.

Нетрудно видеть, что расписание s можно преобразовать в расписание s' , которое отличается от s только тем, что последовательность требований R обслуживается прибором 1 во временном интервале $(t, t + \tau' - \tau]$, а требование i — в интервале $(t + \tau' - \tau, t']$. При этом $\bar{t}_{\max}(s') = \bar{t}_{\max}(s)$.

Повторяя этот процесс исключения прерываний, приходим к расписанию $\bar{s}$, при котором все требования обслуживаются прибором 1 без прерываний, и $\bar{t}_{\max}(\bar{s}) = \bar{t}_{\max}(s)$.

Исключение прерываний в обслуживании требований прибором M можно осуществить аналогичным образом.

Действительно, если при некотором расписании s прибор M обслуживает последовательность требований R в интервале $(\tau, \tau']$, а некоторое требование $i \in N$ — в интервалах $(t, \tau]$ и $(\tau', t']$, то можно преобразовать это расписание в расписание s' , отличающееся от s только тем, что требование i обслуживается прибором M в интервале $(t, t' - \tau' + \tau]$, а последовательность требований R — в интервале $(t' - \tau' + \tau, t']$, причем $\bar{t}_{\max}(s') = \bar{t}_{\max}(s)$.

Доказанное свойство означает, что при $M = 2$ оптимальное (по быстродействию) расписание с прерываниями можно искать в классе расписаний без прерываний.

2.3. Покажем, что и в том случае, когда в обслуживании каждого требования каждым прибором допускаются прерывания, задача построения оптимального (по быстродействию) расписания s^* при $M \geq 3$ является NP-трудной в сильном смысле.

Сформулируем соответствующую задачу распознавания: определить, существует ли такое расписание s^0 обслуживания (с прерываниями) множества $N = \{1, 2, \dots, n\}$ требований M приборами, при котором $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что при $M = 3$ к этой задаче сводится задача о 3-разбиении. Положим $n = 4n_0 + 2$, $N = \{1, 2, \dots, n\}$. Множество N требований разобьем на две группы: U -требования, обозначаемые U_i , $i = 1, \dots, 3n_0$, и V -требования, обозначаемые V_j , $j = 0, \dots, n_0 + 1$. Положим

$$t_{U_i 1} = t_{U_i 3} = e_i, \quad t_{U_i 2} = 0, \quad i = 1, \dots, 3n_0;$$

$$t_{V_0 1} = t_{V_0 2} = 0, \quad t_{V_0 3} = E, \quad t_{V_{n_0+1} 1} = E, \quad t_{V_{n_0+1} 2} = 2E, \quad t_{V_{n_0+1} 3} = 0;$$

$$t_{V_1 1} = 0, \quad t_{V_1 2} = 2E, \quad t_{V_1 3} = E, \quad t_{V_{n_0+1} 1} = E, \quad t_{V_{n_0+1} 2} = t_{V_{n_0+1} 3} = 0;$$

$$t_{V_j 1} = t_{V_j 3} = E, \quad t_{V_j 2} = 2E, \quad j = 2, \dots, n_0 - 1.$$

Покажем, что в построенной задаче расписание s^0 (с прерываниями), при котором $\bar{t}_{\max}(s^0) \leq y = 2n_0 E$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

1) Пусть задача о 3-разбиении имеет решение и N_j^0 , $j = 1, \dots, n_0$ — найденные подмножества множества N^0 . Пусть $\pi_j(U)$ — произвольная перестановка U -требуваний с номерами из множества N_j^0 , $j = 1, \dots, n_0$.

Расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$ и все требования обслуживаются без прерываний, существует и может быть построено следующим образом. Все три прибора функционируют без простоев во временном интервале $(0, y]$, причем прибор 1 обслуживает требования в последовательности $(\pi_1(U), V_2, \pi_2(U), V_3, \dots, V_{n_0-1}, \pi_{n_0-1}(U), V_{n_0}, \pi_{n_0}(U), V_{n_0+1})$, прибор 2 — в последовательности $(V_1, V_2, \dots, V_{n_0})$ и прибор 3 — в последовательности $(V_0, \pi_1(U), V_1, \pi_2(U), V_2, \dots, V_{n_0-2}, \pi_{n_0-1}(U), V_{n_0-1}, \pi_{n_0}(U))$.

2) Пусть существует расписание s^0 . Покажем, что имеет решение задача о 3-разбиении. Как уже отмечалось, не нарушая общности рассуждений, можно считать, что приборы 1 и 3 обслуживают все требования без прерываний. Так как суммарная длительность обслуживания всех требований каждым прибором равна y , то при расписании s^0 все три прибора функционируют без простоев во временном интервале $(0, y]$.

Рассмотрим временной интервал $(0, 2E]$. Поскольку прибор 3 функционирует без простоев и обслуживает требования без прерываний, то требование V_0 должно обслуживаться этим прибором в интервале $(0, E]$. Поскольку требование V_1 — единственное требование, которое может обслуживаться прибором 2, начиная с момента времени 0, то при расписании s^0 имеем $t_{V_1, 2}^0 = 0$ и $\bar{t}_{V_1, 2} \geq 2E$.

Пусть суммарная длительность обслуживания U -требуваний прибором 1 в интервале $(0, 2E]$ меньше E . Тогда имеем простой в функционировании прибора 3, поскольку эти U -требувания будут обслужены прибором 3 раньше момента времени $2E$, в то время как $t_{V_1, 3}^0 \geq \bar{t}_{V_1, 2} \geq 2E$.

Пусть суммарная длительность обслуживания U -требуваний прибором 1 в интервале $(0, 2E]$ больше E . Тогда в интервале $(0, 2E]$ прибор 1 не завершит обслуживание ни одного из V -требуваний и, следовательно, прибор 2 в интервале $(2E, 4E]$ будет простаивать, что невозможно.

Следовательно, прибор 1 в интервале $(0, 2E]$ обслуживает некоторые U -требувания в течение E единиц времени. Если при этом обслуживание хотя бы одного из этих требований прибором 1 завершается в момент времени $2E$, то это требование не может быть обслужено прибором 3 в интервале $(0, 2E]$, т.е. возникает простой прибора 3, что невозможно.

Итак, прибор 1 обслуживает U -требувания в интервале $(0, E]$, а в интервале $(E, 2E]$ он обслуживает одно из V -требуваний. Этим требованием не может быть ни V_{n_0} , ни V_{n_0+1} , так как в противном случае возникает простой прибора 3 либо прибора 2 соответственно. Не нарушая общности рассуждений, можно считать, что прибор 1 обслуживает в интервале $(E, 2E]$ требование V_2 .

Таким образом, прибор 1 обслуживает в интервале $(0, E]$ некоторые U -требования (их будет ровно три), а в интервале $(E, 2E]$ — требования V_2 ; прибор 2 обслуживает требования V_1 без прерываний в интервале $(0, 2E]$ (так как появление прерываний в процессе обслуживания этого требования привело бы к возникновению простоя прибора 2); наконец, прибор 3 обслуживает в интервале $(0, E]$ требование V_0 , а в интервале $(E, 2E]$ — те U -требования, которые были обслужены прибором 1 в интервале $(0, E]$.

Рассуждая аналогичным образом, приходим к заключению, что при расписании s^0 прибор 1 в интервале $(2(j-1)E, (2j-1)E]$ обслуживает три U -требования, а в интервале $((2j-1)E, 2jE]$ — требование V_{j+1} ; прибор 2 в интервале $(2(j-1)E, 2jE]$ обслуживает без прерываний требование V_j ; прибор 3 в интервале $(2(j-1)E, (2j-1)E]$ обслуживает требование V_{j-1} , а в интервале $((2j-1)E, 2jE]$ — те из U -требований, которые обслужены прибором 1 в интервале $(2(j-1)E, (2j-1)E]$, $j = 2, \dots, n_0$. Обозначая через N_j^0 множество номеров U -требований, обслуживаемых прибором 1 во временном интервале $(2(j-1)E, (2j-1)E]$, получаем решение задачи о 3-разбиении. Таким образом, за $O(n_0)$ операций построено сведение задачи о 3-разбиении к рассматриваемой задаче.

2.4. В заключение этого параграфа покажем, что *задача построения оптимального (по быстройдействию) расписания при $M = 3$ остается NP-трудной, если каждое требование множества $N = \{1, 2, \dots, n\}$ обслуживается только двумя из трех имеющихся приборов.*

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 (с прерываниями), при котором $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче сводится задача о разбиении (см. вводную часть этой главы).

Положим $n = n_0 + 2$, $N = \{1, 2, \dots, n\}$; $t_{i1} = t_{i3} = e_i$, $t_{i2} = 0$, $i \in N^0$; $t_{n_0+1,1} = E$, $t_{n_0+1,2} = 2E$, $t_{n_0+1,3} = 0$; $t_{n_0+2,1} = 0$, $t_{n_0+2,2} = 2E$, $t_{n_0+2,3} = E$. Покажем, что задача о разбиении имеет решение тогда и только тогда, когда в построенной задаче существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = 4E$.

1) Пусть задача о разбиении имеет решение и N_1^0, N_2^0 — найденные подмножества. Тогда расписание s^0 существует и может быть построено следующим образом. Прибор 1 во временном интервале $(0, 3E]$ обслуживает требования множества $N \setminus \{n_0 + 2\}$ непрерывно одно за другим в последовательности $(\pi_{N_1^0}, n_0 + 1, \pi_{N_2^0})$, где $\pi_{N_1^0}, \pi_{N_2^0}$ — произвольные перестановки элементов множеств N_1^0 и N_2^0 соответственно. Прибор 2 во временном интервале $(0, 4E]$ обслуживает требования $n_0 + 1$ и $n_0 + 2$ в последовательности $(n_0 + 2, n_0 + 1)$ непрерывно одно за другим. Прибор 3 во временном интервале $(E, 4E]$ обслуживает требования множества $N \setminus \{n_0 + 1\}$ в последовательности $(\pi_{N_1^0}, n_0 + 2, \pi_{N_2^0})$ непрерывно одно за другим.

2) Пусть существует расписание s^0 . Если $\bar{t}_{n_0+2,2} = 4E$, то $\bar{t}_{n_0+2,3} > 4E$ и $\bar{t}_{\max}(s^0) > y$. Таким образом, $\bar{t}_{n_0+1,2} = 4E$, следовательно, $\bar{t}_{n_0+1,1} \leq 2E$. Кроме того, $\bar{t}_{n_0+2,2} \geq 2E$ и $t_{n_0+2,3}^0 \geq 2E$.

Предположим, что задача о разбиении не имеет решения. Пусть $R \subset \{1, 2, \dots, n_0\}$ – множество требований, обслуживание которых прибором 1 начато и закончено в интервале $(0, 2E]$. Так как требование $n_0 + 1$ должно быть обслужено прибором 1 в этом интервале и задача о разбиении не имеет решения, то $\sum_{i \in R} t_{i1} < E$. Но тогда $\sum_{i \in N \setminus R} t_{i3} > 2E$ и так как

обслуживание каждого из требований множества $N \setminus R$ прибором 3 не может быть начато ранее момента времени $2E$, то $\bar{t}_{\max}(s^0) > 4E$, что невозможно. Это означает, что задача о разбиении должна иметь решение.

Таким образом, за $O(n_0)$ операций построено сведение задачи о разбиении к рассматриваемой задаче распознавания.

§ 3. Общее время обслуживания требований без задержек

В этом параграфе рассматривается задача построения оптимального по быстродействию расписания без задержек в процессе обслуживания каждого требования.

3.1. В систему поточного типа, состоящую из M приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L, L = 1, \dots, M$, предполагаются заданными. Каждое требование i обслуживается *без задержек*, т.е. для любых двух приборов L_1 и $L_2, L_1 < L_2$, таких, что $t_{iL_1} > 0, t_{iL_2} > 0$ и либо $L_2 = L_1 + 1$, либо $t_{iL} = 0, L = L_1 + 1, \dots, L_2 - 1$, должно выполняться соотношение $t_{iL_2}^0 = \bar{t}_{iL_1}$, причем процесс обслуживания каждого требования каждым прибором протекает без прерываний. Другими словами, если требование i начнет обслуживаться в момент времени t_i^0 , то процесс обслуживания этого требования завершается в момент времени $\bar{t}_i = t_i^0 + \sum_{L=1}^M t_{iL}$.

3.2. В теории расписаний нулевые длительности обслуживания имеют две различные интерпретации. При первой из них запись $t_{iL} = 0$ содержательно означает, что требование i прибором L не обслуживается. При второй интерпретации запись $t_{iL} = 0$ содержательно означает, что требование i прибором L обслуживается, но длительность обслуживания настолько мала, что ею можно пренебречь и считать, что $\bar{t}_{iL} = t_{iL}^0$.

В зависимости от принятой интерпретации нулевых длительностей обслуживания могут быть построены различные оптимальные расписания с различными значениями оптимизируемого функционала.

Рассмотрим, например, задачу построения оптимального (по быстродействию) расписания обслуживания без задержек четырех требований тремя приборами. Длительности обслуживания t_{iL} приведены в табл. 3.1.

Таблица 3.1

$i \backslash L$				$i \backslash L$			
	1	2	3		1	2	3
1	1	6	3	3	4	1	3
2	2	0	4	4	2	3	1

Если запись $t_{iL} = 0$ понимать таким образом, что требование 2 прибором 2 не обслуживается, то оптимальным является расписание s_1^* , изображенное на рис. 3.1, а. Если же считать, что требование 2 прибором 2 обслуживается, но длительность этого обслуживания мала и ею можно пренебречь, то оптимальным будет расписание s_2^* , изображенное на рис. 3.2, б. Отметим, что $\bar{t}_{\max}(s_1^*) = 15 < \bar{t}_{\max}(s_2^*) = 17$.

3.3. Пусть запись $t_{iL} = 0$ означает, что требование $i \in N = \{1, 2, \dots, n\}$ прибором L , $1 \leq L \leq M$, не обслуживается. Покажем, что в этом случае задача построения оптимального (по быстрдействию) расписания обслуживания без задержек является NP-трудной в сильном смысле при $M \geq 2$.

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче при $M = 2$ сводится задача о 3-разбиении (см. вводную часть этой главы).

Положим $n = 4n_0 + 2$, $N = \{1, 2, \dots, n\}$, множество N требований разобьем на две группы: U -требования, обозначаемые U_i , $i = 1, \dots, 3n_0$, и V -требования, обозначаемые V_j , $j = 0, \dots, n_0 + 1$. Положим

$$t_{U_i 1} = 0, \quad t_{U_i 2} = e_i, \quad i = 1, \dots, 3n_0,$$

$$t_{V_0 1} = 0, \quad t_{V_0 2} = 2E,$$

$$t_{V_j 1} = 2E, \quad t_{V_j 2} = E, \quad j = 1, \dots, n_0 + 1.$$

Покажем, что в построенной задаче расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = (2n_0 + 3)E$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

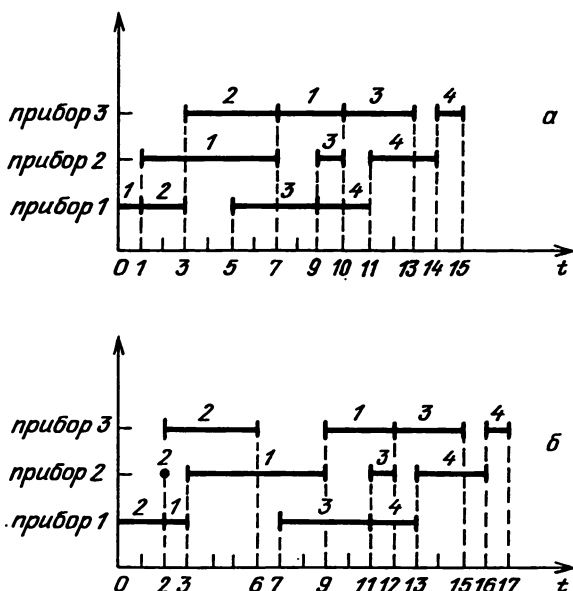


Рис. 3.1

Пусть задача о 3-разбиении имеет решение и N_j^0 , $j = 1, \dots, n_0$, — найденные трехэлементные подмножества элементов множества N^0 . Через π_j обозначим произвольную перестановку U -требований с номерами из множества N_j^0 . Расписание s^0 существует и может быть построено следующим образом. Прибор 1 обслуживает требование V_j во временном интервале $(2(j-1)E, 2jE]$, $j = 1, \dots, n_0 + 1$. Прибор 2 в интервале $(0, 2E]$ обслуживает требование V_0 , в интервале $((2n_0 + 2)E, (2n_0 + 3)E]$ — требование V_{n_0+1} и в каждом из интервалов $(2jE, 2(j+1)E]$ — последовательность требований (V_j, π_j) непрерывно одно за другим, $j = 1, \dots, n_0$.

Пусть существует расписание s^0 . Поскольку $\sum_{i \in N} t_{iL} = y$, то прибор 2 при расписании s^0 функционирует без простоев во временном интервале $(0, y]$ и $\bar{t}_{\max}(s^0) = y$. Так как прибор 1 обслуживает только требования V_j , $j = 1, \dots, n_0 + 1$, то этот прибор при расписании s^0 функционирует без простоев во временном интервале $(0, y - E]$. Требования V_j не отличаются друг от друга по длительностям обслуживания, поэтому можно считать, что эти требования обслуживаются прибором 1 в порядке возрастания их номеров. Поскольку при расписании s^0 требования V_j обслуживаются без задержек, то прибор 2 обслуживает требование V_j в интервале $(2jE, (2j+1)E]$, $j = 1, \dots, n_0 + 1$. При этом требование V_0 обслуживается прибором 2 в интервале $(0, 2E]$ (поскольку это единственный не занятый подынтервал длины $2E$ интервала $(0, y]$). Требования U_i должны обслуживаться прибором 2 в интервалах $((2j+1)E, 2(j+1)E]$, $j = 1, \dots, n_0$. Обозначая через N_j^0 множество номеров U -требований, обслуживаемых прибором 2 в интервале $((2j+1)E, 2(j+1)E]$, $j = 1, \dots, n_0$, получаем решение задачи о 3-разбиении.

3.4. Здесь и далее в этом параграфе запись $t_{iL} = 0$ понимается таким образом, что требование i прибором L обслуживается, но длительность этого обслуживания мала и ею можно пренебречь.

Установим некоторые свойства расписаний обслуживания требований без задержек.

1) Покажем, что если все $t_{iL} > 0$, $i \in N$, $L = 1, \dots, M$, то при любом расписании обслуживания без задержек все приборы обслуживают требования множества N в одной и той же последовательности.

Действительно, если бы прибор $L + 1$ обслуживал требования в некоторой последовательности $\pi' = (\dots, j, i, \dots)$, а прибор L — в последовательности $\pi' = (\dots, i, \dots, j, \dots)$, то имело бы место соотношение $t_{iL+1}^0 \geq \bar{t}_{iL} + t_{jL} + t_{jL+1} > \bar{t}_{iL}$, что невозможно.

Если $t_{iL} \geq 0$, $i \in N$, $L = 1, \dots, M$, то нетрудно видеть, что при указанной интерпретации записи $t_{iL} = 0$ оптимальное расписание также можно искать в классе расписаний, при которых все приборы обслуживают требования в одной и той же последовательности.

2) Пусть $t_{iL} \geq 0$ и все приборы обслуживают требования в одной и той же последовательности $\pi = (i_1, i_2, \dots, i_n)$. Поскольку все требования должны обслуживаться без задержек, то минимальное общее время обслуживания

$$\bar{t}_{\max}(\pi) = \bar{t}_{inM} = t_{in1}^0 + \sum_{L=1}^M t_{inL}.$$

Положим $t_{i_1,1}^0 = 0$ (так как при $t_{i_1,1}^0 > 0$ общее время обслуживания только увеличится). Выберем произвольное $k, 2 \leq k \leq n$. Очевидно,

$$t_{ik,1}^0 \geq t_{ik-1,1}^0 + t_{ik-1,1};$$

$$t_{ik,1}^0 + \sum_{L=1}^{Q-1} t_{ik,L} \geq t_{ik-1,1}^0 + \sum_{L=1}^Q t_{ik-1,L}, \quad Q=2, \dots, M.$$

Другими словами,

$$t_{ik,1}^0 - t_{ik-1,1}^0 \geq \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{ik-1,L} - \sum_{L=1}^{Q-1} t_{ik,L} \right\}. \quad (3.1)$$

В качестве наиболее раннего момента времени $t_{ik,1}^0$ выберем такой момент времени, при котором в (3.1) имеет место равенство.

Таким образом, приходим к рекуррентному соотношению

$$t_{i_1,1}^0 = 0,$$

$$t_{ik,1}^0 - t_{ik-1,1}^0 = \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{ik-1,L} - \sum_{L=1}^{Q-1} t_{ik,L} \right\}, \quad k=2, \dots, n,$$

из которого следует, что

$$t_{i_n,1}^0 = \sum_{k=1}^{n-1} \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{ik,L} - \sum_{L=1}^{Q-1} t_{ik+1,L} \right\},$$

и следовательно,

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{ik,L} - \sum_{L=1}^{Q-1} t_{ik+1,L} \right\} + \sum_{L=1}^M t_{i_n,L}. \quad (3.2)$$

Таким образом, задача построения оптимального (по быстродействию) расписания обслуживания без задержек сводится к задаче нахождения перестановки π^* , которой соответствует наименьшее значение функционала (3.2).

3) Задача минимизации $\bar{t}_{\max}(\pi)$ очевидным образом сводится к задаче о коммивояжере с $n+1$ городами, пронумерованными целыми числами от 0 до n , и матрицей расстояний $\|c_{pq}\|$, где $c_{pp} = +\infty$ и $c_{pq} \geq 0$, $p=0, \dots, n$, $q=0, \dots, n$, $p \neq q$. Пусть $\pi_0 = (0, i_1, i_2, \dots, i_n)$ — некоторая перестановка номеров городов. Задача о коммивояжере заключается в поиске перестановки π_0^* , минимизирующей величину

$$C(\pi_0) = c_{0i_1} + \sum_{k=1}^{n-1} c_{ik} t_{k+1} + c_{i_n 0}. \quad (3.3)$$

Если положить $c_{0p} = 0$, $c_{p0} = \sum_{L=1}^M t_{pL}$, $c_{pq} = \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \right\}$, $p=1, \dots, n$, $q=1, \dots, n$, $p \neq q$, то получаем функционал (3.2).

Тем самым, если $\pi_0^* = (0, i_1^*, i_2^*, \dots, i_n^*)$ — решение задачи о коммивояжере, то перестановка $\pi^* = (i_1^*, i_2^*, \dots, i_n^*)$ определяет оптимальное расписание.

В общем случае задача о коммивояжере является NP -трудной в сильном смысле. Тем не менее при некоторых предположениях относительно вида матрицы $\|c_{pq}\|$ эта задача может быть решена за полиномиальное число операций. Один из таких разрешимых случаев задачи о коммивояжере используется ниже для нахождения перестановки π^* при $M = 2$.

3.5. Пусть каждое требование множества $N = \{1, 2, \dots, n\}$ должно быть обслужено без задержек в системе поточного типа, состоящей из двух приборов. Опишем алгоритм построения оптимального (по быстрдействию) расписания из класса расписаний, при которых оба прибора обслуживают требования в одной и той же последовательности.

Обслуживающие приборы обозначим через A и B , а длительности обслуживания требования $i \in N$ приборами A и B — через a_i и b_i соответственно. Каждое требование обслуживается сначала прибором A , затем прибором B . Если оба прибора обслуживают требования в некоторой последовательности $\pi = (i_1, i_2, \dots, i_n)$, то наименьшее возможное значение общего времени обслуживания

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} \max \{a_{i_k}, a_{i_k} + b_{i_k} - a_{i_{k+1}}\} + a_{i_n} + b_{i_n}.$$

Это соотношение можно переписать в виде

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} \max \{b_{i_k} - a_{i_{k+1}}, 0\} + \sum_{k=1}^n a_{i_k} + b_{i_n}. \quad (3.4)$$

Очевидно, для построения оптимального расписания достаточно найти перестановку π^* элементов множества N , которой соответствует наименьшее значение

$$g(\pi) = \sum_{k=1}^{n-1} \max \{b_{i_k} - a_{i_{k+1}}, 0\} + b_{i_n}. \quad (3.5)$$

Если положить $c_{0p} = 0$, $c_{p0} = b_p$, $c_{pq} = \max \{b_p - a_q, 0\}$, $p = 1, \dots, n$, $q = 1, \dots, n$, $p \neq q$, то задача минимизации функционала (3.5) сводится к задаче о коммивояжере с $n + 1$ городами, пронумерованными целыми числами от 0 до n , и матрицей расстояний $\|c_{pq}\|$ (см. п. 3.4, 3)). При этом если $\pi_0^* = (0, i_1^*, i_2^*, \dots, i_n^*)$ — решение задачи о коммивояжере, то $\pi^* = (i_1^*, i_2^*, \dots, i_n^*)$.

Положим $\alpha_0 = \beta_0 = 0$, $\alpha_i = a_i$, $\beta_i = b_i$, $i \in N$. Введем в рассмотрение функции $u(x) \equiv 0$ и $v(x) \equiv 1$. Тогда

$$c_{pq} = \begin{cases} \int_{\beta_p}^{\alpha_q} u(x) dx, & \alpha_q \geq \beta_p, \\ \int_{\alpha_q}^{\beta_p} v(x) dx, & \alpha_q < \beta_p. \end{cases} \quad (3.6)$$

Опишем полиномиальный алгоритм решения задачи о коммивояжере в предположении, что расстояния c_{pq} между городами вычисляются по формуле (3.6), где $u(x)$ и $v(x)$ — произвольные интегрируемые функции, удовлетворяющие условию $u(x) + v(x) \geq 0$, а α_q и β_p , $0 \leq p, q \leq n$, — произвольные неотрицательные числа.

Перенумеруем города целыми числами от 1 до $m = n + 1$ таким образом, чтобы $\beta_1 \leq \beta_2 \leq \dots \leq \beta_m$. Отыщем перестановку $\varphi = (r_1, r_2, \dots, r_m)$ такую, что $\alpha_{r_1} \leq \alpha_{r_2} \leq \dots \leq \alpha_{r_m}$. Обозначим $\gamma_i = \min\{\beta_i, \alpha_{r_i}\}$, $\delta_i = \max\{\beta_i, \alpha_{r_i}\}$, $i = 1, \dots, m$. Положим $c_\varphi(i) = \int_{\delta_i}^{\gamma_{i+1}} (u(x) + v(x)) dx$, если $\delta_i < \gamma_{i+1}$, и $c_\varphi(i) = 0$ в противном случае.

Построим неориентированный граф G_φ на m вершинах, пронумерованных целыми числами от 1 до m , в котором вершина i смежна с вершиной j тогда и только тогда, когда $i = r_j$, $i, j = 1, \dots, m$. Если граф G_φ связан, то перестановка φ является решением рассматриваемой задачи о коммивояжере (относительно новой нумерации городов). В противном случае построим граф G'_φ , заменив каждую компоненту связности графа G_φ новой вершиной. Если вершины i и $i + 1$ принадлежали разным компонентам связности графа G_φ , то вершины графа G'_φ , соответствующие данным компонентам, соединим ребром R_i , приписав ему вес $c_\varphi(i)$, $i = 1, \dots, m - 1$. Полученный в результате (мульти)граф по-прежнему будем обозначать через G'_φ . Отыщем в графе G'_φ остовное дерево с наименьшей суммой весов ребер (минимальное остовное дерево).

Ребра, вошедшие в остовное дерево, разобьем на две группы. К первой группе отнесем такие ребра R_i , что $\alpha_{r_i} < \beta_i$, а ко второй — все остальные.

Пусть число ребер в первой группе равно $l \geq 1$. Положим j_1 равным наибольшему значению i такому, что ребро R_i принадлежит этой группе. Рассматривая оставшиеся ребра первой группы, аналогичным образом найдем значение j_2 и т.д., пока не будет найдено значение j_l .

Пусть число ребер во второй группе равно $h \geq 1$. Положим j_{l+1} равным наименьшему значению i такому, что ребро R_i принадлежит этой группе. Рассматривая оставшиеся ребра второй группы, аналогичным образом найдем j_{l+2} и т.д., пока не будет найдено значение j_{l+h} .

Рассмотрим перестановку $(1, 2, \dots, m)$. Поменяем местами элементы j_1 и $j_1 + 1$. В полученной перестановке поменяем местами элементы j_2 и $j_2 + 1$ и т.д. до тех пор, пока не поменяются местами элементы j_{l+h} и $j_{l+h} + 1$. В результирующей перестановке элемент i заменим на r_i , $i = 1, \dots, m$, в соответствии с перестановкой φ . Обозначим полученную перестановку через $\tilde{\pi} = (i_1, i_2, \dots, i_m)$. Построим перестановку $\tilde{\pi}^* = (1, j_1^*, j_2^*, \dots, j_n^*)$, полагая $j_1^* = i_1$, $j_k^* = i_{j_{k-1}^*}$, $k = 2, \dots, n$. Возвращаясь к исходной нумерации городов, получаем искомую перестановку π_0^* — решение задачи о коммивояжере.

Наиболее трудоемким этапом нахождения π_0^* является нахождение минимального остовного дерева в графе G'_φ . Известно, что эта задача может быть решена не более чем за $O(e \log v)$ операций, где e и v — число вершин и ребер графа, в котором ищется остовное дерево. Поскольку для

i	1	2	3	4	5
a_i	1	1	2	4	5
b_i	4	3	4	1	1

Таблица 3.3

i	α_i	β_i	r_i	$c_\varphi(i)$
1	0	0	1	1
2	5	1	4	0
3	4	1	5	1
4	1	3	6	1
5	1	4	3	0
6	2	4	2	—

графа G'_φ значения $e \leq n$ и $v \leq n+1$, то решение задачи о коммивояжере в рассматриваемом случае может быть получено за $O(n \log n)$ операций.

Таким образом, оптимальное по быстродействию расписание без задержек при $M=2$ может быть построено за $O(n \log n)$ операций.

3.6. Пример. Две бригады A и B должны выполнить монтаж пяти единиц оборудования. Бригада A заливает фундаменты, а бригада B сразу после подготовки очередного фундамента размещает соответствующую единицу оборудования. Длительность подготовки фундамента под i -ую единицу оборудования обозначим через a_i , а длительность размещения этой единицы на подготовленном фундаменте — через b_i , $i = 1, 2, 3, 4, 5$. Значения a_i и b_i приведены в табл. 3.2.

Необходимо таким образом организовать работу бригад, чтобы все единицы оборудования были установлены как можно раньше.

Если бригады рассматривать как приборы, а единицы оборудования — как требования, то получим ситуацию, описанную в п. 3.5.

Положим $\alpha_0 = \beta_0 = 0$, $\alpha_i = a_i$, $\beta_i = b_i$, $i = 1, 2, 3, 4, 5$; $u(x) \equiv 0$, $v(x) \equiv 1$. Следуя алгоритму, произведем перенумерацию, найдем перестановку $\varphi = (r_1, r_2, r_3, r_4, r_5, r_6)$ и вычислим значения $c_\varphi(i)$ (см. табл. 3.3).

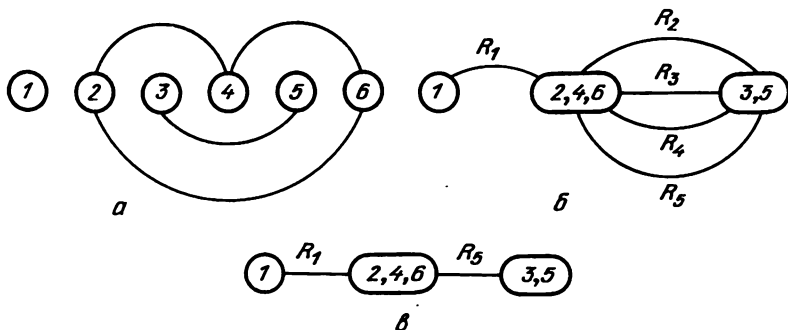


Рис. 3.2

Построим графы G_φ и G'_φ (рис. 3.2, а, б). Минимальное остовное дерево графа G'_φ приведено на рис. 3.2, в и содержит два ребра — R_1 и R_5 . Эти ребра принадлежат второй группе.

Меняя местами в перестановке (1, 2, 3, 4, 5, 6) элементы 1 и 2, а в полученной перестановке — элементы 5 и 6, получаем перестановку (2, 1, 3, 4, 6, 5). Если в этой перестановке элемент, равный i , заменить на элемент, равный r_i , $i = 1, \dots, 6$, приходим к перестановке $\tilde{\pi} = (4, 1, 5, 6, 2, 3)$.

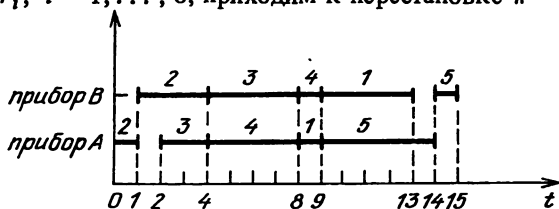


Рис. 3.3

Построим перестановку $\tilde{\pi}^* = (1, 4, 6, 3, 5, 2)$. Возвращаясь к исходной нумерации, получаем перестановку $\pi_0^* = (0, 2, 3, 4, 1, 5)$. Следовательно, при оптимальном расписании оба прибора обслуживают требования в последовательности (2, 3, 4, 1, 5) (рис. 3.3).

3.7. Покажем, что задача построения оптимального (по быстродействию) расписания обслуживания без задержек является NP -трудной в сильном смысле, если число приборов $M \geq 3$.

По-прежнему предполагается, что запись $t_{iL} = 0$ означает, что требование i прибором L обслуживается, но длительностью этого обслуживания можно пренебречь. Оптимальное расписание ищется в классе расписаний, при которых требования обслуживаются всеми приборами в одной и той же последовательности.

Доказательство проведем в несколько этапов.

1) Покажем, что задача нахождения гамильтонова контура минимальной длины в ориентированном графе является NP -трудной в сильном смысле, если длины его дуг принимают значения 0 или 1.

Сформулируем соответствующую задачу распознавания. Задан (ориентированный) граф $G = (V, U)$, длины дуг которого равны 0 или 1. Требуется определить, существует ли в графе гамильтонов контур, длина которого не превосходит заданного числа y .

Покажем, что к этой задаче сводится задача о трехмерном паросочетании (см. вводную часть этой главы).

Пронумеруем тройки из множества E произвольным образом: $e_1, e_2, \dots, e_{|E|}$. Положим $N^0 = \{1, 2, \dots, |E|\}$ и зададим на N^0 произвольную циклическую подстановку φ . Напомним, что подстановкой называется взаимно однозначное отображение конечного множества на себя. Подстановка называется циклической, если при повторении ее достаточное число раз каждый из элементов может быть отображен в любой другой элемент. Пусть $e_r = (x_{r1}, x_{r2}, x_{r3})$, $r \in N^0$. Для каждого элемента $x \in X_p$, $p = 1, 2, 3$, положим $N_x^0 = \{r \in N^0 | x_{rp} = x\}$ и зададим на N_x^0 произвольную циклическую подстановку φ_x . Отметим, что все множества N_x^0 , $x \in X_p$, $p = 1, 2, 3$, не пусты.

Построим граф $G = (V, U)$ следующим образом. Каждой тройке $e_r, r \in N^0$, поставим в соответствие множество V_r , состоящее из восьми вершин. Эти вершины будем обозначать через $[r, j], j = 1, \dots, 8$. Положим $V = \bigcup_{r \in N^0} V_r$. Множество U дуг графа G определим как объединение всех перечисленных ниже множеств:

$$H_\varphi = \{([r, 8], [\varphi(r), 1]) \mid r \in N^0\};$$

$$H_{\varphi_x} = \{([r, 2p], [\varphi_x(r), 2p+1]) \mid r \in N_x^0, x \in X_p, p = 1, 2, 3\};$$

$$F_r = \{([r, 2q-1], [r, 2q]) \mid q = 1, 2, 3, 4\} \cup$$

$$\cup \{([r, 2p+1], [r, 2p]) \mid p = 1, 2, 3\} \cup$$

$$\cup \{([r, 2p-1], [r, 8]) \mid p = 1, 2, 3, r \in N^0\}.$$

Положим длину каждой дуги $([r, 1], [r, 2]), r \in N^0$, равной единице, а длины всех остальных дуг равными нулю.

По построению множество вершин графа G состоит из $8|E|$ вершин вида $[r, j], r \in N^0, j = 1, \dots, 8$. Вершины вида $[r, 2q], r \in N^0, q = 1, 2, 3, 4$, назовем четными (полустепень исхода каждой из этих вершин равна единице). Вершины вида $[r, 2q-1], r \in N^0, q = 1, 2, 3, 4$ назовем нечетными (полустепень захода каждой из этих вершин равна единице).

На рис. 3.4 изображен подграф G_r графа G , порожденный множеством вершин V_r , а также изображены вершины, смежные с вершинами множества V_r (нечетные вершины для наглядности заштрихованы). Множество U дуг графа содержит дуги двух типов. Во-первых, U содержит все дуги множеств H_φ и $H_{\varphi_x}, x \in X_p, p = 1, 2, 3$. Эти дуги будем называть H -дугами. Каждая из них исходит из четной вершины некоторого множества V_r , и заходит в одну из нечетных вершин некоторого другого множества $V_{r'}$ (на рис. 3.4 H -дуги изображены сплошными линиями). Остальные дуги множества U (т.е. все дуги множеств $F_r, r \in N^0$) будем называть F -дугами. Каждая из этих дуг исходит из нечетной вершины некоторого множества V_r и заходит в некоторую четную вершину этого же множества (на рис. 3.4 F -дуги изображены пунктирными линиями.)

Нетрудно проверить, что для каждого $x \in X_p, p = 1, 2, 3$, F -дуги вида $([r, 2p+1], [r, 2p]), r \in N_x^0$, и все H -дуги множества H_{φ_x} образуют кон-

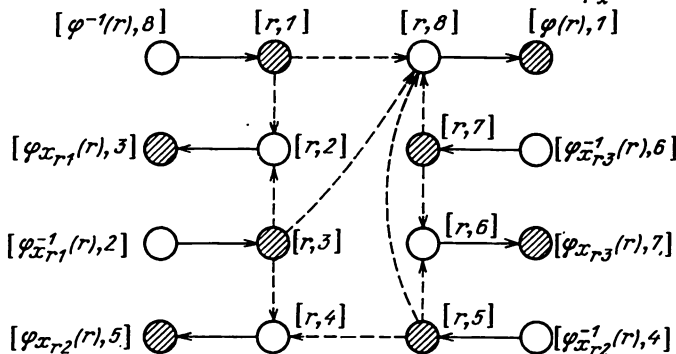


Рис. 3.4

тур K_x . В этом контуре F - и H -дуги чередуются, и он проходит только через вершины $[r, 2p+1]$ и $[r, 2p]$, принадлежащие всем таким множествам V_r , что элемент x встречается в p -ой компоненте соответствующей тройки e_r . Заметим также, что F -дуги вида $([r, 1], [r, 8])$, $r \in N^0$, и все H -дуги из множества $H_\varnothing$ образуют контур K . В этом контуре F - и H -дуги чередуются, и он проходит только через вершины $[r, 1]$ и $[r, 8]$, $r \in N^0$.

Выделим в графе G один или несколько контуров так, чтобы каждая вершина графа принадлежала ровно одному контуру. При движении по этим контурам для каждого из множеств V_r , $r \in N^0$, возможен один из следующих способов обхода его вершин. Обход, схематически представленный на рис. 3.5, а, назовем нулевым. При этом обходе проходятся только дуги контуров K и K_{xrp} , $p = 1, 2, 3$. Суммарная длина дуг, инцидентных вершинам множества V_r , в этом случае равна нулю. Три других способа обхода представлены на рис. 3.5, б, в, г. Обходы такого вида назовем μ -присоединенными, $1 \leq \mu \leq 3$. При каждом из этих обходов суммарная длина дуг, инцидентных вершинам множества V_r , равна единице. Название обхода отражает тот факт, что между "посещениями" вершины $[r, 1]$ и вершины $[r, 8]$ к дугам контура K "присоединяются" дуги контуров K_{xrp} , $p = 1, \dots, \mu$, $1 \leq \mu \leq 3$.

Покажем, что в построенном графе G гамильтонов контур, длина которого не превосходит $y = n_0$, существует тогда и только тогда, когда имеет решение задача о трехмерном паросочетании.

а) Пусть существует паросочетание $E' \subseteq E$, $|E'| = n_0$ и $N' = \{r \in N^0 \mid e_r \in E'\}$. Рассмотрим $3n_0 + 1$ контуров K, K_x , $x \in X_p$, $p = 1, 2, 3$.

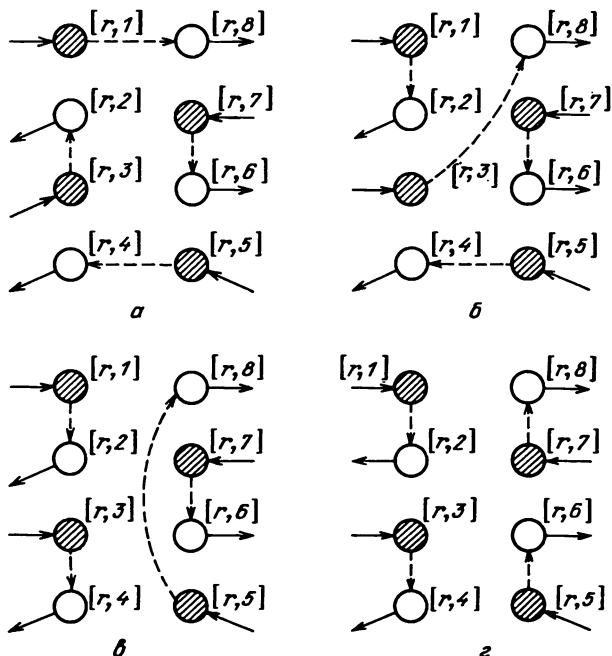


Рис. 3.5

Очевидно, каждая вершина множества V принадлежит равно одному из указанных контуров. В данном наборе контуров содержатся все H -дуги и обходы каждого множества вершин V_r являются нулевыми (см. рис. 3.5, а).

Для каждой тройки $e_r \in E'$ заменим нулевой обход вершин множества V_r на 3-присоединенный. Так как E' — трехмерное парасочетание, то для осуществления такой замены необходимо из каждого контура $K_{x_r p}$, $1 \leq p \leq 3$, удалить ровно одну F -дугу $([r, 2p + 1], [r, 2p])$ и присоединить пути, образовавшиеся из контуров $K_{x_r 1}$, $K_{x_r 2}$, $K_{x_r 3}$ в результате такого удаления, к дугам контура K с удаленными F -дугами $([r, 1], [r, 8])$, $r \in N'$. Получаем гамильтонов контур, в котором ровно n_0 дуг имеют длину, равную единице.

б) Пусть в графе G существует гамильтонов контур, длина которого не превосходит $y = n_0$. Следовательно, при движении по этому контуру обходы вершин по крайней мере $|N^0|$ — у множеств V_r должны быть нулевыми, а обходы вершин не более y множеств V_r должны быть μ -присоединенными, $1 \leq \mu \leq 3$.

Покажем, что для любого из $3n_0$ элементов $x \in X_p$, $p = 1, 2, 3$, найдется по крайней мере одно множество V_r , $r \in N_x^0$, обход вершин которого не является нулевым. Предположим противное. Пусть для некоторого $\bar{x} \in X_p$, $p = 1, 2, 3$, обходы вершин всех множеств V_r , $r \in N_{\bar{x}}^0$, являются нулевыми. Всякий гамильтонов контур обязательно содержит все H -дуги, и в данном случае он должен содержать также все F -дуги вида $([r, 2p + 1], [r, 2p])$, $r \in N_{\bar{x}}^0$. Однако эти F -дуги вместе с H -дугами множества $H_{\bar{x}}$ образуют контур $K_{\bar{x}}$, что невозможно.

Следовательно, имеется ровно n_0 множеств V_r , $r \in N^0$, обходы вершин которых являются μ -присоединенными. Тройки e_r , соответствующие этим множествам V_r , образуют парасочетание. Действительно, если две какие-либо тройки $e_{r'}$ и $e_{r''}$ имеют общую p -ю компоненту, то найдется элемент $\bar{x} \in X_p$ такой, что обходы вершин всех множеств V_r , $r \in N_{\bar{x}}^0$, являются нулевыми, что невозможно.

Поскольку задача о трехмерном парасочетании NP -полна, а реализация ее сведения к рассматриваемой задаче требует полиномиального (относительно $|E|$) числа операций, то задача нахождения гамильтонова контура наименьшей длины в графе, длины дуг которого равны 0 или 1, является NP -трудной в сильном смысле.

2) Рассмотрим полный (ориентированный) граф $\bar{G} = (\bar{V}, \bar{U})$ без петель, вершины которого пронумерованы числами от 1 до $m = |\bar{V}|$. Каждой вершине $v \in \bar{V}$ сопоставлена тройка действительных неотрицательных чисел $(\alpha_v, \beta_v, \gamma_v)$. Длину дуги (v, w) , $v, w \in \bar{V}$, $v \neq w$, положим равной

$$\bar{c}_{vw} = \max \{0, \alpha_w - \beta_v, \gamma_v - \beta_w\}. \quad (3.7)$$

Покажем, что задача построения гамильтонова контура минимальной длины в графе $\bar{G}$ является NP -трудной в сильном смысле.

Сформулируем соответствующую задачу распознавания: определить, существует ли в графе $\bar{G}$ гамильтонов контур, длина которого не превосходит заданного числа $\bar{y}$.

Покажем, что к этой задаче сводится задача о существовании гамильтонова контура, длина которого не превосходит заданного числа y , в графе $G = (V, U)$, введенном в рассмотрение в п. 1).

Построим граф $\bar{G}$, исходя из графа G , следующим образом. Положим $\bar{V} = V = \{[r, j] \mid r \in N^0, j = 1, \dots, 8\}$. Пронумеруем вершины графа $\bar{G}$ числами от 1 до $m = |V|$. Наряду с обозначением вида $[r, j]$ для вершин графа $\bar{G}$ будем употреблять их номера v, w и т.п.

Для любой вершины $[r, j] \in \bar{V}$ положим

$$\beta_{[i, j]} = \begin{cases} (m + j + 8(r - 1))y, & \text{если } j \text{ нечетно;} \\ (j + 8(r - 1))y, & \text{если } j \text{ четно.} \end{cases}$$

Кроме того, положим $\alpha_{[r, 8]} = \beta_{[r, 1]}$, $\alpha_{[r, j]} = \beta_{[r, j-1]}$, $j \in \{2, 4, 6\}$, и $\alpha_{[r, j]} = \beta_v$, $j \in \{1, 3, 5, 7\}$, где дуга $(v, [r, j])$ является H -дугой в графе G .

Наконец, положим $\gamma_{[r, 1]} = 1 + \beta_{[r, 2]}$, $\gamma_{[r, j]} = \beta_{[r, j-1]}$, $j \in \{3, 5, 7\}$ и $\gamma_{[r, j]} = \beta_w$, $j \in \{2, 4, 6, 8\}$, где дуга $([r, j], w)$ является H -дугой в графе G .

Покажем, что в графе $\bar{G}$, длины дуг которого вычислены по формуле (3.7), гамильтонов контур, длина которого не превосходит $\bar{y} = y$, существует тогда и только тогда, когда в графе G существует гамильтонов контур, длина которого не превосходит y .

а) Пусть в G существует искомый гамильтонов контур. Нетрудно проверить, что длины дуг графа G совпадают с длинами соответствующих дуг графа $\bar{G}$. Следовательно, в графе $\bar{G}$ также существует гамильтонов контур, длина которого не превосходит y .

б) Пусть в графе $\bar{G}$ существует гамильтонов контур $\bar{K}$, длина которого не превосходит $\bar{y}$. Покажем, что в этом случае ни одна из дуг множества $\bar{U} \setminus U$ не принадлежит контуру $\bar{K}$.

Покажем, что контур $\bar{K}$ содержит множество всех H -дуг графа G . Напомним, что каждая из этих дуг исходит из четной вершины и заходит в нечетную. Заметим также, что контур $\bar{K}$ не содержит дуг вида (v, w) , исходящих из четной вершины и заходящих в четную, так как в этом случае $\alpha_w \geq \beta_v + y$.

Для каждой H -дуги (v, w) имеем $\bar{c}_{vw} = 0$, так как $\alpha_w = \beta_v$, $\gamma_v = \beta_w$ по построению. Множество H -дуг вида (v, w) упорядочим по возрастанию значений $\beta_v = \alpha_w$ или, что то же самое, по возрастанию (в лексикографическом смысле) их начальных вершин вида $[r, 2q]$, $1 \leq q \leq 4$. Пусть (v', w') — первая (относительно указанного порядка) H -дуга, а w — произвольная нечетная вершина. Так как $\alpha_w - \beta_{v'} \geq 2y$ при $w \neq w'$, то контур $\bar{K}$ должен содержать дугу (v', w') . Рассматривая H -дуги последовательно одну за другой (относительно указанного порядка), убеждаемся, что все они принадлежат контуру $\bar{K}$. Следовательно, контур $\bar{K}$ не содержит дуг, исходящих из нечетной вершины и заходящих в нечетную.

Покажем, что в контуре $\bar{K}$ дуга, исходящая из нечетной вершины $[r, j]$ и заходящая в четную вершину $[r', j']$, является F -дугой. Действительно, если вершина $[r, j]$ лексикографически предшествует вершине $[r', j']$, то $\alpha_{[r', j']} \leq \beta_{[r, j]}$ при условии, что $r' = r, j' \in \{j + 1, 8\}$ и $\alpha_{[r', j']} - \beta_{[r, j]} > y$ в противном случае. Кроме того, $\gamma_{[r, j]} \leq \beta_{[r', j']}$ с единственным исключением $\gamma_{[r, 1]} = 1 + \beta_{[r, 2]}$.

Если $[r', j']$ лексикографически предшествует $[r, j]$, то $\gamma_{[r, j]} = \beta_{[r', j']}$, если $r' = r, j' = j - 1$, и $\gamma_{[r, j]} - \beta_{[r', j']} > y$ в противном случае. Кроме того, $\alpha_{[r', j']} \leq \beta_{[r, j]}$.

Таким образом, в графе $\bar{G}$ длина любой дуги, не принадлежащей графу G , больше, чем y .

Тем самым за $O(|V|^2)$ операций построено сведение задачи о гамильтоновом контуре в графе G к задаче о гамильтоновом контуре в графе $\bar{G}$.

3) Рассмотрим полный ориентированный граф $G_0 = (V_0, U_0)$, вершины которого пронумерованы числами от 0 до $m = |V_0| - 1$. Каждой вершине $v \in V_0$ сопоставлена тройка действительных неотрицательных чисел $(\alpha'_v, \beta'_v, \gamma'_v)$, причем $\alpha'_0 = \beta'_0 = \gamma'_0 = 0$. Покажем, что задача построения гамильтонова контура наименьшей длины в графе G_0 является NP-трудной в сильном смысле, если веса дуг вычислены по формуле (3.7).

Сформулируем соответствующую задачу распознавания: определить, существует ли в графе G_0 гамильтонов контур, длина которого не превосходит заданного числа y_0 .

Покажем, что к этой задаче сводится задача о существовании гамильтонова контура, длина которого не больше заданного числа $\bar{y}$, в графе $\bar{G}$, введенном в рассмотрение в п. 2).

Положим $V_0 = \{0\} \cup \bar{V}$, $\alpha'_0 = \beta'_0 = \gamma'_0 = 0$; $\alpha'_v = \alpha_v, \beta'_v = \beta_v, \gamma'_v = \gamma_v$, $v \in \bar{V}$. Выберем в графе $\bar{G}$ произвольную H -дугу вида $(\bar{v}, \bar{w})$ и положим $\gamma'_{\bar{v}} = \alpha'_{\bar{w}} = 0$. Вычислим длины дуг по формуле (3.7). Покажем, что в графе G_0 гамильтонов контур K_0 , длина которого не более $y_0 = \bar{y}$, существует тогда и только тогда, когда в графе $\bar{G}$ существует контур $\bar{K}$, длина которого не больше $\bar{y}$.

Действительно, пусть существует такой контур $\bar{K}$. Заменяем дугу $(\bar{v}, \bar{w}) \in \bar{K}$ парой дуг $(\bar{v}, 0)$ и $(0, \bar{w})$. Получим контур K'_0 , длина которого равна длине контура $\bar{K}$.

Пусть существует контур K_0 . Поскольку длины всех дуг, заходящих в вершину 0 (кроме дуги $(\bar{v}, 0)$) и исходящих из этой вершины (кроме дуги $(0, \bar{w})$), больше $\bar{y}$, то дуги $(\bar{v}, 0)$ и $(0, \bar{w})$ принадлежат контуру K_0 . Следовательно, дуга $(\bar{v}, \bar{w})$ этому контуру не принадлежит. Заменяя дуги $(\bar{v}, 0)$ и $(0, \bar{w})$ дугой $(\bar{v}, \bar{w})$, получаем контур $\bar{K}$.

4) Покажем наконец, что задача построения оптимального по быстродействию расписания обслуживания без задержек множества $N = \{1, 2, \dots, n\}$ требований в системе поточного типа из трех приборов эквивалентна задаче построения гамильтонова контура минимальной длины в графе G_0 (при $m = n$).

Действительно, из формулы (3.2) при $M = 3$ следует, что

$$\begin{aligned} \bar{t}_{\max}(\pi) = & \sum_{k=1}^{n-1} \max \{ t_{i_{k1}}, t_{i_{k1}} + t_{i_{k2}} - t_{i_{k+1}1}, t_{i_{k1}} + t_{i_{k2}} + \\ & + t_{i_{k3}} - t_{i_{k+1}1} - t_{i_{k+1}2} \} + \sum_{L=1}^3 t_{i_n L} = t_{i_1 1} + \sum_{j=1}^n t_{j2} + \\ & + \sum_{k=1}^{n-1} \max \{ 0, t_{i_{k+1}1} - t_{i_{k2}}, t_{i_{k3}} - t_{i_{k+1}2} \} + t_{i_n 3}. \end{aligned}$$

Если положить $\alpha'_0 = \beta'_0 = \gamma'_0 = 0$, $\alpha'_i = t_{i1}$, $\beta'_i = t_{i2}$, $\gamma'_i = t_{i3}$, $i \in N$, то задача нахождения перестановки π^* , минимизирующей значение $\bar{t}_{\max}(\pi)$, эквивалентна задаче нахождения гамильтонова контура K_0^* минимальной длины в графе G_0 , длины дуг которого вычислены по формуле (3.7).

Поскольку задача построения контура K_0^* является NP -трудной в сильном смысле и ее сведение к рассматриваемой задаче реализовано за $O(|V_0|)$ операций, то задача построения оптимального расписания в данном случае также является NP -трудной в сильном смысле.

§ 4. Максимальное временное смещение

В систему поточного типа, состоящую из M приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительности $t_{iL} \geq 0$ обслуживания требования $i \in N$ прибором L , $L = 1, \dots, M$, предполагаются заданными. Для каждого требования $i \in N$ задан директивный срок D_i , к которому желательно завершить обслуживание требования i всеми приборами. В этом параграфе рассматривается задача построения расписания s^* , при котором значение максимального временного смещения $L_{\max}(s) = \max\{\bar{t}_i(s) - D_i \mid i \in N\}$ является наименьшим. Показано, что эта задача является NP -трудной в сильном смысле при $M \geq 2$ независимо от того, допускаются прерывания в процессе обслуживания или нет. Задача минимизации максимального временного смещения оказывается NP -трудной в сильном смысле при $M \geq 2$ и в том случае, когда все требования обслуживаются без задержек.

4.1. Покажем, что задача построения расписания s^* (как в случае, когда допускаются прерывания, так и в случае, когда прерывания запрещены) является NP -трудной в сильном смысле при $M \geq 2$.

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 , при котором $L_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что при $M = 2$ к этой задаче сводится задача о 3-разбиении (см. вводную часть этой главы).

Положим $n = 4n_0 + 1$, $N = \{1, 2, \dots, n\}$. Множество требований разобьем на две группы: U -требования, обозначаемые U_i , $i = 1, \dots, 3n_0$, и V -требования, обозначаемые V_j , $j = 1, \dots, n_0 + 1$. Положим

$$t_{U_i 1} = e_i, \quad t_{U_i 2} = 2e_i, \quad D_{U_i} = (3n_0 + 2)E, \quad i = 1, \dots, 3n_0;$$

$$t_{V_j 1} = 2E, \quad t_{V_j 2} = E, \quad D_{V_j} = 3jE, \quad j = 1, \dots, n_0;$$

$$t_{V_{n_0+1} 1} = 2E, \quad t_{V_{n_0+1} 2} = 0, \quad D_{V_{n_0+1}} = (3n_0 + 2)E.$$

Покажем, что при $y = 0$ в построенной задаче расписание s^0 (как с прерываниями, так и без прерываний) существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Пусть задача о 3-разбиении имеет решение и N_j^0 , $j = 1, \dots, n_0$, — найденные трехэлементные подмножества элементов множества N^0 . Тогда расписание s^0 существует и может быть построено, например, следующим образом. Все требования обслуживаются без прерываний прибором 1 во временном интервале $(0, (3n_0 + 2)E]$ и прибором 2 во временном интервале $(2E, (3n_0 + 2)E]$. Оба прибора в указанных интервалах функционируют

без простоев и обслуживают требования в одной и той же последовательности $(V_1, \pi_1, V_2, \pi_2, \dots, V_{n_0}, \pi_{n_0}, V_{n_0+1})$, где π_j — произвольная последовательность U -требований с номерами из множества $N_j^0, j = 1, \dots, n_0$.

Пусть существует расписание s^0 . Поскольку $t_{V_1,1} + t_{V_2,2} = 3E = D_{V_1}$, то при расписании s^0 требование V_1 обслуживается без прерываний прибором 1 в интервале $(0, 2E]$ и прибором 2 в интервале $(2E, 3E]$. Поскольку суммарная длительность обслуживания остальных требований прибором 1 равна $3n_0E$, а прибором 2 — $(3n_0 + 1)E$, то при расписании s^0 оба прибора функционируют без простоев в интервалах $(0, (3n_0 + 2)E]$ и $(2E, (3n_0 + 2)E]$ соответственно.

Рассмотрим временной интервал $(3E, 6E]$. В этом интервале прибор 2 должен обслужить требование V_2 . При этом прибор 2 не может начать обслуживание ни одного из оставшихся V -требований. Действительно, пусть V' — некоторое V -требование, отличное от V_1 и V_2 . Даже если $t_{V_2,1}^0 = 2E$ и $\bar{t}_{V_2,1} = t_{V',1}^0 = 4E$, то $t_{V',2}^0 \geq 6E$. Следовательно, в интервале $(3E, 6E]$ прибор 2 должен обслуживать некоторые U -требования в течение $2E$ единиц времени. Перед обслуживанием прибором 2 эти требования должны быть обслужены прибором 1 в течение E единиц времени.

Очевидно, при расписании s^0 значение $\bar{t}_{V_2,1} \leq 5E$, иначе $\bar{t}_{V_2,2} > D_{V_2}$. Предположим, что $\bar{t}_{V_2,1} = 5E - C$, где $E > C > 0$. Тогда прибор 1 в интервале $(2E, 5E - C]$ обслуживает некоторые U -требования в течение $E - C$ единиц времени. Так как $t_{V_2,2}^0 \geq \bar{t}_{V_2,1}$, то прибор 2 должен обслуживать эти U -требования в интервале $(3E, t_{V_2,2}^0]$ по крайней мере в течение $2E - C$ единиц времени. Это означает, что при расписании s^0 прибор 1 в интервале $(2E, 5E]$ обслуживает три U -требования, суммарная длительность обслуживания которых равна E , а также требование V_2 , причем $\bar{t}_{V_2,1} = 5E$. Если прерывания запрещены, то требование V_2 обслуживается в интервале $(3E, 5E]$, а U -требования — в интервале $(2E, 3E]$. Прибор 2 обслуживает требование V_2 без прерываний в интервале $(5E, 6E]$, а также обслуживает (возможно, с прерываниями, если они разрешены) в интервале $(3E, 5E]$ те три U -требования, обслуживание которых прибором 1 завершено.

Рассуждая аналогичным образом, убеждаемся, что при расписании s^0 прибор 2 в каждом из временных интервалов $(3jE, (3j+2)E]$, $1 \leq j \leq n_0$, обслуживает по три U -требования в течение $2E$ единиц времени. Обозначая через N_j^0 множество номеров U -требований, обслуживаемых прибором 2 в интервале $(3jE, (3j+2)E]$, $j = 1, \dots, n_0$, получаем решение задачи о 3-разбиении.

Таким образом, за $O(n_0)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

4.2. Пусть расписание s^* , минимизирующее значение $L_{\max}(s)$, ищется в классе расписаний без задержек.

Если среди t_{iL} имеются нули и запись $t_{iL} = 0$ означает, что требование i прибором L не обслуживается, то задача построения расписания s^* , очевидно, является NP -трудной в сильном смысле при $M \geq 2$, даже если все $D_i = D$. Этот вывод непосредственно следует из результатов п. 3.3 этой главы.

Пусть запись $t_{iL} = 0$ понимается таким образом, что требование i прибором L обслуживается, но длительностью этого обслуживания можно пренебречь. Напомним, что в этом случае оптимальное расписание s^* есте-

ственно искать в классе расписаний, при которых все приборы обслуживают требования в одной и той же последовательности.

Если $D_i = D, i \in N$, то задача построения расписания s^* является NP -трудной в сильном смысле при $M \geq 3$ (см. п. 3.7.) При $M = 2$ расписание s^* может быть построено за $O(n \log n)$ операций (см. п. 3.5).

Покажем, что задача построения s^* является NP -трудной в сильном смысле, если $M = 2$ и не все D_i одинаковы.

Предварительно установим справедливость следующего вспомогательного утверждения.

Пусть $\tilde{G} = (V, \tilde{U})$ — неориентированный граф, содержащий гамильтонов цикл. Покажем, что ориентированный граф $G = (V, U)$, полученный из $\tilde{G}$ путем замены каждого ребра парой противоположно направленных дуг, содержит эйлеров контур, $|V|$ последних дуг которого образуют гамильтонов контур. Напомним, что эйлеровым контуром называется замкнутый остовный маршрут, в котором каждая дуга графа встречается ровно один раз. Для существования такого контура необходимо и достаточно, чтобы граф был (слабо) связан и у каждой его вершины полустепень захода была равна полустепени исхода.

По построению граф G содержит эйлеров контур. Кроме того, если граф $\tilde{G}$ содержит гамильтонов цикл C , то граф G содержит по крайней мере два гамильтонова контура C_1 и C_2 , соответствующих различным ориентациям ребер цикла C . Если из графа G удалить дуги контура C_2 , то полученный при этом граф G' содержит эйлеров контур E' , так как полустепени исхода и захода каждой вершины в графе G' ровно на единицу меньше соответствующих значений в графе G и, следовательно, равны между собой. Если контур E' дополнить контуром C_2 , то получим искомый эйлеров контур графа G .

Перейдем к рассмотрению задачи построения расписания s^* . Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 , при котором $L_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче сводится задача о гамильтоновом цикле (см. вводную часть этой главы). Будем считать, что $|V| = n_0$ и вершины графа $\tilde{G}$ пронумерованы числами от 0 до $n_0 - 1$. Степень вершины v обозначим через $d(v)$, $v = 0, \dots, n_0 - 1$. Напомним, что $\sum_{v \in V} d(v) = 2|\tilde{U}|$.

Положим $n = 4|\tilde{U}|$, $N = \{1, 2, \dots, n\}$. Каждому ребру $[v, w] \in \tilde{U}$ поставим в соответствие два требования U_{vw1} и U_{vw2} . Эти требования будем называть U -требованиями. Каждой вершине $v \in V$ поставим в соответствие $d(v)$ требований V_{vj} , $j = 1, \dots, d(v)$. Эти требования будем называть V -требованиями. Положим

$$t_{U_{vw1}} = v, \quad t_{U_{vw2}} = 2n_0 - w, \quad [v, w] \in \tilde{U},$$

$$t_{U_{vw1}} = w, \quad t_{U_{vw2}} = 2n_0 - v, \quad [v, w] \in \tilde{U},$$

$$t_{V_{vj1}} = 2n_0 - v, \quad t_{V_{vj2}} = v, \quad v \in V, \quad j = 1, \dots, d(v);$$

$$D_1 = 2n_0(2|\tilde{U}| - n_0), \quad D_2 = 4n_0|\tilde{U}|.$$

Директивный срок D_1 назначим всем V -требованиям, кроме требований $V_{vd}(v)$, $v = 0, \dots, n_0 - 1$. Всем остальным требованиям назначим директивный срок D_2 .

Покажем, что при $y = 0$ в построенной задаче расписание s^0 существует тогда и только тогда, когда в графе G существует гамильтонов цикл.

1) Пусть в графе $\tilde{G}$ существует гамильтонов цикл. Построим граф $G = (V, U)$, заменив каждое ребро графа $\tilde{G}$ парой противоположно направленных дуг. Как было показано выше в графе $\tilde{G}$ существует эйлеров контур, n_0 последних дуг которого образуют гамильтонов контур.

Совершим обход указанного эйлерова контура, начиная с вершины 0. В результате может быть выписана чередующаяся последовательность дуг и вершин графа G , начинающаяся с дуги, исходящей из вершины 0, и заканчивающаяся вершиной 0. Каждой дуге (v, w) из этой последовательности поставим в соответствие требование U_{vw} . Каждой вершине v , встретившейся в этой последовательности j -й по порядку раз, сопоставим требование V_{vj} , $j = 1, \dots, d(v)$. В результате получаем чередующуюся последовательность U - и V -требований. Обозначим эту последовательность через π .

Будем считать, что оба прибора обслуживают требования в последовательности π и функционируют без простоев во временном интервале $(0, D_2]$. Последовательность π можно разбить на $2|\tilde{U}|$ подпоследовательностей — пар вида (V_{vj}, U_{vw}) в случае прибора 1 и вида (U_{vw}, V_{wj}) в случае прибора 2. Для обслуживания каждой такой пары требований соответствующим прибором необходимо $2n_0$ единиц времени. Из всех V -требований в последних n_0 парах указанного вида содержится только требования $V_{vd}(v)$, $v = 0, \dots, n_0 - 1$. Следовательно, оставшиеся V -требования будут обслужены до момента времени $D_2 - 2n_0^2 = D_1$. Таким образом, построенное расписание является искомым расписанием s^0 .

2) Пусть существует расписание s^0 . Определим суммарную длительность обслуживания всех требований каждым прибором. Для прибора 1 имеем

$$\begin{aligned} & \sum_{[v,w] \in \tilde{U}} t_{U_{vw}1} + \sum_{[v,w] \in \tilde{U}} t_{U_{wv}1} + \sum_{v=0}^{n_0-1} \sum_{j=1}^{d(v)} t_{V_{vj}1} = \\ & = \sum_{v=0}^{n_0-1} v d(v) + \sum_{v=0}^{n_0-1} (2n_0 - v) d(v) = 4n_0 |\tilde{U}| = D_2. \end{aligned}$$

Аналогично для прибора 2

$$\begin{aligned} & \sum_{[v,w] \in \tilde{U}} t_{U_{vw}2} + \sum_{[v,w] \in \tilde{U}} t_{U_{wv}2} + \sum_{v=0}^{n_0-1} \sum_{j=1}^{d(v)} t_{V_{vj}2} = \\ & = \sum_{v=0}^{n_0-1} (2n_0 - v) d(v) + \sum_{v=0}^{n_0-1} v d(v) = 4n_0 |\tilde{U}| = D_2. \end{aligned}$$

Следовательно, при расписании s^0 оба прибора функционируют без простоев в интервале $(0, D_2]$.

Пусть при расписании s^0 требования обслуживаются обоими приборами в последовательности π^0 . Нетрудно видеть, что π^0 является чередующейся последовательностью U - и V -требований. Сопоставим требованию U_{vw} дугу (v, w) , а требованию V_{vj} — вершину v графа G . В результате получаем чередующуюся последовательность дуг и вершин графа G . Поскольку s^0 является расписанием без задержек и оба прибора функционируют без простоев, то указанная последовательность определяет некоторый обход

эйлерова контура в графе G . Так как при расписании s^0 значения $\bar{t}_{V_{vj}2} \leq \leq D_1, j = 1, \dots, d(v) - 1, v \in V$, то n_0 последних дуг этого контура образуют гамильтонов контур. Следовательно, граф $\tilde{G}$ имеет гамильтонов цикл.

Поскольку число требований, а также длительности их обслуживания и директивные сроки полиномиально зависят от $|V| = n_0$ и задача о гамильтоновом цикле является NP -полной, то задача построения расписания s^* в рассматриваемом случае NP -трудна в сильном смысле.

§ 5. Суммарное время обслуживания

В многостадийную систему поточного типа, состоящую из $M \geq 2$ приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Заданы длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L, L = 1, \dots, M$. Прерывания процесса обслуживания каждого требования каждым прибором не допускаются. Рассмотрим задачу построения расписания s^* , при котором суммарное время обслуживания

$\sum_{i \in N} \bar{t}_i(s)$ минимально. Здесь, как и прежде, $\bar{t}_i(s)$ — момент завершения

обслуживания требования i при расписании s .

5.1. Покажем, что эта задача является NP -трудной в сильном смысле при $M \geq 2$.

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 , при котором $\sum_{i \in N} \bar{t}_i(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче при $M = 2$ сводится задача о 3-разбиении (см. вводную часть этой главы). Положим

$$u = 3n_0E + 1, v = u + 3n_0E + n_0u + n_0(n_0 - 1)u(E + 1)/2,$$

$$r = v - u, f = uv + E + 1, x = 2(n_0 + 2)f + v,$$

$$n = n_0 + 1 + v + n_0u, N = \{1, 2, \dots, n\}.$$

Множество N требований разобьем на три группы: F -требования, обозначаемые $F_j, j = 0, \dots, n_0$, X -требования, обозначаемые $X_l, l = 1, \dots, v$, а также U -требования, среди которых будем различать V -требования, обозначаемые $V_{jk}, j = 1, \dots, n_0, k = 1, \dots, u - 3$, и W -требования, обозначаемые $W_i, i = 1, \dots, 3n_0$.

Доказательство приведем для случая, когда запись $t_{iL} = 0$ означает, что требование i прибором L не обслуживается.

Положим

$$t_{F_01} = 0, t_{F_02} = 1;$$

$$t_{F_j1} = f, t_{F_j2} = 1; j = 1, \dots, n_0;$$

$$t_{X_l1} = 0, t_{X_l2} = x, l = 1, \dots, v;$$

$$t_{V_{jk}1} = 0, \quad t_{V_{jk}2} = v, \quad j = 1, \dots, n_0, \quad k = 1, \dots, u-3;$$

$$t_{W_{i1}} = 0, \quad t_{W_{i2}} = v + e_i, \quad i = 1, \dots, 3n_0.$$

Положим также $y = \bar{F} + \bar{X} + \bar{U}$, где

$$\bar{F} = \sum_{j=0}^{n_0} (jf + 1) = n_0 + 1 + n_0(n_0 + 1)f/2,$$

$$\bar{X} = \sum_{l=1}^v (n_0f + 1 + lx) = v(n_0f + 1) + v(v+1)x/2,$$

$$\bar{U} = \sum_{j=0}^{n_0-1} \sum_{k=1}^u (kv + jf + 1) + 3n_0E = n_0u(n_0 + 1)v/2 + r.$$

Покажем, что в построенной задаче расписание s^0 , при котором $\sum_{t \in N} \bar{t}_t(s^0) \leq y$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

1) Пусть задача о 3-разбиении имеет решение и $N_j^0 = \{\alpha_{j1}, \alpha_{j2}, \alpha_{j3}\}$, $j = 1, \dots, n_0$, — найденные трехэлементные подмножества элементов множества N^0 . Построим расписание, при котором прибор 1 обслуживает F -требования в последовательности $(F_1, F_2, \dots, F_{n_0})$, функционируя без простоев в интервале $(0, n_0f]$. Прибор 2 функционирует без простоев в интервале $(0, n_0vu + 3n_0E + vx]$ и обслуживает требования множества N в последовательности $(F_0, V_{11}, V_{12}, \dots, V_{1,u-3}, W_{\alpha_{11}}, W_{\alpha_{12}}, W_{\alpha_{13}}, F_1, V_{21}, V_{22}, \dots, V_{2,u-3}, W_{\alpha_{21}}, W_{\alpha_{22}}, W_{\alpha_{23}}, F_2, \dots, F_{n_0}, X_1, X_2, \dots, X_v)$. При построенном расписании

$$\bar{t}_{F_j} = jf + 1, \quad j = 0, \dots, n_0;$$

$$\bar{t}_{X_l} = n_0f + 1 + lx, \quad l = 1, \dots, v;$$

$$\bar{t}_{V_{jk}} = (j-1)f + 1 + kv, \quad j = 1, \dots, n_0, \quad k = 1, \dots, u-3;$$

$$\bar{t}_{W_{\alpha_{j1}}} = (j-1)f + 1 + (u-2)v + e_{\alpha_{j1}}, \quad j = 1, \dots, n_0;$$

$$\bar{t}_{W_{\alpha_{j2}}} = (j-1)f + 1 + (u-1)v + e_{\alpha_{j1}} + e_{\alpha_{j2}}, \quad j = 1, \dots, n_0;$$

$$\bar{t}_{W_{\alpha_{j3}}} = (j-1)f + 1 + uv + E, \quad j = 1, \dots, n_0.$$

Имеем

$$\sum_{j=0}^{n_0} \bar{t}_{Fj} = \bar{F}, \quad \sum_{l=1}^v \bar{t}_{Xl} = \bar{X};$$

$$\begin{aligned} & \sum_{j=1}^{n_0} \sum_{k=1}^{u-3} \bar{t}_{Vjk} + \sum_{j=1}^{n_0} \bar{t}_{W\alpha j1} + \sum_{j=1}^{n_0} \bar{t}_{W\alpha j2} + \sum_{j=1}^{n_0} \bar{t}_{W\alpha j3} = \\ & = \sum_{j=1}^{n_0} \sum_{k=1}^{u-3} ((j-1)f+1+kv) + \sum_{j=1}^{n_0} 3((j-1)f+1)+(u-2)v+(u-1)v+uv+ \\ & + \sum_{j=1}^{n_0} (3e_{\alpha j1} + 2e_{\alpha j2} + e_{\alpha j3}) < \\ & < \sum_{j=1}^{n_0} \sum_{k=1}^u ((j-1)f+1+kv) + 3 \sum_{j=1}^{n_0} (e_{\alpha j1} + e_{\alpha j2} + e_{\alpha j3}) = \\ & = \sum_{j=0}^{n_0-1} \sum_{k=1}^u (jf+1+kv) + 3n_0E = \bar{U}. \end{aligned}$$

Следовательно, построенное расписание является искомым расписанием s^0 .

2) Пусть существует расписание s^0 . Не нарушая общности, можно считать, что при этом расписании прибор 1 обслуживает требование F_j во временном интервале $((j-1)f, jf]$, $j = 1, \dots, n_0$, а прибор 2 обслуживает F -требования таким образом, что $\bar{t}_{Fj2} \leq t_{Fj+12}^0$, $j = 0, \dots, n_0 - 1$. Поскольку X -требования не обслуживаются прибором 1 и имеют одинаковые длительности обслуживания прибором 2, то можно считать, что при расписании s^0 имеет место $\bar{t}_{Xl2} \leq t_{Xl+12}^0$, $l = 1, \dots, v - 1$.

а) Покажем, что при расписании s^0 значение $t_{Xl2}^0 \geq n_0f + 1$, $l = 1, \dots, v$. Предположим, что найдется требование X_l , такое, что $t_{Xl2}^0 < n_0f + 1$. Так как $x > n_0f + 1$, то $l' = 1$. При этом к моменту времени $\bar{t}_{Xl2}$ все F -требования будут обслужены прибором 1.

Очевидно, суммарное время обслуживания X -требований будет не меньше чем $X' = x + \sum_{l=2}^v (n_0f + 1 + lx) = \bar{X} - n_0f - 1$.

Так как $t_{Xl2}^0 < n_0f + 1$, то, по крайней мере, $\bar{t}_{F n_0 2} \geq \bar{t}_{Xl2} \geq x$, поэтому суммарное время обслуживания F -требований можно ограничить снизу величиной

$$F' = \sum_{j=0}^{n_0-1} (jf+1) + x > \bar{F} + (n_0+2)f + v.$$

Оценим снизу суммарное время обслуживания U -требуемых. Предположим, что длительность обслуживания каждого из них (включая W -требуемых) прибором 2 равна v и они обслуживаются прибором 2 непрерывно одно за другим в интервале $(0, n_0 v]$. В качестве нижней оценки получим

величину $U' = \sum_{q=1}^{n_0} qv = \bar{U} - r$. Таким образом, суммарное время обслуживания

всех требований множества N не меньше чем $F' + X' + U' > y + (n_0 + 2)f + v - n_0 f - 1 > y$, что невозможно.

б) Итак, $t_{X_1, 2}^0 \geq n_0 f + 1$. Покажем, что при расписании s^0 значение $t_{X_1, 2}^0 = n_0 f + 1$. Предположим от противного, что $t_{X_1, 2}^0 = n_0 f + 2$. Суммарное время обслуживания X -требуемых при этом не меньше чем

$$X'' = \sum_{l=1}^v (n_0 f + 2 + lx) = \bar{X} + v.$$

Тогда суммарное время обслуживания всех требований множества N не меньше чем $\bar{F} + X'' + U' = y + v - r > y$, что невозможно.

в) Покажем, что прибор 2, функционируя без простоев в интервале $(0, n_0 f + 1]$, обслуживает все F - и U -требуемых.

Действительно, в указанном интервале прибор 2 не обслуживает других требований (так как $t_{X_1, 2}^0 = n_0 f + 1$). Кроме того, суммарная длительность обслуживания F - и U -требуемых равна длине интервала. Если предположить, что прибор 2 простаивает во временном интервале $(0, n_0 f + 1]$, то найдется требование V (из числа F - и U -требуемых), обслуживание которого прибором 2 начнется после момента времени $\bar{t}_{X_1, 2} = n_0 f + 1 + x$. Так как $t_{V, 2} < t_{X_1, 2}$, то в последовательности, в которой требования обслуживаются прибором 2, требования X_1 и V можно поменять местами и суммарное время обслуживания при этом не увеличится. Но при построенном расписании $t_{X_1, 2}^0 > n_0 f + 1$, что невозможно.

г) Обозначим через U_j множество всех U -требуемых, обслуживаемых прибором 2 во временном интервале $(0, t_{F_j, 2}^0]$. Покажем, что при расписании s^0 значение $|U_j| = ju$, $j = 0, \dots, n_0$.

Пусть найдется такое q , $1 \leq q \leq n_0$, что $|U_q| < qu$. Даже если множество U_q содержит все W -требуемых, то суммарная длительность обслуживания требований этого множества и требований $F_1, F_2, \dots, F_{q-1}$ не больше чем $q + (qu - 1)v + n_0 E < qf \leq t_{F_q, 2}^0$. Следовательно, прибор 2 простаивает в интервале $(0, t_{F_q, 2}^0]$, что невозможно.

Пусть найдется такое q , $0 \leq q \leq n_0$, что $|U_q| > qu$. Даже если множество U_q не содержит ни одного W -требуемого, то суммарная длительность обслуживания требований этого множества и требований $F_1, F_2, \dots, F_{q-1}$ не меньше $q + (qu + 1)v = qf + v - qE$. Тогда суммарное время обслуживания

ния F -требуваний будет не меньше чем

$$F'' \doteq \sum_{j=0}^{n_0} (jf + 1) + v - qE = \bar{F} + v - qE.$$

Суммарное время обслуживания всех требований множества N окажется не меньше чем $F'' + \bar{X} + U' = y + v - qE - r > y$, что невозможно.

д) Обозначим через U_j^0 множество всех U -требуваний, обслуживаемых прибором 2 во временном интервале $(\bar{t}_{Fj-1,2}, t_{Fj,2}^0]$. Из доказанного выше следует, что при расписании s^0 значения $|U_j^0| = u$, $j = 1, \dots, n_0$. Покажем, что $t_{Fj,2}^0 = jf$, $j = 0, \dots, n_0$.

Пусть найдется такое q , $0 \leq q \leq n_0$, что $t_{Fq,2}^0 > qf$. Так как $t_{X_1,2}^0 = n_0f + 1 > \bar{t}_{Fn_0,2}$, то $q < n_0$. Поскольку $U_0 = \phi$, то $q \neq 0$.

Суммарное время обслуживания требований множества U_{q+1}^0 не меньше $\sum_{k=1}^u (qf + 2 + ku)$. С другой стороны, суммарное время обслуживания требований каждого из множеств U_j^0 , $j = 1, \dots, n_0$, $j \neq q + 1$, не меньше чем $\sum_{k=1}^u (jf + 1 + ku)$. Поэтому суммарное время обслуживания всех U -требуваний ограничено снизу величиной

$$U'' = \sum_{j=0}^{n_0-1} \sum_{k=1}^u (jf + 1 + ku) = \bar{U} + 1.$$

Тем самым, суммарное время обслуживания всех требований множества N не меньше чем $\bar{F} + \bar{X} + U'' = y + 1$, что невозможно.

е) Покажем, что имеет решение задача о 3-разбиении.

Поскольку число U -требуваний, обслуживаемых в каждом из интервалов $(\bar{t}_{Fj-1,2}, t_{Fj,2}^0]$, равно u и длина каждого из этих интервалов равна $uv + E$, то в каждом из них должно обслуживаться ровно $u - 3$ V -требуваний и ровно три W -требувания, суммарная длительность обслуживания которых равна $3v + E$. Обозначая через N_j^0 множество номеров W -требуваний, попавших во множество U_j^0 , $j = 1, \dots, n_0$, получаем решение задачи о 3-разбиении.

Тем самым за $O(n_0^2 E)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

Нетрудно видеть, что приведенное доказательство может быть использовано и в том случае, когда запись $t_{iL} = 0$ означает, что требование i прибором L обслуживается, но длительностью его обслуживания можно пренебречь. Для этого достаточно считать, что при расписании s^0 значение $\bar{t}_{i1} = t_{i1}^0 = 0$ для всех требований i , являющихся U - или X -требуваниями и положить $\bar{t}_{F_0,1} = t_{F_0,2}^0 = 0$.

5.2. В заключение этого параграфа рассмотрим задачу построения расписания s^* без задержек в обслуживании требований, которому соответствует наименьшее значение $\sum_{i \in N} \bar{t}_i(s)$.

Если среди t_{iL} имеются нули и запись $t_{iL} = 0$ понимается таким образом, что требование i прибором L не обслуживается, то задача построения s^* является *NP-трудной в сильном смысле при $M \geq 2$* . Это непосредственно следует из доказательства, приведенного в предыдущем пункте.

Пусть запись $t_{iL} = 0$ означает, что требование i обслуживается, но длительностью этого обслуживания можно пренебречь. Напомним, что в этом случае расписание естественно искать в классе расписаний, при которых все приборы обслуживают требования в одной и той же последовательности. Покажем, что и в этом случае задача построения расписания s^* является *NP-трудной при $M \geq 2$* .

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 обслуживания требований множества $N = \{1, 2, \dots, n\}$ без задержек, при котором $\sum_{i \in N} \bar{t}_i(s) \leq u$ для заданного числа u .

Покажем, что к этой задаче при $M = 2$ сводится задача о гамильтоновом цикле (см. п. 4.2) в неориентированном графе $\tilde{G} = (V, \tilde{U})$, где $|V| = n_0$ и вершины пронумерованы четными числами от 0 до $2n_0 - 2$. Степень вершины $v \in V$ обозначим через $d(v)$, $v = 0, 2, \dots, 2n_0 - 2$.

Множество N требований построим следующим образом. Каждому ребру $[v, w] \in \tilde{U}$ поставим в соответствие два требования U_{vw1} и U_{vw2} . Каждой вершине $v \in V$ поставим в соответствие $d(v)$ требований V_{vj} , $j = 1, \dots, d(v)$. Введем в рассмотрение также требования F_{pk} , $p \in \{1, 2\}$, $k = 1, \dots, n_0^2$.

Положим

$$t_{U_{vw}1} = v, \quad t_{U_{vw}2} = 8n_0 - w, \quad [v, w] \in \tilde{U};$$

$$t_{U_{vw}1} = w, \quad t_{U_{vw}2} = 8n_0 - v, \quad [v, w] \in \tilde{U};$$

$$t_{V_{vj}1} = 8n_0 - v + 1, \quad t_{V_{vj}2} = v, \quad v \in V, \quad j = 1, \dots, n_0;$$

$$t_{F_{1k}1} = 0, \quad t_{F_{1k}2} = 16n_0 + 1, \quad k = 1, \dots, n_0^2;$$

$$t_{F_{2k}1} = 16n_0 + 1; \quad t_{F_{2k}2} = 0, \quad k = 1, \dots, n_0^2;$$

$$y_1 = 16|\tilde{U}|(2|\tilde{U}| + 1)n_0 - \sum_{v \in V} vd(v) + n_0^2;$$

$$y_2 = 2n_0^2(4|\tilde{U}|n_0 + 1) + (16n_0 + 1)n_0^2(n_0^2 + 1).$$

При помощи рассуждений, аналогичных проведенным в п. 4.2, можно показать, что в построенной задаче расписание s^0 , при котором $\sum_{i \in N} \bar{t}_i(s) \leq y_1 + y_2$, существует тогда и только тогда, когда в графе $\tilde{G}$ существует гамильтонов цикл.

§ 6. Упорядоченные матрицы длительностей

В систему поточного типа, состоящую из M последовательных приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительность обслуживания требования $i \in N$ прибором L , $1 \leq L \leq M$, равна $t_{iL} \geq 0$. Запись $t_{iL} = 0$ означает, что требование i прибором L обслуживается, но длительностью этого обслуживания можно пренебречь. Прерывания в процессе обслуживания каждого требования каждым прибором не допускаются. Оптимальное расписание ищется в классе расписаний, при которых каждый прибор обслуживает требования в одной и той же последовательности. В п. 6.1, 6.2 рассматриваются задачи построения оптимальных последовательностей π^* обслуживания требований, которым соответствуют наименьшие значения общего $\bar{t}_{\max}(\pi)$ или суммарного

$\sum_{i \in N} \bar{t}_i(\pi)$ времени обслуживания. В п. 6.3, 6.4 эти же задачи решаются

в предположении, что задержки в обслуживании каждого требования не допускаются. Предполагается, что матрица $\|t_{iL}\|$ обладает определенными свойствами.

Будем говорить, что матрица $\|t_{iL}\|$ *упорядочена по строкам*, если для любой пары требований i и k справедливо либо $t_{iL} \geq t_{kL}$ для всех $L = 1, \dots, M$, либо $t_{iL} \leq t_{kL}$ для всех $L = 1, \dots, M$.

Аналогично будем говорить, что матрица $\|t_{iL}\|$ *упорядочена по столбцам*, если для любой пары приборов Q и H справедливо либо $t_{iH} \geq t_{iQ}$ для всех $i \in N$, либо $t_{iH} \leq t_{iQ}$ для всех $i \in N$.

Если матрица $\|t_{iL}\|$ одновременно упорядочена по строкам и по столбцам, то будем называть ее *упорядоченной*.

Так, например, матрица

$$\left\| \begin{array}{cccc} 10 & 12 & 15 & 7 \\ 2 & 8 & 14 & 6 \\ 1 & 3 & 7 & 5 \end{array} \right\|$$

упорядочена по строкам, в то время как матрица

$$\left\| \begin{array}{cccc} 10 & 12 & 15 & 11 \\ 2 & 8 & 14 & 6 \\ 1 & 5 & 7 & 3 \end{array} \right\|$$

является упорядоченной и по строкам, и по столбцам, (т.е. является упорядоченной).

6.1. Рассмотрим задачу построения оптимального по быстродействию расписания в предположении, что матрица $\|t_{iL}\|$ длительностей обслуживания является упорядоченной.

Напомним (см. п. 1.2 § 1 этой главы), что если требования обслуживаются каждым прибором в последовательности $\pi = (i_1, i_2, \dots, i_n)$, то

$$\bar{t}_{\max}(\pi) = \max_{1 \leq u_1 < u_2 < \dots < u_{M-1} \leq n} \left\{ \sum_{k=1}^{u_1} t_{i_{k1}} + \sum_{k=u_1}^{u_2} t_{i_{k2}} + \dots + \sum_{k=u_{M-1}}^n t_{i_{kM}} \right\}. \quad (6.1)$$

1) Поскольку матрица $\|t_{iL}\|$ является упорядоченной, то существуют прибор R такой, что $t_{iR} = \max \{t_{iL} \mid L = 1, \dots, M\}$ для всех $i \in N$, и требование $r \in N$ такое, что $t_{rL} = \max \{t_{iL} \mid i \in N\}$ для всех $L = 1, \dots, M$. Пусть в перестановке $\pi = (i_1, i_2, \dots, i_n)$ значение $i_q = r$ и максимум в (6.1) достигается при $u_L = v_L$, $L = 1, \dots, M-1$. Ясно, что $v_1 \leq v_2 \leq \dots \leq v_{M-1} \leq n$. Покажем, что существуют такие значения $v_1, v_2, \dots, v_{M-1}$, что $v_{R-1} \leq q \leq v_R$.

Предположим, что $q < v_{R-1}$. Пусть Q — такой прибор, что $v_{Q-1} \leq q \leq v_Q$. Положим

$$\begin{aligned} \hat{t}_{\max}(\pi) = & \sum_{k=1}^{v_1} t_{i_{k1}} + \dots + \sum_{k=v_{Q-2}}^{v_{Q-1}} t_{i_{k,Q-1}} + \sum_{k=v_{Q-1}}^q t_{i_{kQ}} + \\ & + \sum_{L=Q+1}^{R-1} t_{i_{qL}} + \sum_{k=q}^{v_R} t_{i_{kR}} + \sum_{k=v_R}^{v_{R+1}} t_{i_{k,R+1}} + \dots + \sum_{k=v_{M-1}}^n t_{i_{kM}}. \end{aligned}$$

Имеем

$$\begin{aligned} \hat{t}_{\max}(\pi) - \bar{t}_{\max}(\pi) = & \sum_{L=Q}^{R-1} t_{i_{qL}} + \sum_{k=q}^{v_R} t_{i_{kR}} - \\ & - \left(\sum_{k=q}^{v_Q} t_{i_{kQ}} + \sum_{k=v_Q}^{v_{Q+1}} t_{i_{k,Q+1}} + \dots + \sum_{k=v_{R-1}}^{v_R} t_{i_{kR}} \right). \end{aligned}$$

Поскольку в силу упорядоченности матрицы $\|t_{iL}\|$ справедливы неравенства

$$\begin{aligned} \sum_{k=q}^{v_R} t_{i_{kR}} & \geq \sum_{k=q}^{v_{Q-1}} t_{i_{kQ}} + \sum_{k=v_Q}^{v_{Q+1}-1} t_{i_{k,Q+1}} + \dots \\ & \dots + \sum_{k=v_{R-2}}^{v_{R-1}-1} t_{i_{k,R-1}} + \sum_{k=v_{R-1}}^{v_R} t_{i_{kR}}, \\ \sum_{L=Q}^{R-1} t_{i_{qL}} & \geq \sum_{L=Q}^{R-1} t_{i_{v_L L}}, \end{aligned}$$

то $\hat{t}_{\max}(\pi) \geq \bar{t}_{\max}(\pi)$. Из (6.1) следует, что $\bar{t}_{\max}(\pi) \geq \hat{t}_{\max}(\pi)$. Таким образом, $\hat{t}_{\max}(\pi) = \bar{t}_{\max}(\pi)$.

Полагая $v_L = q, L = Q, \dots, R - 1$, и не изменяя остальных значений v_L , получаем $v_{R-1} \leq q$.

Рассуждая аналогичным образом, можно показать, что существуют такие значения $v_1, v_2, \dots, v_{M-1}$, что $v_R \geq q$ (и неравенство $v_{R-1} \leq q$ не нарушается).

2) Пусть $R = 1$, т.е. $\max\{t_{iL} \mid L = 1, \dots, M\} = t_{i1}$ для всех $i \in N$. Покажем, что в этом случае оптимальная последовательность π^* обслуживания требований может быть построена за $O(n \log n)$ операций.

Доказательство проведем в несколько этапов.

а) Пусть при обслуживании требований в последовательности $\pi = (i_1, i_2, \dots, i_n)$ максимум в (6.1) достигается при $u_L = v_L, L = 1, \dots, M - 1$. Покажем, что существуют такие значения $v_1, v_2, \dots, v_{M-1}$, что $\max\{t_{i_{kL}} \mid k = v_1, \dots, n\} = t_{i_{v_1}L}, L = 1, \dots, M$.

Предположим, что $\max\{t_{i_{kL}} \mid k = v_1, \dots, n\} = t_{i_{pL}}, p > v_1$. Пусть $H, 2 \leq H \leq M$, — такой прибор, что $v_{H-1} > p$. Положим

$$\hat{t}_{\max}(\pi) = \sum_{k=1}^{p-1} t_{i_{k1}} + \sum_{L=1}^{H-1} t_{i_{pL}} + \sum_{k=p}^{v_H} t_{i_{kH}} + \dots + \sum_{k=v_{M-1}}^n t_{i_{kM}}.$$

Имеем

$$\begin{aligned} \hat{t}_{\max}(\pi) - \bar{t}_{\max}(\pi) &= \sum_{k=1}^{p-1} t_{i_{k1}} + \sum_{L=1}^{H-1} t_{i_{pL}} + \sum_{k=p}^{v_H} t_{i_{kH}} - \\ &- \left(\sum_{k=1}^{v_1} t_{i_{k1}} + \sum_{k=v_1}^{v_2} t_{i_{k2}} + \dots + \sum_{k=v_{H-1}}^{v_H} t_{i_{kH}} \right). \end{aligned}$$

Поскольку в силу упорядоченности матрицы $\|t_{iL}\|$ и сделанных предположений справедливы неравенства

$$\begin{aligned} \sum_{k=1}^{p-1} t_{i_{k1}} &\geq \sum_{k=1}^{v_1-1} t_{i_{k1}} + \sum_{k=v_1}^{v_2-1} t_{i_{k2}} + \dots \\ &\dots + \sum_{k=v_{H-1}}^{p-1} t_{i_{kH}}; \quad \sum_{L=1}^{H-1} t_{i_{pL}} \geq \sum_{L=1}^{H-1} t_{i_{v_L}L}, \end{aligned}$$

то $\hat{t}_{\max}(\pi) \geq \bar{t}_{\max}(\pi)$. С другой стороны, $\bar{t}_{\max}(\pi) \geq \hat{t}_{\max}(\pi)$. Следовательно, $\bar{t}_{\max}(\pi) = \hat{t}_{\max}(\pi)$.

Полагая $v_L = p, L = 1, \dots, H - 1$, и не изменяя остальных значений v_L , получаем искомые значения $v_1, v_2, \dots, v_{M-1}$.

б) Пусть максимум в (6.1) достигается при $u_L = v_L, L = 1, \dots, M - 1$, где $v_1 = v_2 = \dots = v_{Q-1} \leq v_Q \leq v_{Q+1} \leq \dots \leq v_{M-1} \leq n, Q \geq 2$. Покажем, что существует такое Q , что $\max\{t_{iL} \mid L = Q, \dots, M\} = t_{iQ}, i \in N$.

Предположим, что $\max\{t_{iL} \mid L = Q, \dots, M\} = t_{iH}, i \in N, H > Q$. По условию

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \sum_{k=1}^{v_1} t_{i_{k1}} + \sum_{L=2}^{Q-1} t_{i_{v_1}L} + \sum_{k=v_1}^{v_Q} t_{i_{kQ}} + \dots \\ &\dots + \sum_{k=v_{H-1}}^{v_H} t_{i_{kH}} + \dots + \sum_{k=v_{M-1}}^n t_{i_{kM}}. \end{aligned}$$

Положим

$$\hat{t}_{\max}(\pi) = \sum_{k=1}^{v_1} t_{ik1} + \sum_{L=2}^{H-1} t_{iv_1 L} + \sum_{k=v_1}^{v_H} t_{ikH} + \dots + \sum_{k=v_{M-1}}^n t_{ikM}.$$

Имеем

$$\begin{aligned} \hat{t}_{\max}(\pi) - \bar{t}_{\max}(\pi) &= \sum_{L=Q}^{H-1} t_{iv_1 L} + \sum_{k=v_1}^{v_H} t_{ikH} - \\ &- \left(\sum_{k=v_1}^{v_Q} t_{ikQ} + \sum_{k=v_Q}^{v_{Q+1}} t_{ik, Q+1} + \dots + \sum_{k=v_{H-1}}^{v_H} t_{ikH} \right). \end{aligned}$$

Из доказанного выше следует, что $\sum_{L=Q}^{H-1} t_{iv_1 L} \geq \sum_{L=Q}^{H-1} t_{iv_L L}$.

Кроме того, из условия упорядоченности матрицы $\|t_{iL}\|$ и сделанного предположения следует, что

$$\sum_{k=v_1}^{v_H} t_{ikH} \geq \sum_{k=v_1}^{v_{Q-1}} t_{ikQ} + \sum_{k=v_Q}^{v_{Q+1}-1} t_{ik, Q+1} + \dots + \sum_{k=v_{H-1}}^{v_H} t_{ikH}.$$

Таким образом, $\hat{t}_{\max}(\pi) \geq \bar{t}_{\max}(\pi)$. С другой стороны, $\bar{t}_{\max}(\pi) \geq \hat{t}_{\max}(\pi)$. Следовательно, $\hat{t}_{\max}(\pi) = \bar{t}_{\max}(\pi)$. Полагая $v_L = v_1$, $L = Q, \dots, H-1$, и не изменяя остальных значений v_L , получаем искомые значения $v_1, v_2, \dots, v_{M-1}$.

Рассуждая аналогичным образом, можно показать, что существуют такие значения $v_1, v_2, \dots, v_{M-1}$, при которых из неравенства $v_{Q-1} < v_Q$, $2 \leq Q \leq M-1$, следует, что $t_{iQ} = \max\{t_{iL} \mid L = Q, \dots, M\}$.

в) Пусть для перестановки $\pi = (i_1, i_2, \dots, i_p, i_{p+1}, \dots, i_n)$ значения $t_{ipL} \geq t_{ip+1L}$ для всех $L = 1, \dots, M$. Покажем, что $\bar{t}_{\max}(\pi) \leq \bar{t}_{\max}(\pi')$, где $\pi' = (i_1, i_2, \dots, i_{p-1}, i_{p+1}, i_p, i_{p+2}, \dots, i_n)$.

Предположим, что для последовательности π максимум в выражении (6.1) достигается при $u_L = v_L$, $L = 1, \dots, M-1$.

Если все v_L , $L = 1, \dots, M-1$, отличны от p и $p+1$, то $\bar{t}_{\max}(\pi') \geq \bar{t}_{\max}(\pi)$.

Пусть все v_L , $L = 1, \dots, M-1$, отличны от p , но существуют такие Q и H , $1 \leq Q < H \leq M-1$, что $v_L = p+1$, $L = Q, \dots, H-1$. Тогда

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \sum_{k=1}^{v_1} t_{ik1} + \dots + \sum_{k=v_{Q-1}}^{p+1} t_{ikQ} + \sum_{L=Q+1}^H t_{ip+1L} + \\ &+ \sum_{k=p+2}^{v_H} t_{ikH} + \dots + \sum_{k=v_{M-1}}^n t_{ikM}. \end{aligned}$$

Очевидно,

$$\begin{aligned} \bar{t}_{\max}(\pi') &\geq \sum_{k=1}^{v_1} t_{ik1} + \dots + \sum_{k=v_{Q-1}}^{p+1} t_{ikQ} + \sum_{L=Q+1}^H t_{ipL} + \\ &+ \sum_{k=p+2}^{v_H} t_{ikH} + \dots + \sum_{k=v_{M-1}}^n t_{ikM}. \end{aligned}$$

Следовательно,

$$\bar{t}_{\max}(\pi') - \bar{t}_{\max}(\pi) \geq \sum_{L=Q+1}^H t_{ipL} - \sum_{L=Q+1}^H t_{ip+1L}.$$

Так как по условию $t_{ipL} \geq t_{ip+1L}$ для всех $L = 1, \dots, M$, то $\bar{t}_{\max}(\pi') \geq \bar{t}_{\max}(\pi)$.

Пусть все $v_L, L = 1, \dots, M-1$, отличны от $p+1$, но существуют такие Q и $H, 1 \leq Q < H \leq M-1$, что $v_L = p, L = Q, \dots, H-1$. Из приведенных выше рассуждений следует, что $t_{iQ} = \max\{t_{iL} \mid L = Q, \dots, M\}, i \in N$.

Имеем

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \sum_{k=1}^{v_1} t_{ik1} + \dots + \sum_{k=v_{Q-1}}^p t_{ikQ} + \\ &+ \sum_{L=Q+1}^{H-1} t_{ipL} + \sum_{k=p}^{v_H} t_{ikH} + \dots + \sum_{k=v_{M-1}}^n t_{ikM}. \end{aligned}$$

Очевидно,

$$\begin{aligned} \bar{t}_{\max}(\pi') &\geq \sum_{k=1}^{v_1} t_{ik1} + \dots + \sum_{k=v_{Q-1}}^{p+1} t_{ikQ} + \\ &+ \sum_{L=Q+1}^{H-1} t_{ipL} + \sum_{k=p}^{v_H} t_{ikH} + \dots + \sum_{k=v_{M-1}}^n t_{ikM} - t_{ip+1H}. \end{aligned}$$

Следовательно,

$$\bar{t}_{\max}(\pi') - \bar{t}_{\max}(\pi) \geq t_{ip+1Q} - t_{ip+1H} \geq 0.$$

Аналогичным образом можно убедиться в справедливости неравенства $\bar{t}_{\max}(\pi') \geq \bar{t}_{\max}(\pi)$ и в случае, когда среди $v_L, L = 1, \dots, M$, содержатся значения p и $p+1$.

Таким образом, если $\max\{t_{iL} \mid L = 1, \dots, M\} = t_{i1}$ для всех $i \in N$, то искомая оптимальная последовательность π^* может быть построена за $O(n \log n)$ операций посредством упорядочения требований по невозрастанию значений t_{i1} .

Аналогичным образом можно показать, что если $R = M$, т.е. $\max\{t_{iL} \mid L = 1, \dots, M\} = t_{iM}$ для всех $i \in N$, то требования следует обслуживать в порядке убывания значений t_{iM} .

3) Пусть $\max\{t_{iL} \mid L = 1, \dots, M\} = t_{iR}, 2 \leq R \leq M-1$ для всех $i \in N$. Не нарушая общности, будем считать, что требования пронумерованы по убыванию значений t_{iR} . Покажем, что оптимальная перестановка π^* принадлежит классу так называемых *пирамидальных* перестановок, т.е. перестановок вида $(j_1, \dots, j_{q-1}, j_q, j_{q+1}, \dots, j_n)$, где $j_1 < \dots < j_{q-1}, j_q = n, j_{q+1} > \dots > j_n$ (при этом не исключаются случаи $q = 1$ и $q = n$).

Пусть $N' \subseteq N$ и $\pi_{N'}$ — некоторая перестановка элементов множества N' . Обозначим через $t_{QH}(\pi_{N'})$ момент завершения обслуживания требований множества N' в системе поточного типа, состоящей из приборов $Q, Q+1, \dots, H, 1 \leq Q < H \leq M$, при условии, что процесс обслуживания начинается в момент времени $d = 0$ и каждый из указанных приборов обслуживает требования в последовательности $\pi_{N'}$.

Любую перестановку π элементов множества N можно представить в виде $(\pi_{N_1}, n, \pi_{N_2})$, где $N_1 \cap N_2 = \emptyset$, $N_1 \cup N_2 = N \setminus \{n\}$. Введем в рассмотрение пирамидальную перестановку π' вида $(\pi'_{N_1}, n, \pi'_{N_2})$.

Покажем, что $\bar{t}_{\max}(\pi') \leq \bar{t}_{\max}(\pi)$. Из доказанного в п. а) следует, что

$$\bar{t}_{\max}(\pi) = \bar{t}_{1R}(\pi_{N_1}) + \bar{t}_{RM}(\pi_{N_2}) - t_{nR},$$

$$\bar{t}_{\max}(\pi') = \bar{t}_{1R}(\pi'_{N_1}) + \bar{t}_{RM}(\pi'_{N_2}) - t_{nR}.$$

Как показано в п. 2), $\bar{t}_{1R}(\pi'_{N_1}) \leq \bar{t}_{1R}(\pi_{N_1})$ и $\bar{t}_{RM}(\pi'_{N_2}) \leq \bar{t}_{RM}(\pi_{N_2})$. Это означает, что для любой перестановки π найдется такая пирамидальная перестановка π' , для которой $\bar{t}_{\max}(\pi') \leq \bar{t}_{\max}(\pi)$.

6.2. Покажем, что перестановка π^* элементов множества N которой соответствует наименьшее значение $\sum_{i \in N} \bar{t}_i(\pi)$, может быть построена за $O(n \log n)$ операций в предположении, что матрица $\|t_{iL}\|$ длительностей обслуживания либо упорядочена, либо $M = 2$ и $t_{i1} \geq t_{i2}$ для всех $i \in N$.

1) Если требования обслуживаются в последовательности $\pi = (i_1, i_2, \dots, i_n)$, то, как и прежде, через $\bar{t}_{i_k H}(\pi)$ будем обозначать момент завершения обслуживания требования i_k прибором H .

Покажем, что перестановке $\pi^* = (j_1, j_2, \dots, j_n)$ соответствует наименьшее значение $\sum_{i \in N} \bar{t}_i(\pi)$, если существует такой прибор H , $1 \leq H \leq M$, что

$$\bar{t}_{j_k M}(\pi^*) = \bar{t}_{j_k H}(\pi^*) + \sum_{L=H+1}^M t_{j_k L}, \quad k = 1, \dots, n, \quad (6.2)$$

и для любой перестановки $\pi = (i_1, i_2, \dots, i_n)$ справедливо неравенство

$$\sum_{k=1}^n \bar{t}_{j_k H}(\pi^*) \leq \sum_{k=1}^n \bar{t}_{i_k H}(\pi). \quad (6.3)$$

Действительно, из (6.1) следует, что $\bar{t}_{i_k M}(\pi) \geq \bar{t}_{i_k H}(\pi) + \sum_{L=H+1}^M t_{i_k L}$, $k = 1, \dots, n$, и тем самым

$$\begin{aligned} \sum_{k=1}^n \bar{t}_{i_k M}(\pi) &\geq \sum_{k=1}^n \bar{t}_{i_k H}(\pi) + \sum_{k=1}^n \sum_{L=H+1}^M t_{i_k L} = \\ &= \sum_{k=1}^n \bar{t}_{i_k H}(\pi) + \sum_{i=1}^n \sum_{L=H+1}^M t_{iL}. \end{aligned}$$

С другой стороны, из (6.2) следует, что

$$\begin{aligned} \sum_{k=1}^n \bar{t}_{j_k M}(\pi^*) &= \sum_{k=1}^n \bar{t}_{j_k H}(\pi^*) + \sum_{k=1}^n \sum_{L=H+1}^M t_{j_k L} = \\ &= \sum_{k=1}^n \bar{t}_{j_k H}(\pi^*) + \sum_{i=1}^n \sum_{L=H+1}^M t_{iL}. \end{aligned}$$

Учитывая неравенство (6.3), приходим к заключению, что

$$\sum_{i=1}^n \bar{t}_i(\pi^*) \leq \sum_{i=1}^n \bar{t}_i(\pi).$$

2) Пусть матрица $\|t_{iL}\|$ является упорядоченной и требования пронумерованы таким образом, что $t_{iL} \leq t_{i+1,L}$, $i = 1, \dots, n-1$, $1 \leq L \leq M$. Покажем, что $\pi^* = (1, 2, \dots, n)$ является искомой оптимальной последовательностью обслуживания требований.

Пусть $\pi = (i_1, i_2, \dots, i_n)$ — произвольная перестановка элементов множества N и R , $1 \leq R \leq M$, — такой прибор, что $\max\{t_{iL} \mid L = 1, \dots, M\} = t_{iR}$ для всех $i \in N$.

Как показано в п. 6.1 б) для всех k , $1 \leq k \leq n$, справедливо соотношение $\bar{t}_{kR}(\pi^*) \leq \bar{t}_{kR}(\pi)$ и, следовательно, выполняется соотношение (6.3) (при $H = R$).

В силу упорядоченности матрицы $\|t_{iL}\|$ из п. 6.1 а) следует, что

$$\bar{t}_{kM}(\pi^*) = \bar{t}_{kR}(\pi^*) + \sum_{L=R+1}^M t_{kL}, \quad k = 1, \dots, n,$$

т.е. имеет место соотношение (6.2) (при $H = R$). Таким образом, перестановка $\pi^* = (1, 2, \dots, n)$ является искомой. Для ее построения достаточно упорядочить требования по неубыванию длительностей обслуживания любым прибором, т.е. выполнить не более чем $O(n \log n)$ операций.

3) Пусть $M = 2$ и $t_{i1} \geq t_{i2}$ для всех $i \in N$, т.е. матрица $\|t_{iL}\|$ является упорядоченной по столбцам. Пронумеруем требования таким образом, что $t_{i1} \leq t_{i+1,1}$, $i = 1, \dots, n-1$. Покажем, что перестановка $\pi^* = (1, 2, \dots, n)$ является искомой.

Как известно, перестановка π^* соответствует наименьшее значение суммарного времени обслуживания всех требований прибором 1, т.е. имеет место соотношение (6.3) (при $H = 1$).

Поскольку $t_{k+1,1} \geq t_{k1} \geq t_{k2}$, $k = 1, \dots, n-1$, то

$$\bar{t}_{k2}(\pi^*) = \sum_{i=1}^k t_{i1} + t_{k2}, \quad k = 1, \dots, n,$$

т.е. имеет место соотношение (6.2) (при $H = 1$).

Следовательно, перестановке $\pi^* = (1, 2, \dots, n)$ соответствует наименьшее значение $\sum_{i=1}^n \bar{t}_i(\pi)$, и она может быть построена в результате выполнения не более чем $O(n \log n)$ операций.

6.3. Рассмотрим задачу построения оптимального по быстродействию расписания обслуживания множества $N = \{1, 2, \dots, n\}$ требований в системе поточного типа, состоящей из M приборов. Предполагается, что задержки в обслуживании каждого требования запрещены, матрица $\|t_{iL}\|$ упорядочена по строкам и требования пронумерованы таким образом, что $t_{iL} \leq t_{i+1,L}$, $i = 1, \dots, n-1$, $1 \leq L \leq M$.

Как показано в п. 3.4, 3) § 3 этой главы, последовательность π^* обслуживания требований, которой соответствует оптимальное расписание, может быть найдена в результате решения задачи о коммивояжере с

$n + 1$ городами и матрицей расстояний $\|c_{pq}\|$, где

$$c_{0p} = 0, \quad c_{p0} = \sum_{L=1}^M t_{pL},$$

$$c_{pq} = \max \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \mid Q=1, \dots, M \right\}, \quad p, q \in N, \quad p \neq q.$$

1) Покажем, что в рассматриваемом случае

$$c_{ik} - c_{il} \geq c_{ij} - c_{ij}, \quad (6.4)$$

где $n \geq j > k \geq 1, \quad n \geq l \geq i \geq 1$.

Для пары требований $(p, q), \quad p = 1, \dots, n, \quad q = 1, \dots, n, \quad p \neq q$, обозначим

$$\gamma_{pq}(H) = \max \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \mid Q=1, \dots, H \right\}, \quad H=1, \dots, M.$$

Очевидно, $c_{pq} = \gamma_{pq}(M)$.

Покажем, что для всех $L, \quad 1 \leq L \leq M$, справедливо $U(L) \geq V(L)$, где $U(L) = \gamma_{ik}(L) - \gamma_{ik}(L), \quad V(L) = \gamma_{ij}(L) - \gamma_{ij}(L)$.

Воспользуемся методом математической индукции. При $L = 1$ имеем $U(1) = V(1) = t_{i1} - t_{i1}$. Предположим, что $U(L) \geq V(L)$ для всех $L, \quad 1 \leq L \leq H-1 \leq M-1$. Покажем, что $U(H) \geq V(H)$.

Имеем

$$U(H) = U(H-1) + \max\{0, t_{iH} - t_{k,H-1}\} - \max\{0, t_{iH} - t_{k,H-1}\},$$

$$V(H) = V(H-1) + \max\{0, t_{iH} - t_{j,H-1}\} - \max\{0, t_{iH} - t_{j,H-1}\}.$$

а) Пусть $t_{iH} \leq t_{k,H-1}$. В силу упорядоченности матрицы $\|t_{iL}\|$ по строкам имеем $t_{iH} \leq t_{iH} \leq t_{k,H-1} \leq t_{j,H-1}$. Следовательно, $U(H) - V(H) = U(H-1) - V(H-1) \geq 0$.

б) Пусть $t_{iH} > t_{k,H-1}$ и $t_{iH} \leq t_{j,H-1}$. В этом случае $t_{iH} \leq t_{j,H-1}$.

Если $t_{iH} \leq t_{j,H-1}$, то $U(H) - V(H) = U(H-1) - V(H-1) + t_{iH} - t_{k,H-1} \geq 0$.

Если $t_{iH} > t_{j,H-1}$, то $U(H) - V(H) = U(H-1) - V(H-1) + t_{j,H-1} - t_{k,H-1} \geq 0$.

в) Пусть $t_{iH} > t_{k,H-1}$ и $t_{iH} > t_{j,H-1}$.

Если $t_{iH} \leq t_{j,H-1}$, а следовательно, и $t_{iH} \leq t_{j,H-1}$, то $U(H) - V(H) = U(H-1) - V(H-1) + t_{iH} - t_{iH} \geq 0$.

Если $t_{iH} > t_{j,H-1}$ и $t_{iH} \leq t_{j,H-1}$, то $U(H) - V(H) = U(H-1) - V(H-1) + t_{j,H-1} - t_{iH} \geq 0$.

Если же $t_{iH} < t_{j,H-1}$ и $t_{iH} > t_{j,H-1}$, то $U(H) - V(H) = U(H-1) - V(H-1) \geq 0$.

Таким образом, неравенство (6.4) выполняется.

2) Покажем, что в рассматриваемом случае оптимальная перестановка π^* принадлежит классу пирамидальных перестановок (см. п. 6.1, 3)).

Рассмотрим произвольную перестановку $\pi = (i_1, i_2, \dots, i_{p-1}, i_p, i_{p+1}, \dots, i_{q-1}, i_q, \dots, i_r = n, \dots, i_n)$, где $\max\{t_{ikL} \mid k = 1, \dots, r-1\} = t_{i_{qL}}$ и $\max\{t_{ikL} \mid k = 1, \dots, r-1, k \neq q\} = t_{i_{pL}}, L = 1, \dots, M$. Пока-

жем, что $\bar{t}_{\max}(\pi) \geq \bar{t}_{\max}(\pi')$, где $\pi' = (i_1, \dots, i_{p-1}, i_{p+1}, \dots, i_{q-1}, i_p, i_q, \dots, i_r = h, \dots, i_n)$. Учитывая результаты п. 3.4, 3) § 3 этой главы, нетрудно убедиться, что

$$\Delta = \bar{t}_{\max}(\pi) - \bar{t}_{\max}(\pi') = \\ = (c_{i_{p-1}i_p} + c_{i_pi_{p+1}} - c_{i_{p-1}i_{p+1}}) - (c_{i_{q-1}i_p} + c_{i_pi_q} - c_{i_{q-1}i_q})..$$

Покажем, что $\Delta \geq 0$.

Действительно, если $i_{q-1} > i_{p-1}$, то, прибавляя и вычитая значение $c_{i_{q-1}i_{p+1}}$, имеем

$$\Delta = ((c_{i_{p-1}i_p} - c_{i_{q-1}i_p}) - (c_{i_{p-1}i_{p+1}} - c_{i_{q-1}i_{p+1}})) + \\ + ((c_{i_{q-1}i_q} - c_{i_pi_q}) - (c_{i_{q-1}i_{p+1}} - c_{i_pi_{p+1}})).$$

Поскольку $i_{q-1} > i_{p-1}$ и $i_p > i_{p+1}$, а также $i_{q-1} < i_p$ и $i_q > i_{p+1}$, то в силу (6.4) получаем $\Delta \geq 0$.

Если $i_{q-1} < i_{p-1}$, то, прибавляя и вычитая значение $c_{i_{p-1}i_q}$, имеем

$$\Delta = ((c_{i_{q-1}i_q} - c_{i_{p-1}i_q}) - (c_{i_{q-1}i_p} - c_{i_{p-1}i_p})) + \\ + ((c_{i_{p-1}i_q} - c_{i_pi_q}) - (c_{i_{p-1}i_{p+1}} - c_{i_pi_{p+1}})).$$

Так как $i_{q-1} < i_{p-1}$ и $i_q > i_p$, а также $i_{p-1} < i_p$ и $i_q > i_{p+1}$, то в силу (6.4) получаем $\Delta \geq 0$.

Следовательно, существует оптимальная перестановка, в которой требования, предшествующие n , упорядочены по неубыванию значений t_{iL} (для любого L , $1 \leq L \leq M$). Аналогично можно показать, что существует оптимальная перестановка, в которой требования, следующие за n , упорядочены по невозрастанию значений t_{iL} (для любого L , $1 \leq L \leq M$).

3) Покажем, что перестановка π^* (из класса пирамидальных перестановок), которой соответствует наименьшее значение $\bar{t}_{\max}(\pi)$, может быть построена за $O(n^2M)$ операций.

Рассмотрим перестановку вида $\pi = (\dots, p, q, \dots)$, $p \in N, q \in N, p \neq q$. Из результатов п. 3.4, 2) следует, что величина c_{pq} содержательно означает разность между моментами начала обслуживания требования p и требования q прибором 1, если каждый прибор обслуживает требования в последовательности π .

Введем в рассмотрение величину δ_{pq} , где

$$\delta_{pq} = c_{pq} + \sum_{L=1}^M t_{qL} - \sum_{L=1}^M t_{pL}, \quad p \in N, q \in N, p \neq q. \quad (6.5)$$

Величина δ_{pq} содержательно означает разность между моментами завершения обслуживания требования p и требования q прибором M , если каждый прибор обслуживает требования в последовательности π .

Обозначим через $f_p^1(q)$ наименьшее значение общего времени обслуживания (без задержек) требований множества $\{p, p+1, \dots, n\}$ при условии, что обслуживание этих требований начинается в момент времени $d = 0$ и

требование q ($q > p$) обслуживается последним. При этом не нарушая общности можно считать, что требование p обслуживается первым.

Обозначим через $f_p^2(q)$ наименьшее значение общего времени обслуживания (без задержек) требований множества $\{p, p+1, \dots, n\}$ при условии, что обслуживание этих требований начинается в момент времени $d = 0$ и требование q ($q > p$) обслуживается первым. При этом не нарушая общности можно считать, что требование p обслуживается последним.

Нетрудно видеть, что имеют место рекуррентные соотношения

$$f_p^1(p+1) = \min\{f_{p+1}^2(q) + c_{pq} \mid q > p+1\},$$

$$f_p^1(q) = f_{p+1}^1(q) + c_{p,p+1}, \quad q = p+2, \dots, n,$$

$$f_p^2(p+1) = \min\{f_{p+1}^1(q) + \delta_{pq} \mid q > p+1\},$$

$$f_p^2(q) = f_{p+1}^2(q) + \delta_{p+1,p}, \quad q = p+2, \dots, n, \quad p = n-2, n-3, \dots, 1,$$

с начальными условиями

$$f_{n-1}^1(n) = c_{n-1,n} + \sum_{L=1}^M t_{nL},$$

$$f_{n-1}^2(n) = c_{n,n-1} + \sum_{L=1}^M t_{nL}.$$

Используя приведенные соотношения, можно построить $2n - 2$ пирамидальных перестановок. В качестве π^* следует выбирать ту из них, которой соответствует $\min\{f_1^1(p), f_1^2(p) \mid p = 2, \dots, n\}$.

Вычисление значений c_{pq} и δ_{pq} , $p = 1, \dots, n$, $q = 1, \dots, n$, $p \neq q$, требует выполнения не более чем $O(n^2M)$ операций, для последующего выбора перестановки π^* требуется не более $O(n^2)$ операций.

4) П р и м е р. Четыре группы экскурсантов желают осмотреть выставку, экспонаты которой размещены в пяти последовательно осматриваемых залах. Пусть t_{iL} означает длительность осмотра i -й группой экспозиции зала L , $i = 1, 2, 3, 4$, $L = 1, 2, 3, 4, 5$. Матрица $\|t_{iL}\|$ имеет вид

$$\left\| \begin{array}{ccccc} 1 & 6 & 3 & 4 & 3 \\ 4 & 8 & 3 & 5 & 5 \\ 5 & 8 & 4 & 6 & 7 \\ 6 & 8 & 6 & 7 & 7 \end{array} \right\|.$$

Будем считать, что единица времени равна пяти минутам.

Матрица $\|t_{iL}\|$ упорядочена по строкам, что можно объяснить различием в численном составе групп и индивидуальными интересами. Требуется таким образом организовать просмотр выставки, чтобы в любой момент времени в каждом зале находилось не более одной группы; все экскурсанты каждой группы одновременно находились в одном и том же зале; переход каждой группы из зала в зал осуществлялся непосредственно после завершения осмотра экспозиции предыдущего зала; все четыре группы экскурсантов осмотрели всю выставку как можно быстрее.

Имеем ситуацию, рассматриваемую в данном пункте. Для построения оптимального расписания воспользуемся алгоритмом из п. 3).

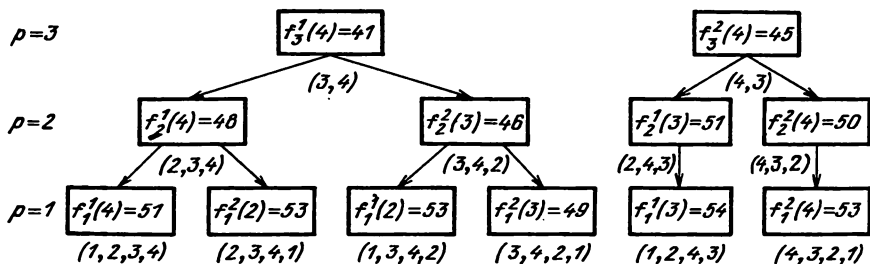


Рис. 6.1

Матрица $\|c_{pq}\|$ имеет вид

$$\begin{vmatrix} - & 3 & 2 & 1 \\ 12 & - & 7 & 6 \\ 17 & 10 & - & 17 \\ 21 & 14 & 11 & - \end{vmatrix},$$

а матрица $\|\delta_{pq}\|$ — вид

$$\begin{vmatrix} - & 12 & 16 & 19 \\ 3 & - & 12 & 15 \\ 3 & 5 & - & 11 \\ 3 & 5 & 7 & - \end{vmatrix}.$$

Процесс дальнейших вычислений удобно изобразить в виде графа (рис. 6.1). В каждой вершине графа указана вычисляемая величина и ее значение. Ниже вершины записана соответствующая последовательность требований (групп). Направления дуг указывают порядок вычисления.

На последнем шаге алгоритма ($p = 1$) получено шесть пирамидальных перестановок. Перестановка (3, 4, 2, 1) определяет оптимальное расписание. Моменты времени, в которые группа покидает зал L (и переходит в зал $L + 1$), $i = 1, 2, 3, 4$, $L = 1, 2, 3, 4$, при условии, что группы осматривают выставку в последовательности (3, 4, 2, 1), приведены в табл. 6.1.

5) Пусть матрица $\|t_{iL}\|$ упорядочена по строкам и $t_{iM} \geq t_{iL}$, $L = 1, \dots, M - 1$, $i \in N$. Покажем, что в этом случае $\pi^* = (1, 2, \dots, n)$.

Таблица 6.1

$L \backslash i$	1	2	3	4	5
3	5	13	17	23	30
4	13	21	27	34	41
2	25	33	36	41	46
1	34	40	42	46	49

Пусть величины δ_{pq} удовлетворяют (6.5). Нетрудно видеть, что для любой перестановки $\pi = (i_1, i_2, \dots, i_n)$.

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} \delta_{i_k i_{k+1}} + \sum_{L=1}^M t_{i_1 L}.$$

Положим

$$\begin{aligned} \delta'_{pq} &= \delta_{pq} - t_{qM} = \\ &= \max_{1 \leq Q \leq M-1} \left\{ \sum_{L=Q}^{M-1} t_{qL} - \sum_{L=Q+1}^M t_{pL}, 0 \right\}, \quad p \in N, \quad q \in N, p \neq q. \end{aligned}$$

Очевидно, $\delta'_{pq} = 0$ при $p > q$. Имеем

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} \delta'_{i_k i_{k+1}} + \sum_{i=1}^n t_{iM} + \sum_{L=1}^M t_{i_1 L}.$$

Положим $\delta'(\pi_0) = \sum_{k=0}^{n-1} \delta'_{i_k i_{k+1}}$, где $\pi_0 = (0, \pi) = (i_0 = 0, i_1, i_2, \dots, i_n)$,

$$\delta'_{i_0 i_k} = \sum_{L=1}^{M-1} t_{i_k L}.$$

Пусть π — пирамидальная перестановка и $i_r = n$. Выберем $i_p, i_q \in N$ такие, что $1 \leq p \leq r, r \leq q \leq n$ и $i_q > i_p > i_{q+1}$. Покажем, что $\delta'(\pi_0) \leq \delta'(\pi'_0)$, где $\pi'_0 = (i_0, \dots, i_{p-1}, i_{p+1}, \dots, i_r, \dots, i_q, i_p, i_{q+1}, \dots, i_n)$.

Имеем

$$\delta'(\pi'_0) - \delta'(\pi_0) = \delta'_{i_{p-1} i_{p+1}} + \delta'_{i_q i_p} + \delta'_{i_p i_{q+1}} - \delta'_{i_{p-1} i_p} - \delta'_{i_p i_{p+1}} - \delta'_{i_q i_{q+1}}.$$

Очевидно, $i_{p-1} < i_p < i_{p+1}$, $\delta'_{i_q i_p} = \delta'_{i_p i_{q+1}} = \delta'_{i_q i_{q+1}} = 0$.

Покажем, что $\delta'_{i_{p-1} i_{p+1}} \geq \delta'_{i_{p-1} i_p} + \delta'_{i_p i_{p+1}}$.

Пусть $p = 1$ и $\delta'_{i_1 i_2} = \sum_{L=Q}^{M-1} t_{i_2 L} - \sum_{L=Q+1}^M t_{i_1 L} > 0$ (в случае $\delta'_{i_1 i_2} = 0$,

очевидно, имеем $\delta'_{i_0 i_2} = \sum_{L=1}^{M-1} t_{i_2 L} \geq \sum_{L=1}^{M-1} t_{i_1 L} = \delta'_{i_0 i_1}$).

Имеем

$$\begin{aligned} \delta'_{i_0 i_1} + \delta'_{i_1 i_2} &= \sum_{L=1}^{M-1} t_{i_1 L} + \sum_{L=Q}^{M-1} t_{i_2 L} - \sum_{L=Q+1}^M t_{i_1 L} = \\ &= \sum_{L=1}^Q t_{i_1 L} - t_{i_1 M} + \sum_{L=Q}^{M-1} t_{i_2 L} \leq \sum_{L=1}^{M-1} t_{i_2 L} = \delta'_{i_0 i_2}. \end{aligned}$$

Пусть $p > 1$. Предположим, что

$$\delta'_{i_p i_{p+1}} = \sum_{L=H}^{M-1} t_{i_{p+1} L} - \sum_{L=H+1}^M t_{i_p L} > 0,$$

$$\delta'_{i_{p-1} i_p} = \sum_{L=Q}^{M-1} t_{i_p L} - \sum_{L=Q+1}^M t_{i_{p-1} L} > 0$$

и $Q > H$ (в остальных случаях доказательство аналогичное).

Имеем

$$\begin{aligned} \delta'_{ip-1ip} + \delta'_{ipip+1} &= \sum_{L=Q}^{M-1} t_{ipL} - \sum_{L=Q+1}^M t_{ip-1L} + \sum_{L=H}^{M-1} t_{ip+1L} - \\ &- \sum_{L=H+1}^M t_{ipL} = \left(\sum_{L=H}^{M-1} t_{ip+1L} - \sum_{L=H+1}^M t_{ip-1L} \right) + \sum_{L=H+1}^Q t_{ip-1L} - \\ &- \sum_{L=H+1}^{Q-1} t_{ipL} - t_{ipM} \leq \delta'_{ip-1ip+1} + \sum_{L=H+1}^Q (t_{ip-1L} - t_{ipL}) \leq \delta'_{ip-1ip+1}. \end{aligned}$$

Таким образом, перестановке $\pi_0^* = (0, 1, 2, \dots, n)$ соответствует наименьшее значение $\delta'(\pi_0)$. Следовательно, перестановка $\pi^* = (1, 2, \dots, n)$ является искомой. Она может быть построена за $O(n \log n)$ операций путем упорядочения требований по неубыванию значений t_{iL} , $i \in N$, $1 \leq L \leq M$.

Если матрица $\|t_{iL}\|$ упорядочена по строкам и $t_{i1} \geq t_{iL}$, $i \in N$, $L = 2, \dots, M$, то рассуждая аналогичным образом, можно показать, что оптимальной является перестановка $\pi^* = (n, n-1, \dots, 1)$.

6.4. В заключение этого параграфа покажем, что перестановка элементов множества N , которой соответствует наименьшее значение $\sum_{i=1}^n \bar{t}_i(\pi)$, может быть построена за $O(n \log n)$ операций. Предполагается, что задержки в обслуживании каждого требования запрещены и матрица $\|t_{iL}\|$ упорядочена по строкам. Пронумеруем требования по неубыванию значений t_{iL} , $i \in N$, $1 \leq L \leq M$. Покажем, что перестановка $\pi^* = (1, 2, \dots, n)$ является искомой.

Пусть $\pi = (i_1, i_2, \dots, i_n)$ — произвольная перестановка элементов множества $N = \{1, 2, \dots, n\}$. Из п. 3.4. § 3 этой главы следует, что

$$\sum_{i \in N} \bar{t}_i(\pi) = \sum_{k=1}^{n-1} (n-k) c_{ik} i_{k+1} + \sum_{i=1}^n \sum_{L=1}^M t_{iL}.$$

1) Пусть $t_{iM} = \max\{t_{iL} \mid L = 1, \dots, M\}$, $i \in N$. Из п. 6.3, 5) следует, что для любой последовательности $\pi = (i_1, i_2, \dots, i_n)$ справедливо $\bar{t}_{kM}(\pi^*) \leq \bar{t}_{kM}(\pi)$. Таким образом,

$$\sum_{i \in N} \bar{t}_i(\pi^*) = \sum_{k=1}^n \bar{t}_{kM}(\pi^*) \leq \sum_{k=1}^n t_{ikM}(\pi) = \sum_{i \in N} \bar{t}_i(\pi)$$

или, что то же, самое,

$$\sum_{k=1}^{n-1} (n-k) c_{k,k+1} \leq \sum_{k=1}^{n-1} (n-k) c_{ik} i_{k+1}.$$

2) Пусть $t_{iR} = \max\{t_{iL} \mid L = 1, \dots, M\}$, $i \in N$, $1 \leq R \leq M$. Очевидно,

$$\begin{aligned} c_{pq} &= \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \right\} = \\ &= \max_{1 \leq Q \leq R} \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \right\}, \quad q > p. \end{aligned}$$

Рассмотрим задачу нахождения последовательности обслуживания требований, которой соответствует наименьшее значение $\sum_{i \in N} \bar{t}_{iR}(\pi)$. Нетрудно видеть, что искомой последовательности соответствует наименьшее значение $\sum_{k=1}^{n-1} (n-k) c'_{ik^{i_{k+1}}}$, где

$$c'_{pq} = \max_{1 \leq Q \leq R} \left\{ \sum_{L=1}^Q t_{pL} - \sum_{L=1}^{Q-1} t_{qL} \right\}.$$

Как показано выше, имеет место соотношение

$$\sum_{k=1}^{n-1} (n-k) c'_{k,k+1} \leq \sum_{k=1}^{n-1} (n-k) c'_{ik^{i_{k+1}}}.$$

Поскольку $c'_{k,k+1} = c_{k,k+1}$ и $c_{pq} \geq c'_{pq}$, $p > q$, то

$$\sum_{k=1}^{n-1} (n-k) c_{k,k+1} \leq \sum_{k=1}^{n-1} (n-k) c_{ik^{i_{k+1}}}.$$

Таким образом, перестановке $\pi^* = (1, 2, \dots, n)$ соответствует наименьшее значение $\sum_{i \in N} \bar{t}_i(\pi)$. Она может быть построена за $O(n \log n)$ операций путем упорядочения требований по неубыванию значений t_{iL} , $i \in N$, $1 \leq L \leq M$.

§ 7. Доминантные матрицы длительностей

В систему поточного типа, состоящую из M последовательных приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительность обслуживания требования $i \in N$ прибором L , $1 \leq L \leq M$, равна $t_{iL} \geq 0$. Запись $t_{iL} = 0$ означает, что требование i прибором L обслуживается, но длительностью этого обслуживания можно пренебречь. Прерывания в процессе обслуживания каждого требования каждым прибором не допускаются. Оптимальное расписание ищется в классе расписаний, при которых каждый прибор обслуживает требования в одной и той же последовательности.

В п. 7.1–7.4 рассматриваются задачи построения оптимальных последовательностей π^* обслуживания требований, которым соответствуют наименьшие значения общего $\bar{t}_{\max}(\pi)$ или суммарного взвешенного $\sum_{i \in N} w_i \bar{t}_i(\pi)$

времени обслуживания. В п. 7.5–7.7 эти же задачи решаются в предположении, что задержки в обслуживании каждого требования не допускаются. Предполагается, что матрица $\|t_{iL}\|$ обладает определенными свойствами.

Матрицу $\|t_{iL}\|$ длительностей обслуживания будем называть *доминантной* (Q -доминантной), если для некоторого Q , $1 \leq Q \leq M$, имеет место соотношение

$$\min \left\{ \sum_{L=H+1}^Q t_{iL} - \sum_{L=H}^{Q-1} t_{jL}, \sum_{L=Q}^R t_{iL} - \sum_{L=Q+1}^{R+1} t_{jL} \right\} \geq 0,$$

где $H = 1, \dots, Q-1$ ($Q \neq 1$); $R = Q, \dots, M-1$ ($Q \neq M$); $i, j \in N$, $i \neq j$.

Так, например, матрица

$$\begin{vmatrix} 6 & 4 & 10 & 1 \\ 7 & 5 & 8 & 4 \\ 3 & 7 & 6 & 2 \end{vmatrix}$$

является 3-доминантной.

7.1. Рассмотрим задачу построения последовательности π^* обслуживания требований, которой соответствует оптимальное по быстродействию расписание. Предполагается, что матрица $\|t_{iL}\|$ является доминантной.

Для заданных последовательностей $U = (u_1, u_2, \dots, u_{M-1})$ целых чисел таких, что $1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n$, и перестановки $\pi = (i_1, i_2, \dots, i_n)$ элементов множества N положим

$$f(U, \pi) = \sum_{k=1}^{u_1} t_{i_{k1}} + \sum_{k=u_1}^{u_2} t_{i_{k2}} + \dots + \sum_{k=u_{M-1}}^n t_{i_{kM}}. \quad (7.1)$$

Напомним, что

$$\bar{t}_{\max}(\pi) = \max_{1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n} \{f(U, \pi)\}. \quad (7.2)$$

1) Покажем, что

$$\bar{t}_{\max}(\pi) = \sum_{L=1}^{M-1} t_{i_1 L} + \sum_{k=1}^n t_{i_{kM}} \quad (7.3)$$

для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда матрица $\|t_{iL}\|$ является M -доминантной.

а) Действительно, если для всех перестановок π выполняется соотношение (7.3), то для любой последовательности $U = (u_1, u_2, \dots, u_{M-1})$ целых чисел таких, что $1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n$, справедливо

$$\sum_{L=1}^{M-1} t_{i_1 L} + \sum_{k=1}^n t_{i_{kM}} \geq f(U, \pi).$$

В частности, для всех H , $1 \leq H \leq M-1$, имеем

$$\sum_{L=1}^{M-1} t_{i_1 L} + \sum_{k=1}^n t_{i_{kM}} \geq \sum_{L=1}^H t_{i_1 L} + \sum_{L=H}^{M-1} t_{i_2 L} + \sum_{k=2}^n t_{i_{kM}},$$

откуда следует, что

$$\sum_{L=H+1}^M t_{i_1 L} \geq \sum_{L=H}^{M-1} t_{i_2 L}.$$

В силу произвольности перестановки π заключаем, что матрица $\|t_{iL}\|$ является M -доминантной.

б) Пусть матрица $\|t_{iL}\|$ является M -доминантной. Покажем, что для любой последовательности $U = (u_1, u_2, \dots, u_{M-1})$ целых чисел таких, что $1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n$, и любой перестановки $\pi = (i_1, i_2, \dots, i_n)$ справедливо

$$f(U, \pi) \leq \sum_{L=1}^{M-1} t_{i_1 L} + \sum_{k=1}^n t_{i_{kM}}. \quad (7.4)$$

Положим $p = u_{M-1}$, $R = \min \{ L \mid L = 1, \dots, M, u_L = p \}$ и

$$q = \begin{cases} u_{R-1}, & R > 1 \\ 1, & R = 1 \end{cases}$$

Определим

$$F(p, q, R, \pi) = \sum_{k=q}^{p-1} t_{ikR} + \sum_{L=R}^{M-1} t_{ipL} + \sum_{k=p}^n t_{ikM}.$$

Так как матрица $\|t_{iL}\|$ является M -доминантной, то, в частности,

$$\sum_{L=R}^{M-1} t_{ikL} \leq \sum_{L=R+1}^M t_{ik-1L}, \quad k = p, p-1, \dots, q+1.$$

Тогда

$$F(p, q, R, \pi) \leq \sum_{L=R}^{M-1} t_{iqL} + \sum_{k=q}^n t_{ikM},$$

и следовательно,

$$\begin{aligned} f(U, \pi) &\leq \sum_{k=1}^{u_1} t_{ik1} + \sum_{k=u_1}^{u_2} t_{ik2} + \dots \\ &\dots + \sum_{k=u_{R-2}}^q t_{ik, R-1} + \sum_{L=R}^{M-1} t_{iqL} + \sum_{k=q}^n t_{ikM}. \end{aligned}$$

Положим $u_R = u_{R+1} = \dots = u_{M-1} = q$, а остальные u_L оставим без изменения. Повторяя описанные выше рассуждения конечное число раз, получаем неравенство (7.4).

Таким образом, если матрица $\|t_{iL}\|$ является M -доминантной, то оптимальной является любая перестановка вида $(j, \dots)$, где

$$\sum_{L=1}^{M-1} t_{jL} = \min \left\{ \sum_{L=1}^{M-1} t_{iL} \mid i = 1, \dots, n \right\}.$$

Элемент $j \in N$ может быть найден за $O(nM)$ операций.

Рассуждая аналогичным образом, можно показать, что

$$\bar{r}_{\max}(\pi) = \sum_{k=1}^n t_{ik1} + \sum_{L=2}^M t_{inL}$$

для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда матрица $\|t_{iL}\|$ является 1-доминантной.

В этом случае оптимальной является любая перестановка вида $(\dots, j)$, где

$$\sum_{L=2}^M t_{jL} = \min \left\{ \sum_{L=2}^M t_{iL} \mid i = 1, \dots, n \right\}.$$

Элемент $j \in N$ может быть найден за $O(nM)$ операций.

2) Покажем, что при некотором Q , $2 \leq Q \leq M-1$,

$$\bar{r}_{\max}(\pi) = \sum_{L=1}^{Q-1} t_{i_1L} + \sum_{k=1}^n t_{ikQ} + \sum_{L=Q+1}^M t_{inL} \quad (7.5)$$

для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда матрица $\|t_{iL}\|$ является Q -доминантной.

Пусть $T_{RH}(\pi)$ означает общее время обслуживания требований множества N системой поточного типа, состоящей из приборов $R, R+1, \dots, H$, при условии, что каждый прибор обслуживает требования в последовательности π и процесс обслуживания начинается в момент времени $d=0$.

Очевидно, соотношение (7.5) имеет место для всех π тогда и только тогда, когда для всех π

$$\bar{t}_{1Q}(\pi) = \sum_{L=1}^{Q-1} t_{i_1 L} + \sum_{k=1}^n t_{i_k Q}; \quad \bar{t}_{QM}(\pi) = \sum_{k=1}^n t_{i_k Q} + \sum_{L=Q+1}^M t_{i_n L}.$$

Как следует из п. 1), это эквивалентно утверждению, что матрица $\|t_{iL}\|$ является Q -доминантной.

В рассматриваемом случае оптимальной является любая перестановка вида $(j_1, \dots, j_n)$, где

$$\begin{aligned} & \sum_{L=1}^{Q-1} t_{j_1 L} + \sum_{L=Q+1}^M t_{j_2 L} = \\ & = \min \left\{ \sum_{L=1}^{Q-1} t_{iL} + \sum_{L=Q+1}^M t_{kL} \mid i=1, \dots, n, k=1, \dots, n, i \neq k \right\}. \end{aligned}$$

Нахождение элементов $j_1, j_2 \in N$ можно осуществить за $O(nM)$ операций.

7.2. Заменяем исходную систему, состоящую из M приборов, двухстадийной системой поточного типа, состоящей из приборов A и B . Каждое требование $i \in N$ обслуживается прибором A в течение a_i единиц времени, затем прибором B в течение b_i единиц времени, где

$$a_i = \sum_{L=1}^{M-1} t_{iL}, \quad b_i = \sum_{L=2}^M t_{iL}, \quad i \in N.$$

В этом пункте приводятся некоторые условия, накладываемые на матрицу $\|t_{iL}\|$, достаточные для того, чтобы в исходной и во вновь построенной системах оптимальное (по быстродействию) расписание определялось одной и той же перестановкой π^* .

1) Пусть $\pi^* = (j_1, j_2, \dots, j_n)$ — оптимальная последовательность обслуживания требований в построенной системе. Покажем, что π^* является оптимальной последовательностью и для исходной системы, если

$$\bar{t}_{\max}(\pi^*) = \max_{1 \leq p \leq n} \left\{ \sum_{k=1}^{p-1} t_{j_k 1} + \sum_{L=1}^M t_{j_p L} + \sum_{k=p+1}^n t_{j_k M} \right\}.$$

Обозначим через $\hat{t}_{\max}(\pi)$ общее время обслуживания требований множества N в системе, состоящей из приборов A и B , при условии, что каждый прибор обслуживает требования в последовательности $\pi = (i_1, i_2, \dots, i_n)$. Нетрудно видеть, что

$$\hat{t}_{\max}(\pi) = \max_{1 \leq p \leq n} \left\{ \sum_{k=1}^{p-1} t_{i_k 1} + \sum_{L=1}^M t_{i_p L} + \sum_{k=p+1}^n t_{i_k M} \right\} + \sum_{L=2}^{M-1} \sum_{i=1}^n t_{iL}.$$

Тогда

$$\bar{t}_{\max}(\pi) \geq \hat{t}_{\max}(\pi) - \sum_{L=2}^{M-1} \sum_{i=1}^n t_{iL} \geq \hat{t}_{\max}(\pi^*) - \sum_{L=2}^{M-1} \sum_{i=1}^n t_{iL} = \bar{t}_{\max}(\pi^*).$$

2) Предположим, что при некотором Q , $1 \leq Q \leq M-1$, матрица $\|t_{iL}\|$ удовлетворяет условию

$$\min \left\{ \sum_{L=1}^H t_{iL} - \sum_{L=2}^{H+1} t_{jL}, \sum_{L=R+1}^M t_{iL} - \sum_{L=R}^{M-1} t_{jL} \right\} \geq 0, \quad (7.6)$$

$$H=1, \dots, Q-1 \quad (Q \neq 1); \quad R=Q+1, \dots, M-1 \quad (Q \neq M-1);$$

$$i, j \in \bar{N}, \quad i \neq j.$$

Условие (7.6) допускает интерпретацию в терминах доминантных матриц. Действительно, сопоставим матрице $\|t_{iL}\|$ пару матриц T_1 и T_2 , состоящих соответственно из Q первых и $M-Q$ последних столбцов матрицы $\|t_{iL}\|$. В матрице T_2 сохраним нумерацию столбцов, принятую в матрице $\|t_{iL}\|$. Условие (7.6) означает, во-первых, что матрица T_1 либо является 1-доминантной, либо состоит из одного столбца с номером $Q=1$, а во-вторых, что матрица T_2 либо является M -доминантной, либо состоит из одного столбца с номером $M=Q+1$.

Так, например, матрица

$$\left\| \begin{array}{ccc|c} 10 & 5 & 6 & 7 \\ 8 & 9 & 4 & 10 \\ 11 & 7 & 7 & 12 \end{array} \right\|$$

удовлетворяет условию (7.6) при $Q=3$, а матрица

$$\left\| \begin{array}{cc|cc} 10 & 3 & 4 & 16 \\ 8 & 7 & 15 & 8 \\ 11 & 1 & 7 & 20 \end{array} \right\|$$

удовлетворяет этому условию при $Q=2$ (здесь вертикальная черта разделяет матрицы T_1 и T_2).

Пусть $U = (u_1, u_2, \dots, u_{M-1})$ — последовательность целых чисел таких, что $1 \leq u_1 \leq u_2 \leq \dots \leq u_{Q-1} \leq u_Q = p \leq u_{Q+1} \leq \dots \leq u_{M-1} \leq n$. Покажем, что в случае, когда справедливо соотношение (7.6), неравенство

$$f(U, \pi) \leq \sum_{k=1}^{p-1} t_{ik1} + \sum_{L=1}^M t_{ipL} + \sum_{k=p+1}^n t_{ikM}$$

имеет место для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$.

Доказательство проведем для $2 \leq Q \leq M-2$ (в случае $Q=1$ или $Q=M-1$ доказательство аналогичное).

Из результатов п. 7.1 и из (7.6) следует, что

$$\sum_{k=1}^{p-1} t_{ik1} + \sum_{k=1}^n t_{ipL} \geq \sum_{k=1}^{u_1} t_{ik1} + \sum_{k=u_1}^{u_2} t_{ik2} + \dots + \sum_{k=u_{Q-1}}^p t_{ikQ},$$

$$\sum_{L=Q+1}^M t_{ipL} + \sum_{k=p+1}^n t_{ikM} \geq \sum_{k=p}^{u_{Q+1}} t_{ik, Q+1} + \dots + \sum_{k=u_{M-1}}^n t_{ikM}$$

Складывая эти неравенства, получаем требуемое соотношение.

3) Покажем, что для любой перестановки $\pi = (i_1, i_2, \dots, i_n)$

$$\bar{t}_{\max}(\pi) = \max_{1 \leq p \leq n} \left\{ \sum_{k=1}^{p-1} t_{ik1} + \sum_{L=1}^M t_{ipL} + \sum_{k=p+1}^n t_{ikM} \right\}, \quad (7.7)$$

если существует такое $\alpha \in [0, 1]$, что

$$\alpha \left(\sum_{L=1}^{Q-1} t_{iL} - \sum_{L=2}^Q t_{jL} \right) + (1-\alpha) \left(\sum_{L=Q+1}^H t_{jL} - \sum_{L=Q}^{H-1} t_{iL} \right) \geq 0,$$

$$Q = 2, \dots, M, H = Q+1, \dots, M. \quad (7.8)$$

Предположим, что для некоторой перестановки π значение $\bar{t}_{\max}(\pi) \leq f(U, \pi)$, причем не все u_L в последовательности $U = (u_1, u_2, \dots, u_{M-1})$ равны между собой, $L = 1, \dots, M-1$. Положим $p = u_1$, $Q = \max\{L \mid L = 1, \dots, M, u_{L-1} = p\}$, $q = u_Q$, $H = \max\{L \mid L = Q, \dots, M, u_{L-1} = q\}$.

По условию

$$\begin{aligned} & \sum_{k=1}^{p-1} t_{ik1} + \sum_{L=1}^{Q-1} t_{ipL} + \sum_{k=p}^{q-1} t_{ikQ} + \sum_{L=Q}^H t_{iqL} > \\ & > \max \left\{ \sum_{k=1}^{p-1} t_{ik1} + \sum_{L=1}^H t_{ipL} + \sum_{k=p+1}^q t_{ikH}, \sum_{k=1}^{q-1} t_{ik1} + \sum_{L=1}^H t_{iqL} \right\}. \end{aligned}$$

Другими словами, имеет место система неравенств

$$\sum_{k=p+1}^{q-1} t_{ikQ} + \sum_{L=Q}^{H-1} t_{iqL} > \sum_{L=Q+1}^H t_{ipL} + \sum_{k=p+1}^{q-1} t_{ikH},$$

$$\sum_{L=2}^Q t_{ipL} + \sum_{k=p+1}^{q-1} t_{ikQ} > \sum_{k=p+1}^{q-1} t_{ik1} + \sum_{L=1}^{Q-1} t_{iqL},$$

которая равносильна системе неравенств

$$\sum_{k=p+1}^q \sum_{L=Q}^{H-1} t_{ikL} > \sum_{k=p}^{q-1} \sum_{L=Q+1}^H t_{ikL}, \quad (7.9)$$

$$\sum_{k=p}^{q-1} \sum_{L=2}^Q t_{ikL} > \sum_{k=p+1}^q \sum_{L=1}^{Q-1} t_{ikL}.$$

С другой стороны, из (7.8) следует, что

$$\begin{aligned} & \alpha \left(\sum_{L=1}^{Q-1} t_{ikL} - \sum_{L=2}^Q t_{ik-1L} \right) + \\ & + (1-\alpha) \left(\sum_{L=Q+1}^H t_{ik-1L} - \sum_{L=Q}^{H-1} t_{ikL} \right) \geq 0, \end{aligned}$$

$$k = p+1, \dots, q.$$

Суммируя эти неравенства, получаем соотношение

$$\alpha \left(\sum_{k=p+1}^q \sum_{L=1}^{Q-1} t_{i_k L} - \sum_{k=p}^{q-1} \sum_{L=2}^Q t_{i_k L} \right) + \\ + (1-\alpha) \left(\sum_{k=p}^{q-1} \sum_{L=Q+1}^H t_{i_k L} - \sum_{k=p+1}^q \sum_{L=Q}^{H-1} t_{i_k L} \right) \geq 0,$$

которое противоречит системе неравенств (7.9).

Положим $u_Q = u_{Q+1} = \dots = u_H = p$, если

$$\sum_{k=1}^{p-1} t_{i_k 1} + \sum_{L=1}^H t_{i_p L} + \sum_{k=p+1}^q t_{i_k H} > \sum_{k=1}^{q-1} t_{i_k 1} + \sum_{L=1}^H t_{i_q L},$$

и $u_1 = u_2 = \dots = u_{Q-1} = q$ — в противном случае; остальные u_L оставим без изменения. Повторяя описанные выше рассуждения конечное число раз, получаем соотношение (7.7).

Например, матрица

$$\begin{vmatrix} 10 & 7 & 12 & 14 \\ 6 & 9 & 6 & 21 \\ 12 & 8 & 15 & 23 \end{vmatrix}$$

удовлетворяет (7.8) при $\alpha \in [1/2, 2/3]$. Для любой перестановки требований имеет место равенство (7.7), причем для перестановки (1, 3, 2) максимум в (7.7) достигается при $p = 2$, а для всех остальных перестановок — при $p = 1$.

Таким образом, условия, приведенные в п. 2) и 3), являются достаточными для выполнения соотношения (7.7) при всех перестановках π . Следовательно, согласно п. 1) оптимальная перестановка π^* может быть найдена при помощи алгоритма из п. 1.3. § 1 этой главы. Трудоемкость ее построения не превосходит $O(nM + n \log n)$ операций.

7.3. Предположим, что при некотором Q , $1 \leq Q \leq M-1$, матрица $\|t_{iL}\|$ удовлетворяет условию

$$\min \left\{ \sum_{L=H+1}^Q t_{iL} - \sum_{L=H}^{Q-1} t_{jL}, \sum_{L=Q+1}^R t_{iL} - \sum_{L=Q+2}^{R+1} t_{jL} \right\} \geq 0, \quad (7.10)$$

$$H = 1, \dots, Q-1 \quad (Q \neq 1); \quad R = Q+1, \dots, M-1 \quad (Q \neq M-1);$$

$$i, j \in N, \quad i \neq j.$$

Условие (7.10) так же, как и условие (7.6), допускает интерпретацию в терминах доминантных матриц. Пусть матрицы T_1 и T_2 те же, что и в п. 7.2 б). Условие (7.10) означает, во-первых, что матрица T_1 либо является Q -доминантной, либо состоит из одного столбца с номером $Q = 1$, а во-вторых, что матрица T_2 либо является $(Q+1)$ -доминантной, либо состоит из одного столбца с номером $M = Q+1$.

Оказывается, что при некотором Q , $1 \leq Q \leq M-1$,

$$\begin{aligned} \bar{t}_{\max}(\pi) = & \sum_{L=1}^{Q-1} t_{i_1 L} + \max_{1 \leq u \leq n} \left\{ \sum_{k=1}^u t_{i_k Q} + \right. \\ & \left. + \sum_{k=u}^n t_{i_k, Q+1} \right\} + \sum_{L=Q+2}^M t_{i_n L} \end{aligned} \quad (7.11)$$

для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда справедливо условие (7.10).

Ограничимся доказательством достаточности этого утверждения. Будем считать, что $2 \leq Q \leq M-2$ (в случае $Q=1$ или $Q=M-1$ доказательство аналогичное). Выберем произвольную перестановку π и рассмотрим такую последовательность $U = (u_1, u_2, \dots, u_{M-1})$ целых чисел, что $1 \leq u_1 \leq u_2 \leq \dots \leq u_{M-1} \leq n$.

Из (7.10) и п. 7.1 следует, что

$$\begin{aligned} \sum_{k=1}^{u_1} t_{i_k 1} + \sum_{k=u_1}^{u_2} t_{i_k 2} + \dots + \sum_{k=u_{Q-1}}^{u_Q} t_{i_k Q} & \leq \sum_{L=1}^{Q-1} t_{i_1 L} + \sum_{k=1}^{u_Q} t_{i_k Q}, \\ \sum_{k=u_Q}^{u_{Q+1}} t_{i_k, Q+1} + \sum_{k=u_{Q+1}}^{u_{Q+2}} t_{i_k, Q+2} + \dots + \sum_{k=u_{M-1}}^n t_{i_k M} & \leq \\ & \leq \sum_{k=u_Q}^n t_{i_k, Q+1} + \sum_{L=Q+2}^M t_{i_n L}. \end{aligned}$$

Складывая эти неравенства, получаем

$$f(U, \pi) \leq \sum_{L=1}^{Q-1} t_{i_1 L} + \sum_{k=1}^{u_Q} t_{i_k Q} + \sum_{k=u_Q}^n t_{i_k, Q+1} + \sum_{L=Q+2}^M t_{i_n L}.$$

В силу произвольности последовательности U заключаем, что имеет место соотношение (7.11).

Оптимальная перестановка может быть построена следующим образом. Найдем перестановку π' , которой соответствует наименьшее общее время обслуживания требований множества N системой поточного типа, состоящей из двух приборов Q и $Q+1$, в предположении, что процесс обслуживания начинается в момент времени $d=0$.

Если $Q=1$ ($Q=M-1$), то, используя (7.11), вычислим наименьшее общее время обслуживания требований при условии, что требование j , $j=1, \dots, n$ обслуживается последним (соответственно первым), а остальные требования обслуживаются в последовательности, задаваемой перестановкой π' . В качестве искомой следует выбрать ту из рассмотренных n перестановок, которой соответствует наименьшее значение $\bar{t}_{\max}(\pi)$. Трудоемкость построения π^* не превосходит $O(nM + n^2)$ операций.

Если $2 \leq Q \leq M-2$, то рассмотрим $n(n-1)/2$ последовательностей, в которых требование i обслуживается первым, требование j — последним, а остальные требования — согласно перестановке π' , $i=1, \dots, n$, $j=1, \dots, n$, $i \neq j$. В качестве искомой выбирается перестановка с наимень-

шим значением $\bar{t}_{\max}(\pi)$. Трудоемкость ее построения не превосходит $O(nM + n^3)$ операций.

7.4. П р и м е р. Рассмотрим задачу построения оптимального по быстродействию расписания обслуживания четырех требований в системе поточного типа из пяти приборов. Предполагается, что обслуживание начинается в момент времени $d = 0$, все приборы обслуживают требования в одной и той же последовательности, а матрица $\|t_{iL}\|$ имеет вид

$$\begin{vmatrix} 1 & 6 & 6 & 3 & 1 \\ 5 & 7 & 4 & 10 & 2 \\ 6 & 5 & 9 & 2 & 3 \\ 2 & 8 & 7 & 8 & 1 \end{vmatrix}.$$

Нетрудно проверить, что эта матрица удовлетворяет условию (7.10) при $Q = 3$. Для нахождения перестановки π^* , определяющей оптимальное расписание, воспользуемся подходом, описанным в п. 7.3.

При помощи алгоритма из п. 1.3 §1 этой главы отыщем перестановку π' , которой соответствует наименьшее общее время обслуживания четырех требований в системе, состоящей из приборов 3 и 4. Имеем $\pi' = (2, 4, 1, 3)$.

Результаты последующих вычислений приведены в табл. 7.1.

Т а б л и ц а 7.1

π	$\bar{t}_{\max}(\pi)$	π	$\bar{t}_{\max}(\pi)$
(1, 4, 3, 2)	58	(2, 4, 1, 3)	49
(2, 4, 3, 1)	49	(3, 4, 1, 2)	69
(1, 2, 4, 3)	48	(2, 1, 3, 4)	59
(3, 2, 4, 1)	58	(4, 1, 3, 2)	65
(1, 2, 3, 4)	56	(3, 2, 1, 4)	63
(4, 2, 3, 1)	50	(4, 2, 1, 3)	50

В качестве π^* выбирается перестановка (1, 2, 4, 3), $\bar{t}_{\max}(\pi^*) = 48$.

7.5. Рассмотрим задачу нахождения перестановки π^* , которой соответствует наименьшее значение суммарного взвешенного времени

$$\sum_{k=1}^n w_{ik} \bar{t}_{ik}(\pi) \text{ обслуживания.}$$

Покажем, что при некотором Q , $1 \leq Q \leq M$,

$$\sum_{k=1}^n w_{ik} \bar{t}_{ik}(\pi) = \sum_{k=1}^n w_{ik} \left(\sum_{L=1}^{Q-1} t_{iL} + \sum_{j=1}^k t_{ijQ} + \sum_{L=Q+1}^M t_{ikL} \right) \quad (7.12)$$

для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда матрица $\|t_{iL}\|$ длительностей обслуживания является Q -доминантной.

Действительно, соотношение (7.12) имеет место для всех перестановок π тогда и только тогда, когда для всех π

$$\bar{t}_{ik}(\pi) = \sum_{L=1}^{Q-1} t_{iL} + \sum_{j=1}^k t_{ijQ} + \sum_{L=Q+1}^M t_{ikL}, \quad k=1, \dots, n.$$

Как следует из п. 7.1, последнее утверждение эквивалентно тому, что матрица $\|t_{iL}\|$ является Q -доминантной.

Оптимальная перестановка π^* может быть построена следующим образом. Найдем перестановку π' , которой соответствует наименьшее значение

$\sum_{k=1}^n w_{ik} \sum_{j=1}^k \bar{t}_{ij} Q$. Для этого достаточно упорядочить требования по не-
возрастанию значений $w_i/t_{iQ}, i \in N$.

Если $Q = 1$, то $\pi^* = \pi'$. Если $Q > 1$, то π^* содержится среди n перестановок, в которых элемент $j, j = 1, \dots, n$, стоит на первом месте, а остальные элементы упорядочены согласно перестановке π' . Используя (7.12), для каждой из них можно вычислить значение суммарного взвешенного времени обслуживания и выбрать наилучшую перестановку π^* . Трудоемкость ее построения не превосходит $O(nM + n^2)$ операций.

7.6. П р и м е р. Рассмотрим задачу построения расписания обслуживания четырех требований в системе поточного типа из пяти приборов, при котором суммарное взвешенное время обслуживания $\sum_{i \in N} w_i \bar{t}_i(\pi)$ ми-

нимально. Предполагается, что обслуживание начинается в момент времени $d = 0$; все приборы обслуживают требования в одной и той же последовательности; $w_1 = w_2 = 2, w_3 = 5, w_4 = 4$, а матрица $\|t_{iL}\|$ имеет вид

$$\begin{vmatrix} 1 & 6 & 6 & 3 & 7 \\ 5 & 7 & 4 & 6 & 2 \\ 6 & 5 & 9 & 4 & 5 \\ 2 & 8 & 7 & 3 & 5 \end{vmatrix}.$$

Нетрудно проверить, что эта матрица является 3-доминантной.

Для нахождения перестановки π^* , определяющей искомое оптимальное расписание, воспользуемся подходом, описанным в п. 7.5.

Упорядочим требования по невозрастанию значений $w_i/t_{i3}, i = 1, 2, 3, 4$. Получим перестановку $\pi' = (4, 3, 2, 1)$. Результаты последующих вычислений приведены в табл. 7.2.

Т а б л и ц а 7.2

π	$\sum_{i \in N} w_i \bar{t}_i(\pi)$	π	$\sum_{i \in N} w_i \bar{t}_i(\pi)$
(4, 3, 2, 1)	443	(2, 4, 3, 1)	473
(3, 4, 2, 1)	457	(1, 4, 3, 2)	430

В качестве π^* выбирается перестановка (1, 4, 3, 2).

7.7. Рассмотрим задачу построения перестановки π^* , которой соответствует наименьшее значение общего времени обслуживания $\bar{t}_{\max}(\pi)$ при условии, что задержки в обслуживании каждого требования запрещены.

1) Как следует из соотношения (3.2) § 3 этой главы, для любой перестановки $\pi = (i_1, i_2, \dots, i_n)$

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \sum_{k=1}^{n-1} \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{i_k L} - \sum_{L=1}^{Q-1} t_{i_{k+1} L} \right\} + \sum_{k=1}^n t_{i_n L} = \\ &= \sum_{k=1}^{n-1} \max_{2 \leq Q \leq M} \left\{ \max \left\{ \sum_{L=2}^Q t_{i_k L} - \sum_{L=1}^{Q-1} t_{i_{k+1} L}, 0 \right\} \right\} + \\ &+ \sum_{k=1}^{n-1} t_{i_k 1} + \sum_{L=1}^M t_{i_n L}. \end{aligned} \quad (7.13)$$

Пусть $t^k(\pi)$ означает момент завершения обслуживания (возможно, с задержками) двух требований i_k и i_{k+1} в рассматриваемой системе при условии, что каждый прибор сначала обслуживает требование i_k , а затем требование i_{k+1} , и обслуживание требования i_k прибором 1 начинается в момент времени $d = 0$, $k = 1, \dots, n-1$.

Очевидно,

$$t^k(\pi) = \max_{1 \leq Q \leq M} \left\{ \sum_{L=1}^Q t_{i_k L} + \sum_{L=Q}^M t_{i_{k+1} L} \right\}, \quad k = 1, \dots, n-1.$$

Преобразуя последнее соотношение, имеем

$$\begin{aligned} t^k(\pi) &= \max_{2 \leq Q \leq M} \left\{ \max \left\{ \sum_{L=2}^Q t_{i_k L} - \sum_{L=1}^{Q-1} t_{i_{k+1} L}, 0 \right\} \right\} + \\ &+ t_{i_k 1} + \sum_{L=1}^M t_{i_{k+1} L}, \quad k = 1, \dots, n-1. \end{aligned} \quad (7.14)$$

Соотношение (7.13) можно переписать в виде

$$\bar{t}_{\max}(\pi) = \sum_{k=1}^{n-1} t^k(\pi) - \sum_{k=2}^{n-1} \sum_{L=1}^M t_{i_k L}. \quad (7.15)$$

2) Покажем, что при некотором Q , $1 \leq Q \leq M$,

$$\bar{t}_{\max}(\pi) = \sum_{L=1}^{Q-1} t_{i_1 L} + \sum_{k=1}^n t_{i_k Q} + \sum_{L=Q+1}^M t_{i_n L} \quad (7.16)$$

для всех перестановок π тогда и только тогда, когда матрица $\|t_{iL}\|$ длительностей обслуживания является Q -доминантной.

Вначале докажем необходимость. Пусть при некотором Q , $1 \leq Q \leq M$, соотношение (7.16) справедливо для всех π . Используя метод математической индукции, покажем, что для всех k , $1 \leq k \leq n-1$, и всех π

$$t^k(\pi) = \sum_{L=1}^Q t_{i_k L} + \sum_{L=Q}^M t_{i_{k+1} L}. \quad (7.17)$$

Пусть $\tau_k(\pi)$ означает сумму элементов строки i_k матрицы $\|t_{iL}\|$, входящих в (7.15). Из (7.15) следует, что $\tau_1(\pi) = \sum_{L=1}^Q t_{i_k L}$ и, следовательно, $t^k(\pi)$ удовлетворяет (7.17) при $k=1$.

Пусть (7.17) справедливо для всех k , $k=1, \dots, p-1$, $1 < p \leq n-1$. Покажем, что это соотношение справедливо и при $k=p$.

Предположим, что

$$t^p(\pi) = \sum_{L=1}^H t_{i_p L} + \sum_{L=H}^M t_{i_{p+1} L}.$$

Тогда

$$\tau_p(\pi) = \sum_{L=1}^H t_{i_p L} + \sum_{L=Q}^M t_{i_p L} - \sum_{L=1}^M t_{i_p L}.$$

С другой стороны, в силу (7.16) $\tau_p(\pi) = t_{i_p Q}$. Следовательно, $H=Q$ и (7.17) выполняется для всех $k=1, \dots, n-1$.

Последнее утверждение в свою очередь означает, что матрица $\|t_{iL}\|$ является Q -доминантной (см. п. 7.1).

Для доказательства достаточности заметим, что из п. 7.1 следует, что для всех π имеет место соотношение (7.17). Подставляя его в (7.15), получаем требуемое соотношение (7.16).

Оптимальная перестановка π^* может быть построена за $O(nM)$ операций так же, как и в п. 7.1.

7.8. Заменяем исходную систему, состоящую из M приборов, двухстадийной системой поточного типа, состоящей из приборов A и B . Каждое требование $i \in N$ обслуживается прибором A в течение a_i единиц времени, затем (без задержки) прибором B в течение b_i единиц времени, где

$$a_i = \sum_{L=1}^{M-1} t_{iL}, \quad b_i = \sum_{L=2}^M t_{iL}, \quad i \in N.$$

Пусть $\hat{t}_{\max}(\pi)$ означает общее время обслуживания требований множества N в построенной системе. Как следует из п. 3.5 § 3 этой главы,

$$\begin{aligned} \hat{t}_{\max}(\pi) = & \sum_{k=1}^{n-1} \max \left\{ \sum_{L=2}^M t_{i_k L} - \sum_{L=1}^{M-1} t_{i_{k+1} L}, 0 \right\} + \\ & + \sum_{L=2}^M t_{i_n L} + \sum_{k=1}^n \sum_{L=1}^{M-1} t_{i_k L}. \end{aligned}$$

Если для любой перестановки π в исходной системе

$$\begin{aligned} \bar{t}_{\max}(\pi) = & \sum_{k=1}^{n-1} \max \left\{ \sum_{L=2}^M t_{i_k L} - \sum_{L=1}^{M-1} t_{i_{k+1} L}, 0 \right\} + \\ & + \sum_{k=1}^{n-1} t_{i_{k1}} + \sum_{L=1}^M t_{i_n L}, \end{aligned} \quad (7.18)$$

то оптимальные по быстродействию расписания в исходной и вновь построенной задачах определяются одной и той же перестановкой π^* .

В этом пункте приводятся достаточные условия выполнимости (7.18) для всех π .

1) Пусть для любых двух требований i и j , $i \in N$, $j \in N$, $i \neq j$, найдется такое $Q = Q(i, j)$, $1 \leq Q \leq M - 1$, что

$$\begin{aligned} \min \left\{ \sum_{L=1}^H t_{iL} - \sum_{L=2}^{H+1} t_{jL}, \sum_{L=R+1}^M t_{iL} - \sum_{L=R}^{M-1} t_{jL} \right\} &\geq 0, \\ \min \left\{ \sum_{L=1}^H t_{jL} - \sum_{L=2}^{H+1} t_{iL}, \sum_{L=R+1}^M t_{jL} - \sum_{L=R}^{M-1} t_{iL} \right\} &\geq 0, \end{aligned} \quad (7.19)$$

$$H = 1, \dots, Q - 1 \quad (Q \neq 1), \quad R = Q + 1, \dots, M - 1 \quad (Q \neq M - 1).$$

Покажем, что в этом случае для всех π выполняется соотношение (7.18).

Заметим, что если (7.19) выполняется при одном и том же Q для всех пар требований, получаем условие (7.6).

Из результатов п. 7.2, 2) следует, что для всех π и k имеет место соотношение

$$t^k(\pi) = \max \left\{ t_{ik1} + \sum_{L=1}^M t_{ik+1L}, \sum_{L=1}^M t_{ikL} + t_{ik+1M} \right\},$$

которое можно переписать в виде

$$t^k(\pi) = \max \left\{ \sum_{L=2}^M t_{ikL} - \sum_{L=1}^{M-1} t_{ik+1L}, 0 \right\} + t_{ik1} + \sum_{L=1}^M t_{ik+1L}.$$

Подставляя это соотношение в (7.15), получаем

$$\begin{aligned} \bar{t}_{\max}(\pi) &= \sum_{k=1}^{n-1} \max \left\{ \sum_{L=2}^M t_{ikL} - \sum_{L=1}^{M-1} t_{ik+1L}, 0 \right\} + \\ &+ \sum_{k=1}^{n-1} t_{ik1} + \sum_{k=2}^n \sum_{L=1}^M t_{ikL} - \sum_{k=2}^{n-1} \sum_{L=1}^M t_{ikL} = \\ &= \sum_{k=1}^{n-1} \max \left\{ \sum_{L=2}^M t_{ikL} - \sum_{L=1}^{M-1} t_{ik+1L}, 0 \right\} + \sum_{k=1}^{n-1} t_{ik1} + \sum_{L=1}^M t_{inL}. \end{aligned}$$

3) Пусть для любой пары требований i и j , $i \in N$, $j \in N$, $i \neq j$, найдется такое $\alpha = \alpha(i, j) \in [0, 1]$, что

$$\begin{aligned} \alpha \left(\sum_{L=1}^{Q-1} t_{jL} - \sum_{L=2}^Q t_{iL} \right) + (1 - \alpha) \left(\sum_{L=Q+1}^M t_{iL} - \right. \\ \left. - \sum_{L=Q}^{M-1} t_{jL} \right) \geq 0, \quad Q = 2, \dots, M - 1. \end{aligned} \quad (7.20)$$

Покажем, что в этом случае для всех π выполняется соотношение (7.18).

Пусть i и j — произвольная пара требований, для которых при некотором $\alpha \in [0, 1]$ справедливо (7.20).

Тогда для всех Q , $2 \leq Q \leq M-1$, выполняется неравенство

$$\sum_{L=2}^Q t_{iL} - \sum_{L=1}^{Q-1} t_{jL} \leq (1-\alpha) \left(\sum_{L=2}^M t_{iL} - \sum_{L=1}^{M-1} t_{jL} \right).$$

Если при этом $\sum_{L=2}^M t_{iL} > \sum_{L=1}^{M-1} t_{jL}$, то

$$\max_{2 \leq Q \leq M} \left\{ \max_{L=2}^Q t_{iL} - \sum_{L=1}^{Q-1} t_{jL}, 0 \right\} = \sum_{L=2}^M t_{iL} - \sum_{L=1}^{M-1} t_{jL}.$$

В противном случае

$$\begin{aligned} \max_{2 \leq Q \leq M} \left\{ \max_{L=2}^Q t_{iL} - \sum_{L=1}^{Q-1} t_{jL}, 0 \right\} &= 0 = \\ &= \max_{L=2}^M \left\{ \sum_{L=2}^M t_{iL} - \sum_{L=1}^{M-1} t_{jL}, 0 \right\}. \end{aligned}$$

Следовательно, для любой перестановки $\pi = (i_1, i_2, \dots, i_n)$ имеют место соотношения

$$\begin{aligned} \max_{2 \leq Q \leq M} \left\{ \max_{L=2}^Q t_{i_k L} - \sum_{L=1}^{Q-1} t_{i_{k+1} L}, 0 \right\} &= \\ &= \max_{L=2}^M \left\{ \sum_{L=2}^M t_{i_k L} - \sum_{L=1}^{M-1} t_{i_{k+1} L}, 0 \right\}, \quad k = 1, \dots, n-1. \end{aligned}$$

Сравнивая эти соотношения с (7.13), убеждаемся, что для всех π справедливо (7.18).

Таким образом, если матрица $\|t_{iL}\|$ длительностей обслуживания удовлетворяет условиям (7.19) или (7.20) для всех пар требований i и j , $i, j \in N$, $i \neq j$, то оптимальная перестановка π^* может быть построена при помощи алгоритма из п. 3.5 § 3 этой главы. Трудоемкость ее построения не превосходит $O(nM + n \log n)$ операций.

7.9. В заключение этого параграфа рассмотрим задачу построения перестановки π^* , которой соответствует наименьшее значение суммарного

взвешенного времени $\sum_{k=1}^n w_{i_k} \bar{t}_{i_k}(\pi)$ обслуживания требований. Предпола-

гается, что каждое требование обслуживается без задержек.

Покажем, что при некотором Q , $1 \leq Q \leq M$, соотношение

$$\sum_{k=1}^n w_{i_k} \bar{t}_{i_k}(\pi) = \sum_{k=1}^n w_{i_k} \left(\sum_{L=1}^{Q-1} t_{i_k L} + \sum_{j=1}^k t_{i_j Q} + \sum_{L=Q+1}^M t_{i_k L} \right) \quad (7.21)$$

имеет место для всех перестановок $\pi = (i_1, i_2, \dots, i_n)$ тогда и только тогда, когда матрица $\|t_{iL}\|$ является Q -доминантной.

Соотношение (7.21), очевидно, имеет место для всех перестановок π тогда и только тогда, когда для всех π

$$\bar{t}_k(\pi) = \sum_{L=1}^{Q-1} t_{i_1 L} + \sum_{j=1}^k t_{i_j Q} + \sum_{L=Q+1}^M t_{i_k L}, \quad k = 1, \dots, n.$$

Как следует из п. 7.7, последнее утверждение эквивалентно тому, что матрица $\|t_{iL}\|$ является Q -доминантной.

Оптимальная перестановка может быть построена за $O(n \log n)$ операций при $Q = 1$ и $O(nM + n^2)$ операций при $Q > 1$ таким же образом, как и в п. 7.5.

§ 8. Приближенные алгоритмы

В этом параграфе описываются некоторые эффективные приближенные алгоритмы решения задач теории расписаний для систем поточного типа, гарантирующие построение расписания, имеющего ограниченную абсолютную или относительную погрешность.

8.1. Пусть дана некоторая задача теории расписаний, заключающаяся в минимизации целевого функционала $F(s)$, и s^* — оптимальное расписание. Предположим, что имеется алгоритм Φ , который по любому набору входных данных задачи отыскивает некоторое допустимое расписание s^0 . Качество расписания s^0 будем оценивать величиной *абсолютной* $|F(s^0) - F(s^*)|$ или *относительной* $\Delta = |F(s^0) - F(s^*)| / |F(s^*)|$ погрешности. Будем говорить, что алгоритм Φ имеет *априорную оценку точности*, если для любого набора входных данных задачи известна (может быть вычислена за полиномиальное число операций) такая величина K , что либо $|F(s^0) - F(s^*)| \leq K$, либо $\Delta \leq K$. Ниже рассматриваются именно такие алгоритмы.

В п. 8.2 доказано утверждение, на основании которого построен алгоритм приближенного решения задачи минимизации общего времени обслуживания требований в системе поточного типа. Алгоритм отыскивает расписание, имеющее ограниченную абсолютную погрешность. В последующих пунктах описываются алгоритмы приближенного решения различных задач теории расписаний для систем поточного типа, которые отыскивают расписания, имеющие ограниченную относительную погрешность Δ .

8.2. В систему поточного типа, состоящую из $M > 2$ приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительности t_{iL} обслуживания каждого требования каждым прибором L , $L = 1, \dots, M$, предполагаются заданными. Обозначим $\max\{t_{iL} \mid i \in N, L = 1, \dots, M\}$ через t^* и $\max\{\sum_{i \in N} t_{iL} \mid L = 1, \dots, M\}$ — через $t^*(N)$.

В этом пункте будем предполагать, что $\sum_{i \in N} t_{iL} = t^*(N)$ для всех L ,

$L = 1, \dots, M$. В противном случае исходные длительности обслуживания можно было бы соответствующим образом "удлинить", не увеличивая при этом значение t^* . Если $\tilde{s}$ — произвольное расписание для задачи с измененными длительностями, то в качестве расписания для задачи с исход-

ными длительностями может быть выбрано расписание s , при котором $t_{iL}^0(s) = t_{iL}^0(\tilde{s})$, $i \in N$, $L = 1, \dots, M$.

Ниже показано, что не более чем за $O(n^2 M^3)$ операций может быть построено такое расписание s^0 , что $\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*) \leq M(M-1)t^*$.

1) Докажем следующее вспомогательное утверждение.

Пусть дано множество V , состоящее из r векторов v с m действительными компонентами. Предполагается, что $\|v\| \leq 1$ для всех $v \in V$ и $\sum_{v \in V} v = 0$. В качестве нормы $\|v\|$ вектора v может быть выбрана любая

норма, например максимум из модулей его компонент.

Покажем, что можно так пронумеровать векторы v из V , что $V = \{v^1, v^2, \dots, v^r\}$ и $\max_{1 \leq i \leq r} \left\| \sum_{k=1}^i v^k \right\| \leq m$, причем трудоемкость такой нумерации не превосходит $O(r^2 m^3)$ операций.

Если $r \leq m$, то указанное утверждение тривиально. В дальнейшем предполагается, что $r > m$.

Найдем такие множества $V_j \subseteq V$ и такие числа $\lambda_j(v)$, $v \in V_j$, $j = m, \dots, r$, что $V_m \subset V_{m+1} \subset \dots \subset V_r = V$, $|V_j| = j$ и справедливы соотношения

$$0 \leq \lambda_j(v) \leq 1, \quad v \in V_j, \quad j = m, \dots, r,$$

$$\sum_{v \in V_j} \lambda_j(v) = j - m, \quad j = m, \dots, r,$$

$$\sum_{v \in V_j} \lambda_j(v) = 0, \quad j = m, \dots, r.$$

Положим $V_r = V$ и $\lambda_r(v) = (r - m)/r$ для всех $v \in V_r$.

Пусть найдены множества $V_r, V_{r-1}, \dots, V_{j+1}$ и соответствующие числа $\lambda_r(v), \lambda_{r-1}(v), \dots, \lambda_{j+1}(v)$, где $j \geq m$. Опишем способ нахождения множества V_j и чисел $\lambda_j(v)$, $v \in V_j$.

Рассмотрим систему линейных соотношений с переменными $\mu(v)$:

$$0 \leq \mu(v) \leq 1, \quad v \in V_{j+1}; \quad (8.1)$$

$$\sum_{v \in V_{j+1}} \mu(v) = j - m; \quad (8.2)$$

$$\sum_{v \in V_{j+1}} \mu(v) v = 0. \quad (8.3)$$

Многогранник, описываемый системой соотношений (8.1) – (8.3), непуст, поскольку, например,

$$\hat{\mu}(v) = \frac{j - m}{j - m + 1} \lambda_{j+1}(v), \quad v \in V_{j+1}, \quad (8.4)$$

является решением системы (8.1) – (8.3).

Для любой вершины указанного многогранника по крайней мере $j+1$ соотношение из (8.1) – (8.3) должно выполняться как точное равенство. Поскольку число равенств (8.2), (8.3) равно $m+1$, то каждая вершина многогранника среди своих $j+1$ координат имеет по крайней мере $j-m$ координат $\mu(v) \in \{0, 1\}$. Если предположить, что у некоторой вершины все координаты строго положительны, то среди этих координат имеется не менее $j-m$ единиц, что противоречит соотношению (8.2). Таким образом, среди точек многогранника, описываемого соотношениями (8.1) – (8.3), имеется точка (например, любая из его вершин), у которой по крайней мере одна из координат равна нулю.

Пусть $\mu^0(v)$, $v \in V_{j+1}$, – координаты такой точки, $v^0 \in V_{j+1}$ и $\mu^0(v^0) = 0$. Положим $V_j = V_{j+1} \setminus \{v^0\}$, $\lambda_j(v) = \mu^0(v)$, $v \in V_j$.

Этот процесс продолжается до тех пор, пока не будут найдены множество V_m и числа $\lambda_m(v)$, $v \in V_m$. Сопоставим вектору $V_j \setminus V_{j-1}$ номер j , т.е. положим $v^j = V_j \setminus V_{j-1}$, $j = m+1, \dots, r$. Остальные векторы пронумеруем числами $1, 2, \dots, m$ в произвольном порядке.

Покажем, что полученная нумерация векторов v множества V является исковой.

При $1 \leq l \leq m$, очевидно,

$$\left\| \sum_{k=1}^l v^k \right\| \leq \sum_{k=1}^l \|v^k\| \leq l \leq m.$$

Если $l > m$, то

$$\begin{aligned} \left\| \sum_{k=1}^l v^k \right\| &= \left\| \sum_{k=1}^l v^k - \sum_{v \in V_l} \lambda_l(v) v \right\| = \\ &= \left\| \sum_{v \in V_l} v - \sum_{v \in V_l} \lambda_l(v) v \right\| \leq \\ &\leq \sum_{v \in V_l} (1 - \lambda_l(v)) \|v\| \leq l - (l - m) = m. \end{aligned}$$

Нетрудно видеть, что наиболее трудоемким этапом описанного метода нумерации векторов v множества V является получение на каждой итерации j , $j = r-1, r-2, \dots, m$, решения $\mu^0(v)$, $v \in V_{j+1}$, системы (8.1) – (8.3) такого, что для некоторого $v^0 \in V_{j+1}$ значение $\mu^0(v^0) = 0$. Опишем один из возможных подходов к нахождению $\mu^0(v)$, $v \in V_{j+1}$.

Если среди значений $\hat{\mu}(v)$, $v \in V_{j+1}$, вычисляемых по формуле (8.4), содержится хотя бы один нуль, то можно положить $\mu^0(v) = \hat{\mu}(v)$, $v \in V_{j+1}$. Если все $\hat{\mu}(v) > 0$, $v \in V_{j+1}$, то в силу (8.2) из множества V_{j+1} можно выделить $m+2$ вектора v таких, что $0 < \hat{\mu}(v) < 1$. Обозначим множество выделенных векторов через W и найдем нетривиальное решение $\hat{\rho}(v)$, $v \in W$,

следующей однородной системы из $m + 1$ линейных уравнений с $m + 2$ неизвестными $\rho(v)$, $v \in W$:

$$\sum_{v \in W} \rho(v) = 0,$$

$$\sum_{v \in W} \rho(v)v = 0.$$

Решение $\hat{\rho}(v)$, $v \in W$, может быть получено за $O(m^3)$ операций. Найдем максимальное значение τ^0 параметра τ , при котором выполняются неравенства $0 \leq \hat{\mu}(v) + \tau \rho(v) \leq 1$, $v \in W$.

Положим $\hat{\rho}(v) = 0$, $v \in V_{j+1} \setminus W$, и $\bar{\mu}(v) = \hat{\mu}(v) + \tau^0 \rho(v)$, $v \in V_{j+1}$. Нетрудно убедиться, что $\bar{\mu}(v)$, $v \in V_{j+1}$ является решением системы (8.1) – (8.3) и среди $\bar{\mu}(v)$, $v \in V_{j+1}$, дробных величин меньше, чем среди $\hat{\mu}(v)$, $v \in V_{j+1}$.

Если среди $\bar{\mu}(v)$, $v \in V_{j+1}$, содержится хотя бы один нуль, то можно положить $\mu^0(v) = \bar{\mu}(v)$, $v \in V_{j+1}$. В противном случае $\bar{\mu}(v)$ можно выбрать в качестве $\hat{\mu}(v)$ и повторить описанный процесс. В результате не более чем через $j - m$ шагов будет получено искомое решение $\mu^0(v)$, $v \in V_{j+1}$.

Тем самым, общее число операций на всех $r - m$ итерациях процесса нумерации векторов v множества V не превосходит $O(r^2 m^3)$.

Отметим, что в случае, когда $\|v\| \leq C$, $v \in V$, при найденной нумерации векторов v множества V имеет место соотношение

$$\max_{1 \leq l \leq r} \left\| \sum_{k=1}^l v^k \right\| \leq C m.$$

2) Описанный выше процесс нумерации векторов может быть использован для приближенного решения задачи построения оптимального по быстродействию расписания обслуживания требований множества $N = \{1, 2, \dots, n\}$ в системе поточного типа из M приборов.

Введем в рассмотрение множество V , состоящее из n векторов v вида $(t_{i1} - t_{i2}, t_{i2} - t_{i3}, \dots, t_{i, M-1} - t_{iM})$. Так как $\sum_{i \in N} t_{iL} = t^*(N)$, $L = 1, \dots, M$, то $\sum_{v \in V} v = 0$, и если в качестве нормы $\|v\|$ вектора v выбрать максимум из модулей его компонент, то $\|v\| \leq t^*$, $v \in V$.

Нумеруя векторы v множества V в соответствии с алгоритмом, описанным в п. 1), получаем $V = \{v^1, v^2, \dots, v^n\}$. Перенумеруем, если это необходимо, требования множества N таким образом, чтобы $v^i = (t_{i1} - t_{i2}, t_{i2} - t_{i3}, \dots, t_{i, M-1} - t_{iM})$, $i \in N$.

Из утверждения, доказанного в п. а), следует, что

$$-(M-1)t^* \leq \sum_{i=1}^k t_{iL} - \sum_{i=1}^k t_{i, L+1} \leq$$

$$\leq (M-1)t^*, \quad k = 1, \dots, n, \quad L = 1, \dots, M-1.$$

Расписание s^0 построим следующим образом. Все приборы обслуживают требования в одной и той же последовательности $(1, 2, \dots, n)$, причем $t_{1L}^0 = (L-1)Mt^* + t_{iL}^0 = t_{i-1,L}^0 + t_{i-1,L}$, $i = 2, \dots, n$, $L = 1, \dots, M$.

Покажем, что при этом ни одно требование не обслуживается двумя и более приборами одновременно, т.е. $\bar{t}_{kL} \leq t_{k,L+1}^0$, $k = 1, \dots, n$, $L = 1, \dots, M-1$.

Действительно,

$$\begin{aligned} t_{k,L+1}^0 - \bar{t}_{kL} &= (LMt^* + \sum_{i=1}^{k-1} t_{i,L+1}) - ((L-1)Mt^* + \sum_{i=1}^k t_{iL}) = \\ &= Mt^* + \sum_{i=1}^{k-1} (t_{i,L+1} - t_{iL}) - t_{kL} \geq Mt^* - (M-1)t^* - t^* = 0. \end{aligned}$$

Очевидно, $\bar{t}_{\max}(s^*) \geq t^*(N)$. С другой стороны, $\bar{t}_{\max}(s^0) = t_{1M}^0 + t^*(N) = M(M-1)t^* + t^*(N)$. Следовательно, $\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*) \leq M(M-1)t^*$.

Трудоемкость алгоритма построения расписания s^0 определяется трудоемкостью нумерации векторов множества V и не превосходит $O(n^2 M^3)$ операций.

Отметим, что при расписании s^0 все приборы обслуживают требования в одной и той же последовательности. Расписание s^0 может быть выбрано в качестве приближенного решения и в том случае, когда разные приборы могут обслуживать требования в различных последовательностях, а также в том случае, когда допускаются прерывания. Оценка абсолютной погрешности при этом не изменится.

8.3. В систему поточного типа, состоящую из $M \geq 2$ приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Заданы длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$. Введем в рассмотрение класс S расписаний таких, что для всех $s \in S$ во временном интервале $(0, t_{\max}(s)]$ в любой момент времени функционирует по крайней мере один прибор. Установим некоторые свойства расписаний из класса S .

1) Рассмотрим задачу минимизации общего времени обслуживания $\bar{t}_{\max}(s)$. Пусть s^* — оптимальное расписание. Очевидно, $s^* \in S$. Покажем, что для любого расписания $s \in S$ имеет место неравенство $\Delta = (\bar{t}_{\max}(s) - \bar{t}_{\max}(s^*)) / \bar{t}_{\max}(s^*) \leq M - 1$.

$$\text{Действительно } \bar{t}_{\max}(s^*) \geq \max \left\{ \sum_{i=1}^n t_{iL} \mid L = 1, \dots, M \right\} \geq \frac{1}{M} \sum_{i=1}^n \sum_{L=1}^M t_{iL}.$$

С другой стороны, очевидно, $\bar{t}_{\max}(s) \leq \sum_{i=1}^n \sum_{L=1}^M t_{iL}$. Отсюда $\bar{t}_{\max}(s) / \bar{t}_{\max}(s^0) \leq M$, т.е. $\Delta \leq M - 1$.

Покажем, что полученная оценка достижима.

Положим $n = M$, $t_{iL} = W$, $i = 1, \dots, M$, $i \neq L$; $t_{iL} = \epsilon_i$, $i = 1, \dots, M$, $L = 1, \dots, M$, $i \neq L$. Здесь $W > \epsilon_1$, $\epsilon_i \geq \epsilon_{i+1} > 0$, $i = 1, \dots, M-1$. При оптимальном расписании s^* каждый прибор обслуживает требования в последова-

тельности $(n, n-1, \dots, 2, 1)$ и $\bar{t}_{\max}(s^*) = \sum_{i=2}^M \epsilon_i + W + (M-1)\epsilon_1$. В качестве

расписания s выберем расписание, при котором каждый прибор обслуживает требования в последовательности $(1, 2, \dots, n)$, т.е. в порядке невозрастания

значений $\sum_{L=1}^M t_{iL}$. Имеем $\bar{t}_{\max}(s) = MW + \sum_{i=1}^{M-1} \epsilon_i$. Устремляя ϵ_i , $i = 1, \dots$
 $\dots, M$, к нулю, получаем $\Delta \rightarrow M - 1$.

2) Рассмотрим задачу минимизации суммарного времени обслуживания $\sum_{i=1}^n \bar{t}_i(s)$. Пусть s^* — оптимальное расписание. Очевидно, $s^* \in S$. Покажем, что для любого расписания имеет место неравенство $\Delta = (\sum_{i=1}^n \bar{t}_i(s) - \sum_{i=1}^n \bar{t}_i(s^*)) / \sum_{i=1}^n \bar{t}_i(s^*) \leq n - 1$.

Действительно, $\sum_{i=1}^n \bar{t}_i(s^*) \geq \sum_{i=1}^n \sum_{L=1}^M t_{iL}$. С другой стороны, при расписании $s \in S$ значения $\bar{t}_i(s) \leq \sum_{i=1}^n \sum_{L=1}^M t_{iL}$, $i = 1, \dots, n$, т.е. $\sum_{i=1}^n \bar{t}_i(s) \leq n \sum_{i=1}^n \sum_{L=1}^M t_{iL}$. Отсюда $\sum_{i=1}^n \bar{t}_i(s) / \sum_{i=1}^n \bar{t}_i(s^*) \leq n$, т.е. $\Delta \leq n - 1$.

Покажем, что полученная оценка *достижима*.

Положим $M = 2$, $t_{11} = \epsilon$, $i = 1, \dots, n$; $t_{12} = W$, $t_{22} = \delta$, $i = 2, \dots, n$. Здесь $0 < \delta < \epsilon < W/n^2$. При оптимальном расписании s^* оба прибора обслуживают требования в последовательности $(2, 3, \dots, n, 1)$. При этом $\sum_{i=1}^n \bar{t}_i(s^*) = (\epsilon + \delta) + (2\epsilon + \delta) + \dots + ((n-1)\epsilon + \delta) + (n\epsilon + W) = n(n+1)\epsilon/2 + (n-1)\delta + W$. В качестве расписания s выберем расписание, при котором оба прибора обслуживают требования в последовательности $(1, 2, \dots, n)$. Отметим, что это расписание будет оптимальным по быстродействию и может быть получено при помощи алгоритма из п. 1.3 § 1 этой главы. Имеем $\bar{t}_i(s) = W + \epsilon + (i-1)\delta$, т.е. $\sum_{i=1}^n \bar{t}_i(s) = n(W + \epsilon) + (n-1)n\delta/2$.

Устремляя ϵ и δ к нулю, получаем $\Delta \rightarrow n - 1$.

8.4. Опишем алгоритмы построения такого расписания $s^0 \in S$, при котором $\Delta = (\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*)) / \bar{t}_{\max}(s^*) \leq m - 1$, где $m = \lceil M/2 \rceil$, и запись $\lceil a \rceil$ означает наименьшее целое число, не меньшее a .

1) Пусть процесс обслуживания каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$, протекает без прерываний и последовательности, в которых приборы обслуживают требования, могут быть различными для разных приборов.

Если число приборов M нечетно, то множество приборов дополним фиктивным прибором $M + 1$, положив $t_{i, M+1} = 0$, $i \in N$. Разобьем множество приборов на m пар вида $(2L - 1, 2L)$, $L = 1, \dots, m$. Каждую пару приборов указанного вида будем интерпретировать как двухстадийную систему поточного типа, в которую в момент времени $d = 0$ поступает множество требований $N = \{1, 2, \dots, n\}$.

Согласно алгоритму из п. 1.3 § 1 этой главы найдем последовательность π_L , которой соответствует оптимальное по быстродействию расписание обслуживания требований множества N в системе поточного типа, состоящей из приборов $2L - 1$ и $2L$, $L = 1, \dots, m$, с общим временем обслуживания, равным $f(\pi_L)$. Искомое расписание s^0 определим как расписание

из класса S , при котором приборы $2L - 1$ и $2L$ обслуживают требования в последовательности π_L , $L = 1, \dots, m$. Трудоемкость построения s^0 не превосходит $O(nM \log n)$ операций.

Оценим величину относительной погрешности Δ построенного расписания s^0 . Очевидно, $\bar{t}_{\max}(s^*) \geq \max\{f(\pi_L) \mid L = 1, \dots, m\}$. С другой стороны,

$$\bar{t}_{\max}(s^0) \leq \sum_{L=1}^m f(\pi_L) \leq m \max\{f(\pi_L) \mid L = 1, \dots, m\}. \text{ Тогда } \bar{t}_{\max}(s^0)/\bar{t}_{\max}(s^*) \leq m, \text{ т.е. } \Delta \leq m - 1.$$

Покажем, что полученная оценка *достижима*. Положим $M = 3$, $t_{i1} = \delta$, $t_{i2} = W$, $t_{i3} = 0$, $i = 1, \dots, n-1$; $t_{n1} = t_{n2} = \epsilon$, $t_{n3} = (n-1)W$. Здесь $W > \epsilon > \delta > 0$. При оптимальном расписании s^* приборы 1 и 2 обслуживают требования в одной и той же последовательности $(n, 1, 2, \dots, n-1)$, а прибор 3 обслуживает требование n , причем $t_{n3}^0(s^*) = 2\epsilon$. Следовательно, $\bar{t}_{\max}(s^*) = 2\epsilon + (n-1)W$. Для построения расписания s^0 дополним множество приборов прибором 4, положив $t_{i4} = 0$, $i = 1, \dots, n$. Убеждаемся, что при расписании s^0 приборы 1 и 2 обслуживают требования в последовательности $(1, 2, \dots, n)$, которую можно найти, используя алгоритм из п. 1.3. § 1 этой главы. С другой стороны, прибор 3 обслуживает требование n , причем $t_{n3}^0(s^0) = \epsilon + \delta + (n-1)W$. Отсюда $\bar{t}_{\max}(s^0) = \epsilon + \delta + 2(n-1)W$.

Устремляя ϵ и δ к нулю, имеем $\bar{t}_{\max}(s^0)/\bar{t}_{\max}(s^*) \rightarrow 2 = \lceil 3/2 \rceil$, т.е. $\Delta \rightarrow 1$.

Отметим, что расписание s^0 может быть выбрано в качестве приближенного решения рассматриваемой задачи и в том случае, когда *допускаются прерывания процесса обслуживания*. Оценка относительной погрешности при этом не увеличится.

2) Пусть процесс обслуживания каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$, протекает без прерываний, но в отличие от п. а) все приборы должны обслуживать требования в одной и той же последовательности.

Заменим исходную систему двухстадийной системой поточного типа, состоящей из двух приборов A и B . Каждое требование i множества N обслуживается сначала прибором A в течение $a_i = \sum_{L=1}^m t_{iL}$ единиц времени, затем прибором B в течение $b_i = \sum_{L=m+1}^M t_{iL}$ единиц времени. Используя алгоритм из п. 1.3 § 1 этой главы, отыщем последовательность π^0 , которой соответствует оптимальное (по быстрдействию) расписание обслуживания требований множества N в построенной двухстадийной системе при условии, что обслуживание требований начинается в момент времени $d = 0$. Искомое расписание s^0 определим как расписание из класса S , при котором приборы $1, 2, \dots, M$ исходной системы обслуживают требования в одной и той же последовательности π^0 . Трудоемкость построения s^0 , очевидно, не превосходит $O(nM + n \log n)$ операций.

Оценим величину относительной погрешности Δ построенного расписания. Общее время обслуживания при расписании s^0 и при оптимальном расписании s^* (которое определяется последовательностью π^*) обозначим соответственно через $\bar{t}_{\max}(\pi^0)$ и $\bar{t}_{\max}(\pi^*)$.

Пусть Q и H , $1 \leq Q \leq m < H \leq M$, — такие приборы, что $\sum_{i \in N} t_{iQ} =$
 $= \max \left\{ \sum_{i \in N} t_{iL} \mid L = 1, \dots, m \right\}$ и $\sum_{i \in N} t_{iH} = \max \left\{ \sum_{i \in N} t_{iL} \mid L = m + 1, \dots \right.$
 $\left. \dots, M \right\}$. Через $f_{QH}(\pi^0)$ обозначим общее время обслуживания требований

множества N в системе поточного типа, состоящей из двух приборов Q и H , в предположении, что, во-первых, длительности обслуживания требований приборами Q и H равны соответственно исходным длительностям t_{iQ} и t_{iH} , $i \in N$, во-вторых, обслуживание начинается в момент времени $d = 0$ и, в третьих, оба прибора Q и H обслуживают требования в одной и той же последовательности π^0 .

Очевидно, $\bar{t}_{\max}(\pi^*) \geq f_{QH}(\pi^0)$. С другой стороны, $\bar{t}_{\max}(\pi^0) \leq m f_{QH}(\pi^0)$. Отсюда $\bar{t}_{\max}(\pi^0) \bar{t}_{\max}(\pi^*) \leq m$, т.е. $\Delta \leq m - 1$.

Покажем, что полученная оценка достижима. Рассмотрим M требований i , положив $t_{iL} = \epsilon$, $i \neq L$, и $t_{iL} = W$, $i = L$, $i = 1, \dots, M$, $L = 1, \dots, M$. Пусть N_i — множество требований, состоящее из $k > 1$ экземпляров требования i указанного вида, $i = 1, \dots, M$. Положим $N = \bigcup_{i=1}^M N_i$. Нетрудно убедиться,

что $\pi^* = (\pi_{NM}, \pi_{NM-1}, \dots, \pi_{N_1})$, и следовательно, $\bar{t}_{\max}(\pi^*) = kW + (M - 1)(k + 1)\epsilon$. С другой стороны, $\pi^0 = (\pi_{Nm+1}, \pi_{Nm+2}, \dots, \pi_{NM}, \pi_{N_1}, \pi_{N_2}, \dots, \pi_{Nm})$ и, следовательно, $\bar{t}_{\max}(\pi^0) = mkW + (M - 1 + km)\epsilon$. Устремляя ϵ к нулю, имеем $\bar{t}_{\max}(\pi^0) / \bar{t}_{\max}(\pi^*) \rightarrow m$, т.е. $\Delta \rightarrow m - 1$.

Аналогичный подход может быть использован для приближенного решения задачи построения оптимального по быстродействию расписания без задержек в процессе обслуживания каждого требования. В этом случае в качестве расписания s^0 выбирается расписание из класса S , при котором все приборы обслуживают требования в одной и той же последовательности π^0 . Эта последовательность отыскивается по алгоритму из п. 3.5 § 3 этой главы. Трудоемкость построения и в этом случае не превосходит $O(nM + n \log n)$ операций, а относительная погрешность $\Delta \leq m - 1$.

8.5. Рассмотрим задачу минимизации суммарного времени обслуживания требований. Опишем алгоритм построения расписания $s^0 \in S$, при

котором $\Delta = \left(\sum_{i=1}^n \bar{t}_i(s^0) - \sum_{i=1}^n \bar{t}_i(s^*) \right) / \sum_{i=1}^n \bar{t}_i(s^*) \leq M - 1$.

Пусть π^0 — перестановка требований, упорядоченных по неубыванию значений $\sum_{L=1}^M t_{iL}$. Расписание s^0 определим как расписание из класса S , при котором все приборы обслуживают требования в одной и той же последовательности π^0 .

Оценим величину относительной погрешности Δ построенного расписания. Не нарушая общности, будем предполагать, что $\sum_{L=1}^M t_{1L} \leq \sum_{L=1}^M t_{2L} \leq \dots \leq \sum_{L=1}^M t_{nL}$, т.е. $\pi^0 = (1, 2, \dots, n)$. Тогда $\bar{t}_i(s^0) \leq \sum_{k=1}^i \sum_{L=1}^M t_{kL}$, $i =$

$= 1, \dots, n$, и следовательно, $\sum_{i=1}^n \bar{t}_i(s^0) \leq \sum_{i=1}^n \sum_{k=1}^i \sum_{L=1}^M t_{kL}$. Пусть $\pi^* = (i_1, i_2, \dots, i_n)$ — перестановка, определяющая оптимальное расписание s^* . Тогда $\bar{t}_{i_j}(s^*) \geq (\sum_{k=1}^j \sum_{L=1}^M t_{kL})/M \geq (\sum_{k=1}^j \sum_{L=1}^M t_{kL})/M$. Таким образом, $\sum_{i=1}^n \bar{t}_i(s^*) \geq (\sum_{i=1}^n \sum_{k=1}^i \sum_{L=1}^M t_{kL})/M$. Следовательно, $\sum_{i=1}^n \bar{t}_i(s^0) / \sum_{i=1}^n \bar{t}_i(s^*) \leq M$ и $\Delta \leq M - 1$.

Покажем, что полученная оценка достижима. Положим $M = 2, N = \{1, 2, \dots, 2n\}$; $t_{i1} = W, t_{i2} = \delta, i = 1, \dots, n$; $t_{i1} = \epsilon, t_{i2} = W, i = n + 1, \dots, 2n$. Здесь $0 < \delta < \epsilon < W/n^2$. Очевидно, $\pi^0 = (1, 2, \dots, 2n)$. Тогда при расписании s^0 имеем

$$\bar{t}_i(s^0) = iW + \delta, \quad i = 1, \dots, n,$$

$$\bar{t}_i(s^0) = iW + \epsilon, \quad i = n + 1, \dots, 2n,$$

$$\text{поэтому } \sum_{i=1}^{2n} \bar{t}_i(s^0) = \sum_{i=1}^{2n} iW + n\delta + n\epsilon.$$

Нетрудно проверить, что в качестве s^* можно выбрать расписание, при котором оба прибора обслуживают требования, например, в последовательности $\pi^* = (n + 1, 1, n + 2, 2, \dots, 2n, n)$. При этом

$$\bar{t}_i(s^*) = iW + \delta + i\epsilon, \quad i = 1, \dots, n,$$

$$\bar{t}_i(s^*) = (i - n)W + (i - n)\epsilon, \quad i = n + 1, \dots, 2n.$$

$$\text{Поэтому } \sum_{i=1}^{2n} \bar{t}_i(s^*) = 2 \sum_{i=1}^n iW + 2 \sum_{i=1}^n i\epsilon + n\delta.$$

Устремляя ϵ и δ к нулю, имеем $\sum_{i \in N} \bar{t}_i(s^0) / \sum_{i \in N} \bar{t}_i(s^*) \rightarrow 2n(2n + 1)W / 2n(n + 1)W$, т.е. $\Delta \rightarrow 1$ при $n \rightarrow \infty$.

§ 9. Библиографическая справка

Системы поточного типа принадлежат к числу наиболее известных много-стадийных систем, изучаемых в рамках теории расписаний.

Далее предполагается (если не оговорено противное), что требования поступают в очередь на обслуживание в момент времени $d = 0$, длительности обслуживания $t_{iL} \geq 0$ и $t^* = \max \{t_{iL} \mid i \in N, L = 1, \dots, M\}$. Все требования обслуживаются без прерываний и все приборы обслуживают требования в одной и той же последовательности.

9.1. Задача минимизации общего времени обслуживания известна в литературе как $(n \times M)$ -задача Беллмана — Джонсона. Свойства расписаний для этой задачи (см. п. 1.2) приведены в [147]. В.Ф. Золотухин [110] предложил подход, позволяющий исключить из рассмотрения заведомо не оптимальные расписания из числа тех, при которых не все приборы обслуживают требования в одной и той же последовательности. Большинство авторов рассматривают ситуацию, описанную в п. 1.2, 5). В этом случае задача состоит в нахождении перестановки, доставляющей минимум функ-

ционалу (1.1), который впервые, по-видимому, встречается в явном виде в работе Г. Яшке [609]. Задача максимизации функционала (1.1) на множестве всех перестановок требований рассматривалась Н.А. Лепешинским [188] и В. Шварцем [806]. Разнообразные свойства перестановок, которым соответствует оптимальное (по быстродействию) расписание, установлены О.И. Мельниковым [200].

Алгоритм решения задачи при $M = 2$ (см. п. 1.3) предложен С. Джонсоном [612] и Р. Беллманом [25, 464, 465]. Этот алгоритм можно использовать и в том случае, когда задано число $\Delta > 0$ такое, что $t_{i2}^0 \geq \Delta$, $i \in N$, [149]. Н.А. Лепешинский [187] исследовал вопрос о том, в каких пределах можно изменять значения длительностей обслуживания, чтобы перестановка, генерируемая алгоритмом из п. 1.3, оставалась неизменной. Условия, при которых любая перестановка является оптимальной, установлены О.И. Мельниковым [204]. Методы нахождения всех оптимальных перестановок предложены в [118, 201, 722, 806].

Обобщение задачи минимизации $\bar{t}_{\max}(s)$ при $M = 2$, описанное в п. 1.4, рассмотрено С.А. Баркан [20]. Известны и другие обобщения этой задачи. Пусть заданы длительности $\delta_{iL} \geq 0$, $\tau_{iL} \geq 0$ пуско-наладочных и переналадочных работ, выполняемых для прибора L перед обслуживанием и после обслуживания требования $i \in N$ этим прибором, $L = 1, 2$. Предполагаются также заданными величины $\xi_i \geq 0$, $\eta_i \geq 0$, $\zeta_i \geq 0$ такие, что при любом расписании значения $t_{i2}^0 - t_{i1}^0 \geq \xi_i$, $\bar{t}_{i2} - \bar{t}_{i1} \geq \eta_i$, $t_{i2}^0 - \bar{t}_{i1} \geq \zeta_i$, $i \in N$. Задачи построения оптимальных расписаний рассматривались в [48, 178, 179] (учитывались только значения ξ_i), в [690] ($\xi_i = \eta_i$), в [671, 795, 796, 799] (ζ_i), в [613, 630, 691] (ξ_i, η_i), в [838] (δ_{i1}, δ_{i2}), в [703] (ξ_i, η_i, τ_{i1}), в [675] (ξ_i, η_i, ζ_i), в [619] ($\delta_{i1}, \delta_{i2}, \xi_i = \eta_i$), в [789, 790] ($\delta_{i1}, \delta_{i2}, \tau_{i1}, \tau_{i2}$), в [445] ($\delta_{i1}, \delta_{i2}, \tau_{i1}, \tau_{i2}, \zeta_i$), в [178–180] ($\delta_{i1}, \delta_{i2}, \tau_{i1}, \tau_{i2}, \xi_i, \eta_i$).

Как правило, для каждой из указанных задач перестановка, определяющая оптимальное расписание, может быть построена за $O(n \log n)$ операций. Как показал В.С. Танаев [336], многие из этих задач сводятся к задаче минимизации функционала $\max \left\{ \sum_{k=1}^u \alpha_{ik} + \beta_{iu} \mid u = 1, \dots, n \right\}$ на мно-

жестве всех перестановок вида $\pi = (i_1, i_2, \dots, i_n)$. Если не все приборы обслуживают требования в одной и той же последовательности, то эти задачи могут оказаться NP-трудными [180]. Обобщения указанных ситуаций на случай $M > 2$ обсуждаются в [705, 790, 809].

Некоторые другие задачи построения расписаний для системы поточного типа при $M = 2$ рассматривались в [441, 495]. Случай, когда на каждой из двух стадий имеются параллельные приборы, исследуется в [56, 363, 689].

9.2. Если $M = 2$, на множестве N требований задано отношение строгого порядка $\rightarrow$ и запись $i \rightarrow j$ означает, что при любом допустимом расписании $\bar{t}_{i2} \leq t_{j1}^0$, $i, j \in N$, то задача минимизации $\bar{t}_{\max}(\pi)$ является NP-трудной, даже если отношением $\rightarrow$ связаны ровно два требования [657].

Пусть $M = 2$, на множестве N требований задано отношение строгого порядка $\rightarrow$, запись $i \rightarrow j$ означает, что при любом допустимом расписании $\bar{t}_{iL} \leq t_{jL}^0$, $i, j \in N$, $i \neq j$, $L = 1, 2$, и G – граф редукции этого отношения. Задача построения оптимального по быстродействию расписания в случае,

когда каждая компонента связности графа G есть либо цепь, либо одновершинный граф, рассмотрена Т. Курису [632]. Случай древовидного графа G исследовался Я.М. Шафранским [392, 393], Е.В. Левнером и М.Ш. Левиным [176]; случай последовательно-параллельного графа G — В.С. Гордоном и Я.М. Шафранским [81–83], М.Ш. Левиным [175], К. Монмой и Дж. Сиднеем [697, 701, 774]. Я. Секигучи [770] разработал алгоритм решения более общей задачи в предположении, что G — последовательно-параллельный граф. Трудоемкость построения оптимального расписания для всех указанных задач не превышает $O(n \log n)$. Оптимизируемые функционалы в задаче Беллмана — Джонсона при $M = 2$ и некоторых ее обобщениях являются приоритето-порождающими в смысле [392, 699]. Систематическое изложение результатов, касающихся минимизации приоритето-порождающих функционалов на упорядоченных множествах содержится в монографии В.С. Танаева, В.С. Гордона и Я.М. Шафранского [350]. Если G — произвольный бесконтурный граф, то задача минимизации $\bar{t}_{\max}(\pi)$ при $M = 2$ является NP -трудной [698]. В этом случае для ее решения, а также для минимизации произвольного приоритето-порождающего функционала разработаны различные методы направленного перебора вариантов [44, 394, 395, 593, 631, 699, 775]. Частные случаи задачи минимизации общего времени обслуживания при $M = 2, 3$ и заданном на N отношении порядка рассмотрены в [382, 634, 697].

Определенный интерес представляет ситуация, когда оба прибора обслуживают требования в последовательности, которая содержит одну или несколько заранее заданных подпоследовательностей. Задача минимизации $\bar{t}_{\max}(\pi)$ и некоторые ее обобщения в указанном случае рассматривались в [445, 620, 668, 670, 672, 673]. Как правило, трудоемкость построения оптимального расписания во всех рассматриваемых случаях не превосходит $O(n \log n)$ операций.

9.3. По-видимому, первый результат, из которого следует NP -трудность задачи построения оптимального по быстродействию расписания при $M \geq 3$, получен Э.М. Лившицем и В.И. Рублинецким [190]. Другие доказательства этого факта можно найти в [480, 552, 657, 765]. Доказательство NP -трудности в сильном смысле, приведенное в п.2.1, использует замечание В.Г. Тимковского [354] относительно работы М. Гэри, Д. Джонсона и Р. Сети [533]. Основу пп. 2.2 — 2.4 составили результаты Т. Гонсалеса и С. Сахни [552].

О.И. Мельников [203] исследовал устойчивость оптимальных расписаний, а также рассмотрел случай, когда все длительности обслуживания равны нулю либо единице, и операции нулевой длительности не выполняются [202].

Определенный интерес представляет задача минимизации $\bar{t}_{\max}(s)$ при дополнительном условии, что каждый прибор должен функционировать без промежуточных простоев [14, 43]. Если $M = 2$, то соблюдение этого условия не приводит к увеличению общего времени обслуживания (см. п. 1.2). Если $M \geq 3$, то задача построения оптимального расписания является NP -трудной в сильном смысле независимо от того, допускаются прерывания или нет (см. § 2, а также [317, 419]).

9.4. Во многих публикациях исследуется задача минимизации $\bar{t}_{\max}(s)$ при запрещении задержек в процессе обслуживания требований. Принципиальный характер различий той или иной интерпретации нулевых длительностей обслуживания подчеркивается в [596, 730, 764]. Результат, приведенный в п. 3.3, получен С. Сахни и Ю. Чо [764]. Утверждение из п. 3.4, 1) доказано В. Шварцем [807], уточнившим доказательство Дж. Гупты [586]. Функционал (3.2) и связь его минимизации с задачей о коммивояжере изучены Я. Пилером [738], Н.А. Лепешинским [185, 186] и другими авторами [94, 557, 662, 749, 750, 823, 824, 837]. Ю. Грабовский и М. Шишло [567, 792] свели к задаче о коммивояжере несколько более общую задачу. В [424, 839] показано, что задача минимизации функционала (3.2) сводится также к задаче нахождения гамильтоновой цепи максимальной длины в ориентированном графе специального вида. Задача минимизации общего времени обслуживания (без задержек), когда на каждой стадии имеется несколько параллельных идентичных приборов, рассмотрена М. Сальвадором [766].

Полиномиальный алгоритм решения задачи о коммивояжере с матрицей постоянный вида (3.6) разработан П. Гилмором и Р. Гомори [543, 544]. Трудоемкость алгоритма в том виде, в котором он опубликован в [544], составляет $O(n^2)$ операций [542]. Однако, как отмечено в [734], трудоемкость алгоритма может быть уменьшена до $O(n \log n)$ за счет применения на одном из его шагов быстродействующего алгоритма нахождения минимального остовного дерева (см., например, [238]). Описание алгоритма в п. 3.5 проведено с учетом этого замечания. Несколько отличается от него алгоритм из [545], также имеющий трудоемкость $O(n \log n)$. Последняя работа содержит строгое обоснование алгоритма в более современном изложении, чем в [544], а также достаточно подробный обзор других эффективно разрешимых классов задачи о коммивояжере. Отметим также, что Р. Чадрасекаран [489] построил полиномиальный алгоритм проверки выполнимости условий (3.6) для произвольной матрицы.

Я. Ленстра, А. Ринной Кан и П. Бруккер [657] (см. также [651, 754]) установили NP -трудность задачи построения оптимального расписания без задержек для переменного числа приборов. Х. Пападимитриу и П. Канелакис [734] доказали аналогичный результат для $M \geq 4$. Доказательство из п. 3.7 принадлежит Х. Рёку [757] (см. также [656]). В [758] показано, что задача остается NP -трудной в сильном смысле при $M = 3$, если допускается либо $\bar{t}_{i2} \neq t_{i3}^0, i \in N$, либо $\bar{t}_{i1} \neq t_{i2}^0, i \in N$.

Если в любой момент времени для обслуживания каждого требования каждым прибором необходимо использовать один вид ресурса, запас которого и потребление каждым требованием в любой момент времени известны, то задача построения оптимального по быстродействию расписания (без задержек) при $M = 2$ и $t_{i1} = t_{i2} = 1, i \in N$, может быть решена за $O(n \log n)$ операций с использованием алгоритма из п. 3.5 [758]. При этом общее время обслуживания не уменьшается, если задержки будут разрешены (независимо от того, допускаются прерывания или нет) [758]. При $M \geq 3$ эта задача становится NP -трудной в сильном смысле [758]. Если же в каждый момент времени запас ресурса равен единице, а его потребление каждым требованием равно нулю или единице, то при $M = 2$

задача построения оптимального по быстродействию расписания (без прерываний, задержки разрешены) может быть решена не более чем за $O(n)$ операций, если $t_{i1} = t_{i2} = 1$, $i \in N$, и является NP -трудной в сильном смысле в противном случае [470, 471].

9.5. Следует отметить, что условие запрещения задержек, вообще говоря, не эквивалентно условию отсутствия промежуточных емкостей складирования между соседними приборами. В последнем случае требование i , $i \in N$, не обязательно обслуживается без задержек, а может "пролеживать" на приборе L , $1 \leq L \leq M - 1$, после завершения его обслуживания этим прибором в ожидании освобождения очередного прибора, причем во время пролеживания требования i прибор L не может обслуживать никакого другого требования. Если все приборы обслуживают требования в одной и той же последовательности, то задача минимизации $\bar{t}_{\max}(s)$ допускает естественную сетевую интерпретацию, предложенную Е.В. Левнером [177]. Нижние и верхние оценки общего времени обслуживания для этой задачи получены В.А. Струсовичем [316]. При $M = 2$ условия запрещения задержек и отсутствия промежуточных емкостей складирования эквивалентны [177, 749]. Случай $M = 3$ при отсутствии емкостей только между приборами 2 и 3 рассмотрен в [46].

Определенный интерес представляет ситуация, когда между соседними приборами существуют емкости складирования ограниченной вместимости b , $0 < b < n - 1$ (случай $b \geq n - 1$ равносильен "обычному" характеру обслуживания). Задача построения оптимального по быстродействию расписания является NP -трудной при $M \geq 2$ и любом b , $0 < b < n - 1$ [734]. Для ее решения при $M = 2$ С. Датта и А. Кэннингхам [510] предложили алгоритм, основанный на идеях динамического программирования, а С. Редди [748] — алгоритм трудоемкости $O(n^{2^{c+1}})$, где $c = n - 2 - b > 0$.

9.6. В [93, 290, 409] рассматривается ситуация, когда для каждого требования порядок прохождения приборов задается последовательностью $1, 2, \dots, M$, повторенной определенное количество раз. В.Г. Тимковский [354, 356] исследовал сложность задачи построения расписаний для таких систем и предложил приближенные алгоритмы ее решения. В. Лев и А. Адири [660] исследовали сложность построения расписаний для систем, в которых маршрут обслуживания каждого требования определяется последовательностью $1, 2, \dots, M - 1, M, M - 1, \dots, 2, 1$.

Ряд публикаций посвящены исследованию систем с операторами переноса. Вопросы совмещения примитивных циклических процессов рассматривались Д.А. Супруненко, Р.И. Тышкевич, В.С. Айзенштатом и А.С. Метельским [4, 5, 207–209, 328, 330]. Процессы с ограниченным и неограниченным числом операторов переноса изучены А.Ш. Блохом, В.С. Танаевым и И.В. Романовским [27, 257, 335, 351]. (См. также [85, 86, 167, 168, 227, 283].)

9.7. NP -трудность задачи минимизации максимального временного смещения установлена Я. Ленстрой, А. Гинной Каном и П. Бруккёром [657] для случая $M \geq 2$ при запрещении прерываний.

В основу п. 4.1 положена работа Ю. Чо и С. Сахни [493]. Результаты, изложенные в п. 4.2, получены Х. Реком [758]. Более подробные сведения об эйлеровых циклах и контурах содержатся в монографиях [111, 232,

377]. Если в любой момент времени для обслуживания каждого требования каждым прибором необходимо использовать один вид ресурса, то задача минимизации $L_{\max}(s)$ является NP -трудной в сильном смысле при $M \geq 2$ как в случае, когда разрешаются прерывания, так и в случае, когда они запрещены, а также при запрещении задержек [758]. Отметим, что NP -трудность указанных задач означает также NP -трудность соответствующих задач минимизации общего времени обслуживания неодновременно поступающих в систему требований.

9.8. Доказательство, приведенное в п. 5.1, получено М. Гэри, Д. Джонсоном и Р. Сети [533]. Детальное доказательство утверждения из п. 5.2 можно найти в [758]. Если разрешены прерывания, то, как указано в [635], задача минимизации суммарного времени обслуживания является NP -трудной в сильном смысле при $M \geq 3$ (доказательство не опубликовано).

Если все $t_{iL} \in \{0, 1\}$ и запись $t_{iL} = 0$ означает, что требование i прибором L не обслуживается, то задача минимизации $\sum_{i=1}^n \bar{t}_i(s)$ является NP -трудной при переменном числе приборов независимо от того, допускаются ли прерывания и запрещены ли задержки [550]. Аналогичный результат справедлив и для минимизации общего времени обслуживания.

Анализ сложности задачи минимизации суммарного взвешенного времени обслуживания упорядоченного множества требований при $M = 2$ и $t_{i1} = t_{i2} = 1$, $i \in N$, проведен в [653].

9.9. Упорядоченные матрицы длительностей обслуживания введены в рассмотрение М. Смитом, С. Панвалкаром и Р. Дудеком [776]. Результаты, приведенные в п. 6.1, содержатся в работах [776, 777] (см. также [778]). Эффективные алгоритмы решения некоторых других задач теории расписаний с упорядоченными матрицами можно найти в [618, 729].

Ситуации, описанные в п. 6.2, 2) и п. 6.2, 3), рассмотрены соответственно С. Панвалкаром, А. Ханом [727] и О.И. Мельниковым, Я.М. Шафранским [205]. Результат из п. 6.2, 1) получен В. Шварцем [810].

Тот факт, что перестановка, которой соответствует оптимальное по быстродействию расписание (без задержек) в случае упорядоченной по строкам матрицы длительности обслуживания, принадлежит классу пирамидальных перестановок (см. п. 6.3, 1), 2)), установили Р. Арора и С. Рана [427]. В их работе приведен алгоритм построения оптимальной перестановки, имеющий трудоемкость $O(n^3 M)$. Менее трудоемкий алгоритм, описанный в п. 6.3, 3), предложен С. Аксатером [440]. Результаты, изложенные в п. 6.3, 4) и п. 6.4, представляют собой некоторое обобщение работ С. Панвалкара и С. Вуллама [731, 732].

Задачу построения оптимального по быстродействию расписания с упорядоченной по столбцам матрицей длительностей обслуживания рассмотрели Ф. Чин и Л. Цай [492]. В этом случае полиномиальные алгоритмы нахождения оптимальной перестановки удастся построить лишь при некоторых дополнительных предположениях. Случай $M = 3$ исследовался рядом авторов [24, 69, 90, 228, 236, 237, 413, 485]. Отметим алгоритм трудоемкости $O(n \log n)$ для случая $t_{i2} \leq \min \{t_{i1}, t_{i3}\}$, $i \in N$, предложенный Н.И. Глебовым [69], а также Ф. Бурнсом и Дж. Рукером [485]. Дж. Ачутбью и

Ф. Чин [413] предложили алгоритм трудоемкости $O(n^7)$ для случая $t_{i1} \leq t_{i2} \leq t_{i3}, i \in N$, либо $t_{i1} \geq t_{i2} \geq t_{i3}, i \in N$. Попытка построения алгоритма меньшей трудоемкости предпринята в [228]. В [229] предложен полиномиальный алгоритм построения оптимального по быстродействию расписания без задержек, если $M = 3$ и $t_{i1} \geq \max\{t_{i2}, t_{i3}\}, i \in N$, либо $t_{i3} \geq \max\{t_{i1}, t_{i2}\}, i \in N$.

9.10. Основу § 7 составили результаты, полученные В.А. Струсовичем [313–316, 318]. Для минимизации функционала (7.11) в [179, 325] предложены менее трудоемкие алгоритмы, чем приведенные в п. 7.3. Результаты, приведенные в § 7, являются обобщением ранее известных результатов, формулируемых в терминах последовательно доминирующих приборов (понятие доминирующих приборов рассматривается в § 2 гл. 3). Вопросы такого рода рассматривались в [41, 69, 170, 178, 179, 322, 414, 419, 428, 483, 484, 511, 583, 586, 609, 613, 796, 799, 800, 802, 803, 805, 808]. А. Смитс и К. Бейкер [781] провели вычислительный эксперимент с целью выяснения того, насколько часто выполняются те или иные условия, гарантирующие существование полиномиальных алгоритмов построения оптимального по быстродействию расписания при $M = 3$. Некоторые другие условия существования полиномиальных алгоритмов можно найти в [321, 700, 710].

9.11. Утверждение о том, что векторы v множества V , удовлетворяющие условиям из п. 8.2, 1), можно пронумеровать таким образом, что $V = \{v^1, v^2, \dots, v^r\}$ и

$$\max\{\|\sum_{k=1}^l v^k\| \mid l=1, \dots, r\} \leq c(m),$$

известно в отечественной литературе как лемма Штейница. Идея об использовании этого утверждения для приближенного решения задачи построения оптимального по быстродействию расписания высказана И.С. Беловым и Я.Н. Столиным [26], а также С.В. Севастьяновым [271, 272].

Сначала в работах [26, 73, 121, 272, 273] (см. также [215]) были предложены экспоненциальные по трудоемкости алгоритмы нахождения указанной нумерации векторов множества V и соответственно построения расписания s^0 с ограниченной абсолютной погрешностью $\delta = \bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*)$. Значение δ при этом также оказывалось довольно большим. И. Барани [460] предложил алгоритм построения расписаний s^0 с $\delta \leq (3M-1)(M-1)t^*/2$ не более чем за $O(n^2M^3 + nM^4)$ операций. С.В. Севастьянов [275], а также И. Барани и Т. Фиала [461], опираясь на результат, доказанный в [87], разработали алгоритмы построения s^0 , для которого $\delta \leq M(M-1)t^*$. Трудоемкость алгоритма из [461] составляет $O(n^2M^3)$ операций (см. п. 8.2). Как утверждается в работе [275], трудоемкость описанного в ней алгоритма не превышает $O(n^2M^2)$, если на одном из его этапов применять алгоритм из [274].

Для конкретных значений M известны алгоритмы с меньшей оценкой величины δ . Так для случая $M = 3$ известны алгоритмы трудоемкости $O(n \log n)$ и $O(n)$, гарантирующие построение расписания s^0 с $\delta \leq 3t^*$ и $\delta \leq 4t^*$ соответственно [96, 227]. Для $M = 4$ в [276] предложен алгоритм

трудоемкости $O(n^2)$ построения расписания s^0 с $\delta \leq 9t^*$. Более трудоемкие и (или) менее точные алгоритмы предложены в [96] (для $M = 4$) и в [89, 311, 519] (для $M = 3$). В [18, 276] обсуждается возможность использования данного подхода для приближенного решения других задач теории расписаний (см. также § 6 гл. 2 и § 2 гл. 3). Следует отметить также работу [462], где дана несколько иная интерпретация леммы Штейнница.

9.12. Основу пп. 8.3 — 8.6 составили результаты Т. Гонсалеса и С. Сахни [552], а также Х. Река и Г. Шмидта [759].

К. Потс [743] рассмотрел задачу построения оптимального по быстродействию расписания s^* при $M = 2$ и одновременном поступлении требований. Им предложены три алгоритма трудоемкости $O(n \log n)$ построения расписания s^0 , для которого относительная погрешность $\Delta = (\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*)) / \bar{t}_{\max}(s^*) \leq 1$, и алгоритм трудоемкости $O(n^3 \log n)$ построения расписания с $\Delta \leq 2/3$.

М.Я. Ковалев [138] разработал ϵ -приближенный алгоритм трудоемкости $O((n/\epsilon)^{2m-1})$ минимизации $\bar{t}_{\max}(s)$ для двухстадийной системы, на каждой стадии которой имеется по m параллельных (идентичных) пар приборов.

В [734] рассмотрена задача минимизации $\bar{t}_{\max}(s)$ при условии, что $M = 2$ и между приборами имеется емкость ограниченной вместимости b , $0 < b < n - 1$. Предложен алгоритм трудоемкости $O(n \log n)$ построения расписания s^0 с $\Delta \leq b/(b+1)$.

В.А. Струевич [317] рассмотрел задачу минимизации $\bar{t}_{\max}(s)$ при условии, что все приборы функционировать без промежуточных простоев. Описан алгоритм трудоемкости $O(nM \log n)$ построения расписания s^0 , при котором $(\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*)) / (\bar{t}_{\max}(s^*) + \sum_{i=1}^n \sum_{L=2}^{M-1} t_{iL}) \leq 1 - 1/(M-1)$.

О.И. Мельников и Я.М. Шафранский [206] рассмотрели задачу построения оптимального по быстродействию расписания при условии, что $M = 2$ и $t_{i1} = a_{i1} + \varphi(t_{i1}^0)$, $t_{i2} = a_{i2} + \varphi(t_{i2}^0)$, $i \in N$, где $\varphi(t)$ — монотонно возрастающая функция. Предложен полиномиальный алгоритм решения с апостериорной оценкой абсолютной погрешности. Сведения о некоторых других приближенных алгоритмах минимизации $\bar{t}_{\max}(s)$ с априорными и апостериорными оценками погрешности приведены в [1, 84, 356]. Асимптотическое поведение расписаний для этой и некоторых других задач исследовано В.М. Португалом [248–250] (см. также [386]). В [680] показано, что если при расписании s^0 требования упорядочены по неубыванию директивных сроков и $M = 2$, то $(L_{\max}(s^0) - L_{\max}(s^*)) / (L_{\max}(s^*) + \max\{D_i \mid i \in N\}) \leq 1$.

9.13. Большинство задач теории расписаний относится к классу NP -трудных (по оценкам, сделанным в [755], их доля составляет около 80 % от общего числа задач). Для решения таких задач разработан ряд алгоритмов, основанных на общих идеях метода последовательного конструирования, анализа и отсеивания вариантов. Многие из этих алгоритмов специально разработаны для систем поточного типа. Их обзор приведен в § 6 гл. 4.

В ряде публикаций предлагается сводить задачи построения оптимальных расписаний к решению задач (целочисленного) линейного и нелинейного программирования [58, 152, 224, 312, 433, 459, 541, 677] и теории игр [459]. Как правило, размерность возникающих при этом задач настолько велика, что получить точное решение за приемлемое время весьма затруднительно. Для нахождения приближенного решения в [224, 541] предлагается отказываться от условия целочисленности переменных с последующим округлением полученного решения.

Использование метода динамического программирования для построения оптимальных расписаний оказывается оправданным лишь при малых значениях M ($M \leq 3$) [25, 57, 95, 411, 510, 513].

Для нахождения приближенного решения рассматриваемых в этой главе задач разработано большое количество эвристических алгоритмов. Для задачи минимизации общего времени обслуживания предлагаемые алгоритмы, как правило, состоят в назначении каждому требованию специальным образом вычисляемого приоритета и последующего упорядочения требований согласно этим приоритетам. При другом подходе в результате многократного применения алгоритма из п. 1.3 строится несколько перестановок требований, из которых выбирается наилучшая. Эвристические процедуры, основанные на этих подходах, описаны в [134, 230, 233, 241, 312, 415, 486, 496, 579, 581, 584, 589, 712, 718, 719, 721, 735, 788, 811]. Обзоры, содержащие результаты обширных вычислительных экспериментов, опубликованы в [415, 496, 663, 735]. Ряд других эвристических процедур предложен в [28, 197, 255, 279, 534, 628, 629, 664, 763].

А.И. Бабушкин и И.С. Белов [19] разработали алгоритм трудоемкости $O(n^7)$ построения наилучшей перестановки требований для $M = 3$ из числа тех, которые могут быть получены в результате комбинирования двух произвольных правил назначения приоритетов. В этой же работе указаны два таких правила, комбинируя которые, за $O(n^3)$ операций можно построить расписание s^0 с абсолютной погрешностью $\delta \leq 4t^*$. Обсуждаются также обобщения этого подхода на случай произвольного M .

Эвристические процедуры минимизации суммарного и взвешенного суммарного времени обслуживания описаны в [695] и в [674, 694] соответственно.

За исключением работы [473] в литературе не рассматриваются методы, специально предназначенные для решения задачи минимизации общего времени обслуживания при запрещении задержек. Это объясняется тем, что указанная задача сводится к задаче о коммивояжере, которой посвящена весьма обширная библиография [605, 810].

Декомпозиционный подход к решению задачи минимизации $\bar{t}_{\max}(s)$ обсуждается в [429, 588, 772]. В последней работе излагаются общие принципы построения декомпозиционных алгоритмов и их связь с методом ветвей и границ. В.Н. Владимиров и С.В. Егоров [55] предложили декомпозиционный метод решения существенно более общей задачи.

Разнообразные вероятностные методы построения расписаний для систем поточного типа предложены в [9, 40, 75, 135, 331–334, 375, 597, 720]. Задачи со случайными длительностями обслуживания рассматривались в [42, 504, 631, 676, 723, 739, 740, 744, 835].

Множество $N = \{1, 2, \dots, n\}$ требований обслуживается в системе, состоящей из M приборов. Требование $i \in N$ поступает в систему в момент времени $d_i \geq 0$. Обслуживание этого требования включает r_i последовательных стадий. На стадии q , $1 \leq q \leq r_i$, требование i обслуживается прибором $L_q^i \in \mathcal{M} = \{1, 2, \dots, M\}$. Приборы L_q^i , $1 \leq q \leq r_i$, не обязательно различны. Маршруты прохождения приборов $L^i = (L_1^i, L_2^i, \dots, L_{r_i}^i)$ предполагаются заданными для всех требований $i \in N$ и могут быть различными для разных требований. Каждый прибор обслуживает требования последовательно, причем не более одного требования в каждый момент времени.

Процесс обслуживания требования отдельным прибором на некоторой конкретной стадии обычно называется операцией. В этой терминологии процесс обслуживания требования i состоит в последовательном выполнении r_i операций. Каждая операция характеризуется упорядоченной тройкой чисел (i, L, q) , $i \in N$, $L \in \mathcal{M}$, $1 \leq q \leq r_i$. Длительности $t_{iLq} > 0$ выполнения всех операций предполагаются заданными.

В зависимости от постановки задачи прерывания в выполнении каждой операции могут либо допускаться, либо не допускаться. Если прерывания запрещены, то расписание обслуживания требований однозначно определяется моментами начала t_{iLq}^0 или завершения $\bar{t}_{iLq}$ выполнения всех операций. В этом случае $\bar{t}_{iLq} = t_{iLq}^0 + t_{iLq}$.

Момент начала (завершения) обслуживания требования i в системе при расписании s будем обозначать через $t_i^0(s)$ (соответственно $\bar{t}_i(s)$). Очевидно, что $t_i^0(s) = t_{iL_1}^0$ и $\bar{t}_i(s) = t_{iL_{r_i}}^0$ независимо от того, допускаются прерывания или нет.

При доказательстве NP -трудности задач теории расписаний в данной главе используются следующие эталонные задачи.

Задача о разбиении. Дано множество $N^0 = \{1, 2, \dots, n_0\}$, каждому элементу i которого сопоставлено число e_i , и $\sum_{i \in N^0} e_i = 2E$, где e_i и E — натуральные числа. Пусть $N_k^0 \subset N^0$, тогда обозначим $E_k = \sum_{i \in N_k^0} e_i$.

Требуется определить, существует ли разбиение множества N^0 на два подмножества N_1^0 и N_2^0 таких, что $E_1 = E_2$.

Задача о 3-разбиении. Даны множество $N^0 = \{1, 2, \dots, 3n_0\}$ и натуральное число E . Каждому элементу $i \in N^0$ сопоставлено натуральное число e_i такое, что $E/4 < e_i < E/2$ и $\sum_{i \in N^0} e_i = n_0 E$. Существует ли такое

разбиение множества N^0 на n_0 трехэлементных подмножеств N_j^0 , что $\sum_{i \in N_j^0} e_i = E, j = 1, \dots, n_0$?

Задача о трехмерном паросочетании. Пусть задано конечное множество $N^0 = \{1, 2, \dots, n_0\}$ и $E \subseteq N^0 \times N^0 \times N^0$ — множество упорядоченных троек элементов из N^0 . Существует ли такое подмножество $E' \subseteq E$, называемое трехмерным паросочетанием, что $|E'| = n_0$ и две любые тройки, вошедшие в E' , не совпадают ни по одной компоненте?

Задача о разбиении и задача о трехмерном паросочетании являются NP -полными [133], задача о 3-разбиении NP -полна в сильном смысле [531].

§ 1. Оптимальное обслуживание двух требований

В систему, состоящую из M приборов, в момент времени $d = 0$ поступают два требования, т.е. $N = \{1, 2\}$. Известны маршруты $L^i = (L_1^i, L_2^i, \dots, L_{r_i}^i)$ обслуживания требований $i \in N$ и длительности выполнения всех операций.

Качество расписания s характеризуется значением заданной неубывающей по каждому из аргументов действительной функции $F(t_1, t_2)$ при $t_1 = \bar{t}_1(s)$ и $t_2 = \bar{t}_2(s)$. Расписание s^* считается *оптимальным*, если ему соответствует наименьшее значение указанной функции.

Поскольку $F(t_1, t_2)$ — неубывающая функция, то при поиске расписания s^* можно ограничиться рассмотрением класса S расписаний s , при которых в каждый момент времени из интервала $(0, \max\{\bar{t}_1(s), \bar{t}_2(s)\}]$ обслуживается хотя бы одно требование.

1.1. Опишем широко используемую в этом параграфе форму *графического представления* расписаний.

Пусть $x_u = \sum_{j=1}^u t_{1L_j^1} j, u = 1, \dots, r_1, y_v = \sum_{j=1}^v t_{2L_j^2} j, v = 1, \dots, r_2, x_0 = 0, y_0 = 0$. Введем в рассмотрение прямоугольную систему координат на плоскости. Изобразим в этой системе прямоугольник W с вершинами $(0, 0), (X, 0), (0, Y), (X, Y)$, где $X = x_{r_1}, Y = y_{r_2}$. Под длиной отрезка, соединяющего точки (x, y) и (x', y') из W , будем понимать величину $\rho((x, y), (x', y')) = \max\{|x' - x|, |y' - y|\}$, а под длиной ломаной линии — величину, равную сумме длин составляющих ее отрезков.

Расписание s будем представлять в виде некоторой непрерывной ломаной линии (*траектории*) σ в прямоугольнике W , соединяющей вершины $(0, 0)$ и (X, Y) . Пусть точка $(x, y) \in \sigma, x_{u-1} \leq x \leq x_u, 1 \leq u \leq r_1, y_{v-1} \leq y \leq y_v, 1 \leq v \leq r_2$, и $t = \rho_\sigma((0, 0), (x, y))$ — длина траектории σ от точки $(0, 0)$ до точки (x, y) .

Это означает, что при расписании s к моменту времени t завершено выполнение операций $(1, L_1^1, 1), (1, L_2^1, 2), \dots, (1, L_{u-1}^1, u-1)$ и $(2, L_1^2, 1), (2, L_2^2, 2), \dots, (2, L_{v-1}^2, v-1)$, а на выполнение операций $(1, L_u^1, u)$ и $(2, L_v^2, v)$ затрачено $x - x_{u-1}$ и $y - y_{v-1}$ единиц времени соответственно.

По условию задачи составляющими траектории σ могут быть отрезки следующих трех видов: *горизонтальные* (обслуживается требование 1), *вертикальные* (обслуживается требование 2) и *наклонные* под углом 45° к оси абсцисс (одновременно обслуживаются оба требования).

Через W_{uv} будем обозначать прямоугольник с вершинами $(x_{u-1}, y_{v-1}), (x_u, y_{v-1}), (x_{u-1}, y_v), (x_u, y_v)$. Выделим все прямоугольники W_{uv} с $L_u^1 = L_v^2$. Эти прямоугольники и их внутренние точки будем называть *особыми*.

Поскольку каждый прибор в любой момент времени может обслуживать не более одного требования, то наклонные отрезки траектории σ не могут содержать особых точек. Если прерывания при выполнении любой операции запрещены, то траектория σ вообще не может содержать указанных точек. В этом случае она может проходить только вне особых прямоугольников и по их границам.

1.2. П р и м е р. На стендовом полигоне необходимо провести испытания двух новых изделий. Для каждого изделия (требования) известен порядок (маршрут) прохождения испытательных стендов (приборов). На полигоне имеется по одному испытательному стенду A, B, C и D . Порядки и длительности испытаний на стендах первого и второго изделий заданы соответственно в табл. 1.1 и табл. 1.2.

Прямоугольник W для этого примера изображен на рис. 1.1. Значения $x_u, u = 1, 2, 3, 4, 5$, и значения $y_v, v = 1, 2, 3, 4, 5, 6$, отложены соответственно на оси абсцисс и оси ординат. Особые прямоугольники заштрихованы.

На этом же рисунке изображены траектории σ_1 и σ_2 , для расписаний s_1 и s_2 , представленных на рис. 1.2, a и b соответственно. При расписании s_1 в процессе выполнения операций нет прерываний. Поэтому траектория σ_1

Таблица 1.1

Порядок испытаний	A	B	C	D	A
Длительности испытаний	2	2	8	3	6

Таблица 1.2

Порядок испытаний	A	B	A	C	A	C
Длительности испытаний	1	3	2	4	4	2

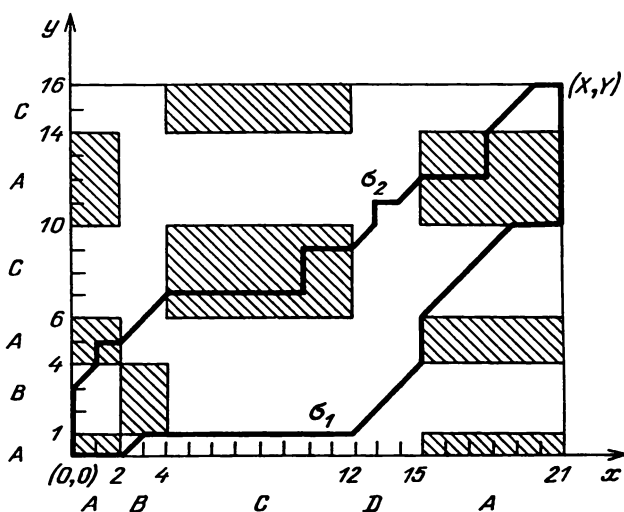


Рис. 1.1

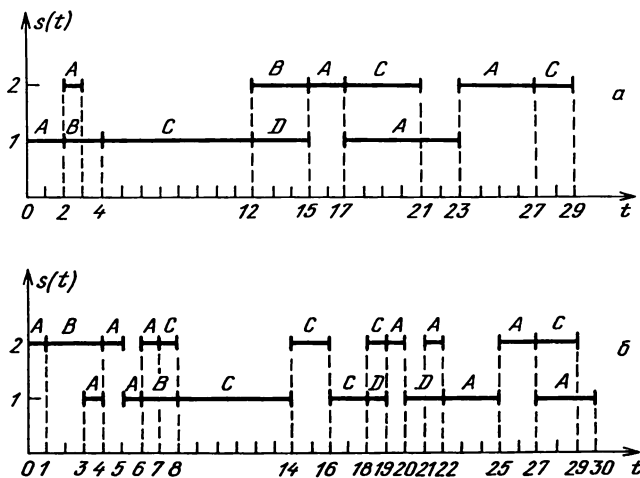


Рис. 1.2

проходит либо вне, либо по границам заштрихованных прямоугольников. При расписании s_2 допускаются прерывания. В частности, при выполнении операции $(2, C, 4)$ имеют место прерывания в моменты времени $t_1 = \rho_{\sigma_2}((0, 0), (4, 7)) = 8$ и $t_2 = \rho_{\sigma_2}((0, 0), (10, 9)) = 16$.

Моменты завершения обслуживания требований при расписании s_1 и расписании s_2 принимают следующие значения:

$$\bar{t}_1(s_1) = \rho_{\sigma_1}((0, 0), (21, 10)) = 23,$$

$$\bar{t}_2(s_1) = \rho_{\sigma_1}((0, 0), (21, 16)) = 29,$$

$$\bar{t}_1(s_2) = \rho_{\sigma_2}((0, 0), (21, 16)) = 30,$$

$$\bar{t}_2(s_2) = \rho_{\sigma_2}((0, 0), (20, 16)) = 29.$$

1.3. Рассмотрим задачу нахождения оптимального расписания s^* при условии, что прерывания в процессе выполнения каждой операции допускаются в любой момент времени. Как уже отмечалось, расписание s^* содержится в классе S расписаний, каждое из которых может быть представлено некоторой траекторией σ в прямоугольнике W . При этом наклонные отрезки траектории σ не содержат особых точек.

Покажем, что задача нахождения расписания s^* (траектории σ^*) сводится к задаче нахождения путей наименьшей длины в некотором ориентированном графе $G = (Q, R)$, где Q — множество вершин (некоторых точек из прямоугольника W), R — множество дуг.

Опишем алгоритм последовательного конструирования графа G . Сопоставим вершинам (x_u, y_{v-1}) и (x_{u-1}, y_v) каждого особого прямоугольника W_{uv} параметры α_{uv} и β_{uv} соответственно. Эти параметры будут принимать значения 1 или 0 в зависимости от того, включена или не включена соответствующая вершина прямоугольника во множество вершин конструируемого графа G .

На первом шаге полагаем $Q^0 = \{(x_0, y_0), (X, Y)\}$, $R^0 = \emptyset$, $\alpha_{uv} = \beta_{uv} = 0$ для всех особых прямоугольников W_{uv} .

Если прямоугольник W_{11} является особым, то полагаем $Q^1 = Q^0 \cup \{(x_0, y_1), (x_1, y_0)\}$, $R^1 = \{((x_0, y_0), ((x_0, y_1))), ((x_0, y_0), (x_1, y_0))\}$ и $\alpha_{11} = \beta_{11} = 1$. В противном случае выполним следующую процедуру.

Из точки (x_0, y_0) проводим наклонную (под углом 45° к оси абсцисс) прямую до тех пор, пока не встретится особая точка либо не будет достигнута граница прямоугольника W . Пусть указанная особая точка принадлежит прямоугольнику W_{uv} и (x, y) — ближайшая к началу координат точка пересечения наклонной прямой с границей прямоугольника W_{uv} .

Если $x = x_{u-1}$ и $y = y_{v-1}$, то полагаем $Q^1 = Q^0 \cup \{(x, y), (x_{u-1}, y_v), (x_u, y_{v-1})\}$, $R^1 = R^0 \cup \{((x_0, y_0), (x_{u-1}, y_{v-1})), ((x_{u-1}, y_{v-1}), (x_u, y_{v-1})), ((x_{u-1}, y_{v-1}), (x_{u-1}, y_v))\}$ и $\alpha_{uv} = \beta_{uv} = 1$. Если $x \neq x_{u-1}$ и $y = y_{v-1}$, то полагаем $Q^1 = Q^0 \cup \{(x, y), (x, y_v), (x_u, y_{v-1})\}$, $R^1 = R^0 \cup \{((x_0, y_0), (x, y)), ((x, y), (x, y_v)), ((x, y), (x_u, y_{v-1}))\}$ и $\alpha_{uv} = 1$. Если $x = x_{u-1}$ и $y \neq y_{v-1}$, то полагаем $Q^1 = Q^0 \cup \{(x, y), (x_u, y), (x_{u-1}, y_v)\}$, $R^1 = R^0 \cup \{((x_0, y_0), (x, y)), ((x, y), (x_u, y)), ((x, y), (x_{u-1}, y_v))\}$ и $\beta_{uv} = 1$.

Пусть достигнута граница прямоугольника W и (x, y) — точка пересечения наклонной прямой с этой границей, тогда полагаем $Q^1 = Q^0 \cup \{(x, y)\}$ и $R^1 = R^0 \cup \{((x_0, y_0), (x, y)), ((x, y), (X, Y))\}$.

Для выполнения l -го шага алгоритма выбираем во множестве Q^{l-1} отличную от (X, Y) вершину (x', y') , из которой не исходит ни одной дуги множества R^{l-1} . Если такая вершина существует, то к точке (x', y') , множеству вершин Q^{l-1} и множеству дуг R^{l-1} применим процедуру, аналогичную той, которая применялась к точке (x_0, y_0) , множеству вершин Q^0 и множеству дуг R^0 . При этом если $y = y_{v-1}$ и $\alpha_{uv} = 1$, т.е. вершина $(x_u, y_{v-1}) \in Q^{l-1}$, то полагаем $Q^l = Q^{l-1} \cup \{(x, y), (x, y_v)\}$. Аналогично, если $x = x_{u-1}$ и $\beta_{uv} = 1$, то полагаем $Q^l = Q^{l-1} \cup \{(x, y), (x_u, y)\}$.

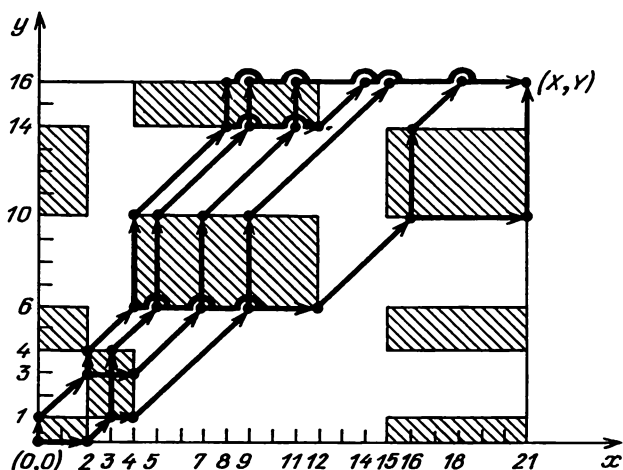


Рис. 1.3

Если (X, Y) — единственная вершина из множества Q^{l-1} с нулевой полустепенью исхода, то искомым граф $G = (Q, R)$ построен: $Q = Q^{l-1}$ и $R = R^{l-1}$.

1.4. В условиях описанного в п. 1.2 примера граф G представлен на рис. 1.3.

1.5. Обозначим через $\sigma(G)$ множество всех траекторий — путей в графе G из вершины (x_0, y_0) в вершину (X, Y) . По построению каждая траектория $\sigma \in \sigma(G)$ представляет некоторое расписание $s \in S$. Это расписание будем обозначать через $s(\sigma)$. Аналогично, если $s \in S$ и σ — соответствующая ему траектория в прямоугольнике W , то эту траекторию будем обозначать через $\sigma(s)$.

Покажем, что существует оптимальное расписание $s^* = s(\sigma^*)$, где $\sigma^* \in \sigma(G)$. Доказательство проведем по индукции относительно числа h особых прямоугольников, которые встретились в процессе построения графа G .

При $h = 0$ множество $\sigma(G)$ состоит из единственной траектории вида $\sigma = ((x_0, y_0), (x, y), (X, Y))$, где либо $x = X$, либо $y = Y$, либо $\sigma = ((x_0, y_0), (X, Y))$. Поскольку для любого расписания $s \in S$ выполняются соотношения $\bar{t}_1(s) \geq X = \bar{t}_1(s(\sigma))$, $\bar{t}_2(s) \geq Y = \bar{t}_2(s(\sigma))$ и функция $F_1(t_1, t_2)$ неубывающая, то $s(\sigma) = s^*$.

Предположим, что утверждение справедливо при всех $h \leq h_0$, и докажем его справедливость при $h = h_0 + 1$. Пусть на шаге 1 описанного алгоритма первая особая точка, которую встретит наклонная прямая, проведенная из точки (x_0, y_0) , принадлежит прямоугольнику W_{uv} . Предположим для определенности, что $\max\{x_{u-1}, y_{v-1}\} = y_{v-1}$ (рис. 1.4).

Через W_1 обозначим прямоугольник с вершинами (x_u, y_{v-1}) , (X, y_{v-1}) , (x_u, Y) , (X, Y) , а через W_2 — прямоугольник с вершинами (y_{v-1}, y_v) , (X, y_v) , (y_{v-1}, Y) , (X, Y) . Пусть $F_1(t_1, t_2) = F(t_1 + x_u, t_2 + x_u)$ и $F_2(t_1, t_2) = F(t_1 + y_v, t_2 + y_v)$. Рассмотрим две новые задачи: задачу 1 поис-

ка расписания s_1^* при $x_0 = x_u, y_0 = y_{v-1}, W = W_1$ и функции $F(t_1, t_2) = F_1(t_1, t_2)$ и задачу 2 поиска расписания s_2^* при $x_0 = y_{v-1}, y_0 = y_v, W = W_2$ и функции $F(t_1, t_2) = F_2(t_1, t_2)$.

По индуктивному предположению существуют расписания $s_1^* = s(\sigma_1^*)$ и $s_2^* = s(\sigma_2^*)$, для которых $\sigma_1^* \in \sigma(G_1), \sigma_2^* \in \sigma(G_2)$ и графы $G_1 = (Q_1, R_1)$ и $G_2 = (Q_2, R_2)$ построены по описанному в п. 1.3 алгоритму для задачи 1 и 2 соответственно.

Обозначим $\sigma_1' = ((0, 0), (y_{v-1}, y_{v-1}), (x_u, y_{v-1}))$, $\sigma_2' = ((0, 0), (y_{v-1}, y_{v-1}), (y_{v-1}, y_v))$ и докажем, что хотя бы одно из расписаний $s(\sigma_1', \sigma_1^*)$ или $s(\sigma_2', \sigma_2^*)$ является оптимальным для исходной задачи.

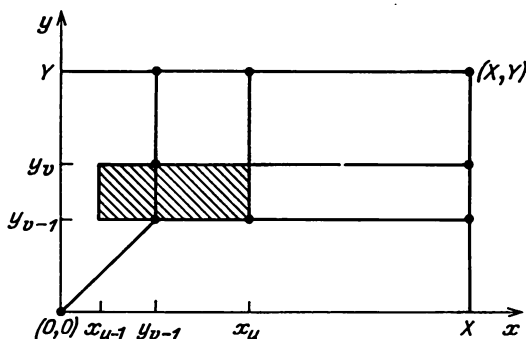


Рис. 1.4

Предположим, что существует расписание $s \in S$, для которого выполняется неравенство

$$F(\bar{t}_1(s), \bar{t}_2(s)) = C < \min \{F(\bar{t}_1(s(\sigma_1'), \sigma_1^*)), \bar{t}_2(s(\sigma_1', \sigma_1^*)), F(\bar{t}_1(s(\sigma_2'), \sigma_2^*)), \bar{t}_2(s(\sigma_2', \sigma_2^*))\}. \quad (1.1)$$

Траектория $\sigma(s)$ должна пересечь один из следующих отрезков: а) $((x_u, y_{v-1}), (X, y_{v-1}))$, б) $((x_u, y_{v-1}), (x_u, y_v))$, в) $((y_{v-1}, y_v), (x_u, y_v))$ или г) $((y_{v-1}, y_v), (y_{v-1}, Y))$.

Рассмотрим, например, случай, когда траектория $\sigma(s)$ имеет общую точку (x_u, y') с отрезком $((x_u, y_{v-1}), (x_u, y_v))$. Обозначим часть траектории $\sigma(s)$ от точки $(0, 0)$ до точки (x_u, y') через σ' , а от точки (x_u, y') до точки (X, Y) — через σ'' . Положим $\sigma'_0 = (\sigma'_1, (x_u, y'))$ и $\sigma''_0 = ((x_u, y_{v-1}), \sigma'')$.

Поскольку прямоугольник W_{uv} особый, то $\rho_{\sigma'}((0, 0), (x_u, y')) \geq \rho_{\sigma''}((0, 0), (x_u, y'))$ и в силу монотонности функции $F(t_1, t_2)$ выполняется неравенство

$$F(\bar{t}_1(s(\sigma'_0, \sigma'')), \bar{t}_2(s(\sigma'_0, \sigma''))) \leq F(\bar{t}_1(s(\sigma)), \bar{t}_2(s(\sigma))) = C.$$

Отсюда и из соотношения (1.1) получаем

$$F_1(\bar{t}_1(s(\sigma'_0, \sigma'')), \bar{t}_2(s(\sigma'_0, \sigma''))) = F(\bar{t}_1(s(\sigma'_0, \sigma'')), \bar{t}_2(s(\sigma'_0, \sigma''))) \leq C < F(\bar{t}_1(s(\sigma'_1, \sigma_1^*)), \bar{t}_2(s(\sigma'_1, \sigma_1^*))) = F_1(\bar{t}_1(s(\sigma_1^*)), \bar{t}_2(s(\sigma_1^*))),$$

что противоречит оптимальности расписания $s(\sigma_1^*)$ для задачи 1.

Если траектория $s(\sigma)$ пересекает отрезок а), то получаем аналогичное соотношение, которое противоречит предположению об оптимальности расписания $s(\sigma_1^*)$ для задачи 1. Если же траектория $s(\sigma)$ пересекает отрезок в) или г), то получаем противоречие с предположением об оптимальности расписания $s(\sigma_2^*)$ для задачи 2.

Следовательно, не существует расписания $s \in S$, для которого выполняется соотношение (1.1). Поскольку $Q = Q_1 \cup Q_2 \cup \{(0, 0), (y_{v-1}, y_{v-1})\}$ и $R = R_1 \cup R_2 \cup \{((0, 0), (y_{v-1}, y_{v-1})), ((y_{v-1}, y_{v-1}), (x_u, y_{v-1})), ((y_{v-1}, y_{v-1}), (y_{v-1}, y_v))\}$, то $(\sigma_1^*, \sigma_1^*) \in s(G)$ и $(\sigma_2^*, \sigma_2^*) \in s(G)$, что завершает доказательство утверждения.

1.6. Каждая траектория $\sigma \in \sigma(G)$ имеет вид $\sigma = ((0, 0), \dots, (x, y), (X, Y))$, где а) $x = X$, $y < Y$, либо б) $x < X$, $y = Y$, либо в) $x < X$, $y < Y$. Обозначим длину траектории σ от точки $(0, 0)$ до точки (x, y) через $\hat{t}$. В зависимости от вида траектории моменты завершения обслуживания требований при расписании $s = s(\sigma)$ вычисляются по формулам

$$\text{а) } \bar{t}_1(s) = \hat{t}, \quad \bar{t}_2(s) = \hat{t} + Y - y;$$

$$\text{б) } \bar{t}_1(s) = \hat{t} + X - x, \quad \bar{t}_2(s) = \hat{t};$$

$$\text{в) } \bar{t}_1(s) = \bar{t}_2(s) = \hat{t} + \rho_\sigma((x, y), (X, Y)) = \rho_\sigma((0, 0), (X, Y)).$$

Поскольку $F(t_1, t_2)$ — неубывающая функция, то при поиске оптимальной траектории σ^* среди всех траекторий с фиксированной точкой (x, y) следует выбирать траекторию с наименьшим значением $\hat{t}$.

Таким образом, задача нахождения траектории σ^* сводится к

1) построению графа $G = (Q, R)$, выделению всех его вершин вида (X, y) и (x, Y) ;

2) нахождению путей наименьшей длины из вершины $(0, 0)$ в каждую из выделенных вершин, вычислению значений $\bar{t}_1(s)$ и $\bar{t}_2(s)$ по приведенным выше формулам;

3) вычислению значений функции $F(t_1, t_2)$ при $t_1 = \bar{t}_1(s)$ и $t_2 = \bar{t}_2(s)$ и выбору пути, которому соответствует наименьшее из этих значений.

1.7. Возвращаясь к примеру из п. 1.2, рассмотрим задачу построения оптимального расписания s^* (при разрешении прерываний операций), если $F(t_1, t_2) = 30t_1 + t_2^2$.

Граф G для этого примера представлен на рис. 1.3. В результате поиска кратчайших путей из вершины $(0, 0)$ в вершины $(8, 16)$, $(9, 16)$, $(11, 16)$, $(14, 16)$, $(15, 16)$, $(18, 16)$, $(21, 10)$ и вычисления соответствующих значений функции $F(t_1, t_2)$ получаем $\sigma^* = ((0, 0), (2, 0), (3, 1), (4, 1), (9, 6), (9, 10), (15, 16), (21, 16))$, $\bar{t}_1(s^*) = 19 + (21 - 15) = 25$, $\bar{t}_2(s^*) = 19$, $F(\bar{t}_1(s^*), \bar{t}_2(s^*)) = 30 \cdot 25 + 19^2 = 1111$. Здесь $s^* = s(\sigma^*)$.

1.8. Перейдем к рассмотрению задачи нахождения оптимального расписания s^* , когда прерывания в процессе выполнения каждой операции запрещены. В этом случае расписание однозначно определяется моментами начала t_{iLq}^0 или завершения $\bar{t}_{iLq}$ выполнения операций, причем $\bar{t}_{iLq} = t_{iLq}^0 + t_{iLq}$.

Как и в предыдущем случае, расписание s^* содержится в классе S расписаний, каждое из которых s может быть представлено траекторией

$\sigma(s)$ в прямоугольнике W . Существенное отличие от рассмотренной в пп. 1.3–1.7 задачи состоит в том, что траектория $\sigma(s)$ может проходить только вне особых прямоугольников и по их границам.

Опишем алгоритм последовательного конструирования графа $\bar{G} = (\bar{Q}, \bar{R})$, в котором существует путь, определяющий оптимальное распределение s^* .

Сопоставим вершинам (x_u, y_{v-1}) и (x_{u-1}, y_v) каждого особого прямоугольника W_{uv} параметр γ_{uv} , который будет принимать значение 1 или 0 в зависимости от того, включены или нет указанные вершины прямоугольника во множество вершин конструируемого графа $\bar{G}$.

На первом шаге полагаем $\bar{Q}^0 = \{(x_0, y_0), (X, Y)\}$, $\bar{R}^0 = \emptyset$ и $\gamma_{uv} = 0$ для всех особых прямоугольников W_{uv} .

Если прямоугольник W_{11} является особым, то полагаем $\gamma_{11} = 1$, $\bar{Q}^1 = \bar{Q}^0 \cup \{(x_0, y_1), (x_1, y_0)\}$ и $\bar{R}^1 = \{((x_0, y_0), (x_0, y_1)), ((x_0, y_0), (x_1, y_0))\}$. В противном случае выполняем следующую процедуру.

Из точки (x_0, y_0) проводим наклонную (под углом 45° к оси абсцисс) прямую до тех пор, пока не встретится особая точка или не будет достигнута граница прямоугольника W . Если указанная особая точка принадлежит прямоугольнику W_{uv} , то полагаем $\bar{Q}^1 = \bar{Q}^0 \cup \{(x_{u-1}, y_v), (x_u, y_{v-1})\}$, $\bar{R}^1 = \bar{R}^0 \cup \{((x_0, y_0), (x_{u-1}, y_v)), ((x_0, y_0), (x_u, y_{v-1}))\}$ и $\gamma_{uv} = 1$.

Если достигнута граница прямоугольника W и (x, y) — точка пересечения наклонной прямой с этой границей, то полагаем $\bar{Q}^1 = \bar{Q}^0 \cup \{(x, y)\}$ и $\bar{R}^1 = \bar{R}^0 \cup \{((x_0, y_0), (x, y)), ((x, y), (X, Y))\}$.

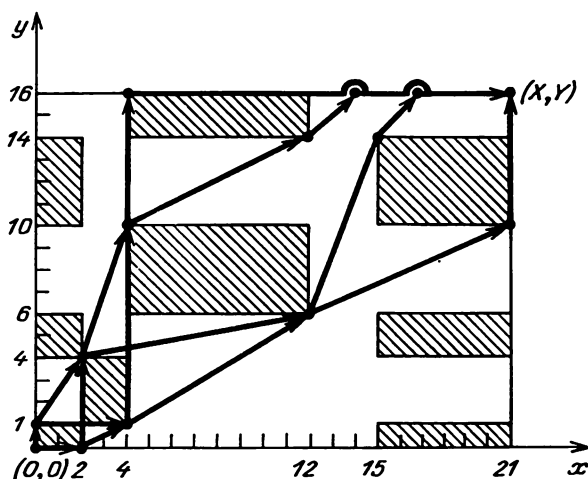
Для выполнения l -го шага алгоритма во множестве $\bar{Q}^{l-1}$ выбираем отличную от (X, Y) вершину (x', y') , из которой не исходит ни одной дуги множества $\bar{R}^{l-1}$. Если такая вершина существует, то к точке (x', y') , множеству вершин $\bar{Q}^{l-1}$ и множеству дуг $\bar{R}^{l-1}$ применим процедуру, аналогичную той, которая применялась к точке (x_0, y_0) , множеству вершин $\bar{Q}^0$ и множеству дуг $\bar{R}^0$. При этом если $\gamma_{uv} = 1$, то полагаем $\bar{Q}^l = \bar{Q}^{l-1}$, если же $\gamma_{uv} = 0$, то полагаем $\bar{Q}^l = \bar{Q}^{l-1} \cup \{(x_{u-1}, y_v), (x_u, y_{v-1})\}$.

Если (X, Y) — единственная вершина из множества $\bar{Q}^{l-1}$ с нулевой полустепенью исхода, то искомый граф $\bar{G} = (\bar{Q}, \bar{R})$ построен: $\bar{Q} = \bar{Q}^{l-1}$ и $\bar{R} = \bar{R}^{l-1}$.

1.9. Пример. На рис. 1.5 изображен граф $\bar{G}$ для примера, описанного в п. 1.2.

1.10. В отличие от графа G не каждый путь в графе $\bar{G}$ из вершины $(0,0)$ в вершину (X, Y) является траекторией. В качестве составляющих он может содержать наклонные отрезки с углом наклона к оси абсцисс, отличным от 45° .

Нетрудно показать, что каждый такой отрезок $((x_{u'}, y_{v'}), (x_u, y_v))$, $u' < u$, $v' < v$, можно заменить единственной ломаной линией, соединяющей те же точки $(x_{u'}, y_{v'})$ и (x_u, y_v) , имеющей длину, равную длине исходного отрезка, и состоящей из двух отрезков, первый из которых



$((x_u', y_v'), (x, y))$ имеет угол наклона 45° к оси абсцисс, а второй $((x, y), (x_u, y_v))$ является вертикальным ($x = x_u$) или горизонтальным ($y = y_v$).

Тем самым каждому пути $\bar{\sigma}$ в графе $\bar{G}$ сопоставляется единственная траектория σ в прямоугольнике W , проходящая через вершины этого пути и имеющая ту же длину. По построению графа $\bar{G}$ траектория σ не содержит особых точек и, следовательно, представляет некоторое расписание $s(\sigma) \in S$. Поскольку все вертикальные (горизонтальные) отрезки траектории σ имеют вид $((x, y), (x_u, y_v))$, где $x = x_u$ (соответственно $y = y_v$), то при расписании $s(\sigma)$ прерывания в процессе выполнения операций не допускаются. Обозначим это расписание через $s(\bar{\sigma})$.

Как и в случае задачи с прерываниями операций, можно показать, что существует оптимальное расписание $s^* = s(\bar{\sigma}^*)$, где $\bar{\sigma}^*$ – некоторый путь в графе $\bar{G}$.

Доказательство этого утверждения аналогично приведенному в п. 1.5. Отметим только, что для рассматриваемой задачи траектория $\sigma(s)$ должна иметь общую точку с отрезком $((x_u, y_{v-1}), (X, y_{v-1}))$ или отрезком $((x_{u-1}, y_v), (x_{u-1}, Y))$ для любого особого прямоугольника W_{uv} .

Как и в п. 1.6, можно показать, что в рассматриваемом случае задача нахождения расписания s^* также сводится к задаче нахождения кратчайших путей в графе $\bar{G} = (\bar{Q}, \bar{R})$ из вершины $(0,0)$ в каждую вершину $(x, y) \in \bar{Q}$, для которой $x = X$ или $y = Y$.

1.11. Пример. В результате поиска кратчайших путей в графе G (см. рис. 1.5) из вершины $(0,0)$ в вершины $(4,16)$, $(14,16)$, $(17,16)$, $(21,10)$ и вычисления соответствующих значений целевой функции $F(t_1, t_2) = 30t_1 + t_2^2$ получаем искомым путь $\bar{\sigma}^* = ((0,0), (0,1), (2,4), (4,10), (12,14), (14,16), (21,16))$, которому соответствует траектория $\sigma^* = ((0,0), (0,1), (2,3), (2,4), (4,6), (4,10), (8,14), (12,14), (14,16), (21,16))$. Зна-

чения $\bar{t}_1(s^*) = 27$, $\bar{t}_2(s^*) = 20$, $F(\bar{t}_1(s^*), \bar{t}_2(s^*)) = 30 \cdot 27 + 20^2 = 1210$.
Здесь $s^* = s(\bar{\sigma}^*)$.

1.12. Оценим сложность описанных алгоритмов построения оптимальных расписаний обслуживания двух требований M приборами.

Обозначим через Q_W множество вершин графа $G = (Q, R)$, являющихся вершинами особых прямоугольников W_{uv} . Поскольку число всех особых прямоугольников равно $r_1 r_2$, то $|Q_W|$ не превосходит $O(r^2)$, где $r = \max\{r_1, r_2\}$.

Множество вершин особых прямоугольников W_{uv} , для которых $\min\{u, v\} = k$, $k < r$, обозначим через Ω_k .

В графе G существует не более одного пути из каждой вершины (x, y) множества $Q \cap \Omega_k$ в вершину (X, Y) , не содержащего вершин множества Q_W , отличных от (x, y) и (X, Y) . Этот путь содержит не более чем $O(r - k)$ вершин множества $Q \setminus Q_W$. Поскольку $|\Omega_k|$ не превосходит $O(r - k)$, то все указанные пути содержат не более $O((r - k)^2)$ вершин множества $Q \setminus Q_W$. Следовательно, $|Q \setminus Q_W|$ не превосходит $O(\sum_{k=1}^r (r - k)^2) = O(r^3)$.

Поскольку число $|Q|$ вершин графа G не превосходит $O(r^3)$, число $|R|$ дуг не превосходит $2|Q|$ и граф не содержит контуров, то для выделения кратчайших путей из вершины $(0, 0)$ во все остальные вершины достаточно выполнить $O(r^3)$ элементарных действий.

Очевидно, что для построения графа G требуется не более $O(r^3)$ действий.

Таким образом, оптимальное расписание (с прерываниями) может быть построено в результате выполнения $O(r^3)$ действий.

В случае, когда прерывания в процессе обслуживания требований запрещены, указанная оценка может быть уменьшена до $O(r^2 \log r)$.

Действительно, вершины графа $\bar{G} = (\bar{Q}, \bar{R})$ являются вершинами особых прямоугольников W_{uv} (за возможным исключением вершин $(0, 0)$ и (X, Y)). Следовательно, число $|\bar{Q}|$ вершин и число $|\bar{R}|$ дуг графа $\bar{G}$ не превосходит $O(r^2)$.

Опишем процедуру построения графа $\bar{G}$, трудоемкость которой не превосходит $O(r^2 \log r)$.

Пронумеруем особые прямоугольники W_{uv} в лексикографическом порядке пар индексов (u, v) . Если s — наибольший из полученных номеров, то номер $s + 1$ присвоим области $W' = \{(x, y) | x \geq 0, y \geq Y\}$, а номер $s + 2$ — области $W'' = \{(x, y) | x \geq X, 0 \leq y \leq Y\}$.

Через $\bar{\Omega}$ обозначим множество вершин вида (x_u, y_{v-1}) и (x_{u-1}, y_v) всех особых прямоугольников W_{uv} . Положим $\Omega = \bar{\Omega} \cup \{(0, Y), (X, Y)\}$. За $O(r^2 \log r)$ элементарных действий упорядочим вершины (x, y) из множества Ω в порядке неубывания величины $h(x, y) = x - y$.

Все наименьшие номера особых прямоугольников, с которыми могут встретиться (в процессе построения графа $\bar{G}$) наклонные прямые, проходящие через вершины множества $\bar{\Omega}$, можно определить в результате последовательного просмотра вершин множества Ω в порядке неубывания величины $h(x, y)$. Для этого будем использовать список λ упорядоченных по возрастанию номеров особых прямоугольников и областей W' и W'' ,

с которыми пересекается наклонная прямая, проходящая через рассматриваемую вершину из множества Ω . Перед просмотром первой вершины множества Ω список λ полагаем пустым.

При просмотре очередной вершины $(x, y) \in \Omega$ включаем в список λ номер прямоугольника W_{uv} , если $x = x_{u-1}$ и $y = y_v$, и исключаем номер этого прямоугольника, если $x = x_u$ и $y = y_{v-1}$. Кроме того, если $x = 0$ и $y = Y$, то в список λ включаем номер $s + 1$, а если $x = X$ и $y = Y$, то включаем номер $s + 2$ и исключаем из списка λ номер $s + 1$.

За $O(\log r)$ элементарных действий в упорядоченном списке λ можно найти наименьший номер s^* , который больше номера прямоугольника W_{uv} . Номер s^* соответствует первому особому прямоугольнику (или областям W', W''), с которыми "встретится" наклонная прямая, направленная из вершины (x, y) .

Таким образом, для построения графа $\bar{G}$ достаточно выполнить $O(r^2 \log r)$ элементарных действий.

Отметим, что в случае, когда каждое требование обслуживается каждым прибором не более одного раза (т.е. $r_i \leq M$, $L_{j_1}^i \neq L_{j_2}^i$ при $j_1 \neq j_2$, $i \in \{1, 2\}$), оптимальное расписание может быть построено в результате выполнения не более чем $O(M^2)$ действий при разрешении прерываний и $O(M \log M)$ действий, если прерывания запрещены. Этот результат непосредственно следует из приведенных выше рассуждений с учетом того факта, что число особых прямоугольников W_{uv} в данном случае не превосходит M .

§ 2. Максимальное временное смещение

В этом параграфе рассматривается задача минимизации максимального временного смещения при обслуживании двумя приборами требований с одинаковыми длительностями всех операций. В качестве вспомогательной используется задача построения расписания, при котором все требования обслуживаются не позднее заданных директивных сроков.

2.1. В систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Маршрут L^i обслуживания требования $i \in N$ представляет собой чередующуюся последовательность приборов $(L_1^i, L_2^i, \dots, L_{r_i}^i)$, в которой $L_q^i \in \{A, B\}$, $q = 1, \dots, r_i$, и $L_q^i \neq L_{q+1}^i$, $q = 1, \dots, r_i - 1$.

Длительности всех операций предполагаются одинаковыми (не нарушая общности, их можно считать единичными). Прерывания в процессе выполнения каждой операции запрещены. Известны директивные сроки D_i обслуживания требований $i \in N$.

Необходимо построить расписание s^* , которому соответствует наименьшее значение функционала

$$L_{\max}(s) = \max\{\bar{t}_i(s) - D_i \mid i \in N\},$$

где $\bar{t}_i(s)$ — момент завершения обслуживания требования i прибором $L_{r_i}^i$ при расписании s , а $\bar{t}_i(s) - D_i$ — временное смещение в обслуживании требования i . Расписание s^* будем называть оптимальным.

При решении поставленной задачи можно ограничиться рассмотрением класса S расписаний s , при которых каждая операция начинается (и завер-

шается) в целочисленные моменты времени и в каждый момент времени t , $0 \leq t \leq \bar{t}_{\max}(s)$, выполняется хотя бы одна операция. Здесь, как и раньше, $\bar{t}_{\max}(s) = \max\{\bar{t}_i(s) | i \in N\}$. В этом параграфе операции будем обозначать через (i, q) (требование $i \in N$ на стадии q , $1 \leq q \leq r_i$, обслуживается прибором $L \in \{A, B\}$, однозначно определяемым значениями i, q и L_1^i). Символ γ будем употреблять для обозначения простоя прибора.

Присвоим временным интервалам единичной длины, начиная с момента времени $d = 0$, номера $1, 2, \dots, R$, где $R = \sum_{i=1}^n r_i$. Интервал с номером t имеет вид $(t-1, t]$.

Очевидно, что для любого расписания $s \in S$ выполняется неравенство $\bar{t}_{\max}(s) \leq R$. Для задания расписания s достаточно указать две последовательности вида $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_R)$ и $\beta = (\beta_1, \beta_2, \dots, \beta_R)$, где $\alpha_t = (i, q)$ ($\beta_t = (i, q)$), если операция (i, q) выполняется во временном интервале с номером t и $L_q^i = A$ (соответственно $L_q^i = B$). Если в указанном временном интервале прибор A (прибор B) простаивает, то $\alpha_t = \gamma$ (соответственно $\beta_t = \gamma$). Поскольку общее число операций равно R , то суммарное число символов γ в последовательностях α и β равно R .

2.2. Опишем алгоритм построения оптимального расписания.

Каждой операции (i, q) сопоставим вес

$$v(i, q) = D_i - r_i + q. \quad (2.1)$$

Содержательно $v(i, q)$ означает самый поздний срок (возможно, отрицательный) завершения выполнения операции (i, q) , при котором не нарушится директивный срок D_i обслуживания требования i , если операции $(i, q+1), (i, q+2), \dots, (i, r_i)$ будут выполняться непрерывно одна за другой (начиная с момента времени $v(i, q)$).

Сформируем список $\lambda = ((i_1, q_1), (i_2, q_2), \dots, (i_R, q_R))$ всех операций (i, q) , $q = 1, \dots, r_i$, $i \in N$, упорядочив их по неубыванию весов $v(i, q)$. Заметим, что по определению величины $v(i, q)$ порядок следования операций в списке λ не противоречит порядку следования операций при обслуживании каждого требования $i \in N$.

Построим расписание, соответствующее списку λ . На каждом шаге алгоритма для очередной операции (i, q) из списка λ будем находить временной интервал ее выполнения. При этом будем использовать следующие параметры: для каждого требования $i \in N$ параметр $t(i)$ будет обозначать номер единичного интервала, в течение которого выполняется операция $(i, q-1)$; параметр $T(L)$ будет обозначать наименьший номер единичного интервала, в течение которого прибор $L = L_q^i$ простаивает.

Первоначально полагаем $T(A) = T(B) = 1$, $\alpha_t = \beta_t = \gamma$ для всех $t = 1, \dots, R$ и $t(i) = 0$ для всех $i \in N$. На первом шаге алгоритма в списке λ выбираем первую по порядку следования операцию (i_1, q_1) . Очевидно, $q_1 = 1$. Если $L_1^{i_1} = A$, то $\alpha_1 = \gamma$ заменяем на $\alpha_1 = (i_1, 1)$ и полагаем $T(A) = 2$ и $t(i_1) = 1$. Если $L_1^{i_1} = B$, то $\beta_1 = \gamma$ заменяем на $\beta_1 = (i_1, 1)$ и полагаем $T(B) = 2$ и $t(i_1) = 1$.

Пусть после выполнения $m - 1$, $2 \leq m \leq R$, шагов построены некоторые последовательности α и β . На m -ом шаге в списке λ выбираем очередную операцию (i_m, q_m) и вычисляем значение $\bar{t}$ по формуле

$$\bar{t} = \max\{t(i_m) + 1, T(L_{q_m}^{i_m})\}. \quad (2.2)$$

Пусть для определенности $L_{q_m}^{i_m} = A$, тогда полагаем $t(i_m) = \bar{t}$ и $\alpha_{\bar{t}} = (i_m, q_m)$. Остальные компоненты последовательностей α и β оставляем без изменения. Если $\bar{t} = T(A)$, то просматриваем подпоследовательность $\alpha_{\bar{t}+1}, \alpha_{\bar{t}+2}, \dots, R$, начиная с компоненты $\alpha_{\bar{t}+1}$ до первой по порядку следования компоненты $\alpha_{t'} = \gamma$, и полагаем $T(A) = t'$.

Обозначим последовательности α и β , полученные в результате выполнения R шагов описанного алгоритма, через $\tilde{\alpha} = (\tilde{\alpha}_1, \tilde{\alpha}_2, \dots, \tilde{\alpha}_R)$ и $\tilde{\beta} = (\tilde{\beta}_1, \tilde{\beta}_2, \dots, \tilde{\beta}_R)$.

2.3. Покажем, что последовательности $\tilde{\alpha}$ и $\tilde{\beta}$ определяют некоторое расписание из множества S .

Очевидно, каждая операция (i, q) , $q = 1, \dots, r_i$, $i \in N$, встречается в указанных последовательностях не более одного раза. Покажем, что каждая операция встречается точно один раз в соответствующей последовательности $\tilde{\alpha}$ или $\tilde{\beta}$.

Для этого достаточно показать, что если на шаге m алгоритма $t(i_m) + 1 > T(L_{q_m}^{i_m})$, то при $L_{q_m}^{i_m} = A$ компонента $\alpha_{t(i_m)+1} = \gamma$, а при $L_{q_m}^{i_m} = B$ компонента $\beta_{t(i_m)+1} = \gamma$. Для определенности рассмотрим случай $L_{q_m}^{i_m} = A$ и докажем вначале следующее вспомогательное утверждение.

Если в процессе выполнения алгоритма на шаге m компонента α_t последовательности α равна γ , то из равенства $\alpha_{k+1} = (i, q)$ при $k \geq t$ и $q > 1$ следует равенство $\beta_k = (i, q - 1)$.

Для доказательства справедливости этого утверждения воспользуемся методом математической индукции относительно числа k .

Утверждение справедливо при $k = t$. Действительно, если $\alpha_{t+1} = (i, q)$ и $\beta_t \neq (i, q - 1)$, то выполнение операции (i, q) должно быть назначено раньше интервала времени с номером $t + 1$ (например, в интервале с номером t , поскольку $\alpha_t = \gamma$).

Предположим, что утверждение имеет место для всех $k < k^0$, и пусть $\alpha_{k^0+1} = (i, q)$. Выберем максимальное значение h , для которого $t \leq h \leq k^0$ и $\alpha_h = \gamma$. По индуктивному предположению β_{k-1} и α_k — последовательные операции по обслуживанию одного и того же требования для каждого k , $h + 1 \leq k \leq k^0 - 1$.

Предположим, что $\beta_{k^0} \neq (i, q - 1)$. Тогда для каждого $k \in \{h, h + 1, \dots, k^0 - 1\}$ компонента $\beta_k \neq (i, q)$, $q \in \{1, 2, \dots, r_i\}$, поскольку β_k и α_{k+1} являются операциями по обслуживанию одного и того же требования при $h \leq k \leq k^0 - 1$ (рис. 2.1).

Следовательно, выполнение операции (i, q) должно быть назначено в интервале времени с номером h или раньше. Это противоречит предполо-

жению о том, что $\beta_{k^0+1} = (i, q)$, и завершает доказательство вспомогательного утверждения.

Вернемся к доказательству равенства $\alpha_{t(i_m)+1} = \gamma$ при $L_{q_m}^{i_m} = A$.

Предположим, что $\alpha_{t(i_m)+1} \neq \gamma$. Из неравенства $t(i_m) + 1 > T(A)$ следует, что $q_m > 1$. Непосредственно по определению параметров $T(A)$ и $t(i_m)$ получаем $\alpha_{T(A)} = \gamma$ и $\beta_{t(i_m)} = (i_m, q_m - 1)$. Тогда в силу доказанного утверждения $\beta_{t(i_m)}$ и $\alpha_{t(i_m)+1}$ являются операциями по обслуживанию одного и того же требования i_m , т.е. $\alpha_{t(i_m)+1} = (i_m, q_m)$, что противоречит предположению о том, что на шаге m из списка λ выбирается операция (i_m, q_m) .

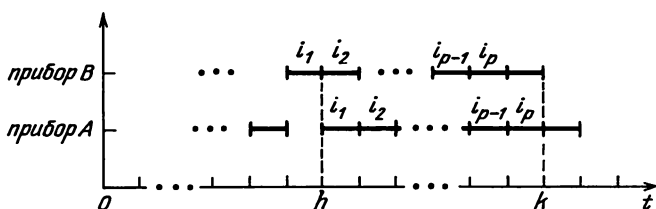


Рис. 2.1

Следовательно, каждая операция (i, q) , $q = 1, \dots, r_i$, $i \in N$, точно один раз встречается в соответствующей последовательности $\tilde{\alpha}$ или $\tilde{\beta}$.

По построению порядок следования операций в последовательностях $\tilde{\alpha}$ и $\tilde{\beta}$ не противоречит маршруту $(L_1^i, L_2^i, \dots, L_{r_i}^i)$ обслуживания требования $i \in N$. При этом если для некоторого $t \in \{1, 2, \dots, R-1\}$ значение $\tilde{\alpha}_k = \tilde{\beta}_t = \gamma$, то $\tilde{\alpha}_{k+1} = \tilde{\beta}_{t+1} = \gamma$.

Таким образом, последовательности $\tilde{\alpha}$ и $\tilde{\beta}$ определяют некоторое расписание $\tilde{s}$ из множества S . Это расписание обычно называется *списочным* или λ -расписанием.

2.4. Покажем, что расписание $\tilde{s}$ является искомым оптимальным расписанием.

Для этого достаточно доказать, что из условия существования расписания, при котором все требования обслуживаются в заданные директивные сроки, следует, что и λ -расписание $\tilde{s}$ является допустимым относительно тех же директивных сроков. Действительно, если s^* — оптимальное расписание, $l^* = L_{\max}(s^*)$ и $D'_i = D_i + l^*$, $i = 1, \dots, n$, то s^* является допустимым расписанием относительно модифицированных директивных сроков D'_i (т.е. $\bar{t}_i(s^*) \leq D'_i$, $i = 1, \dots, n$) и хотя бы при одном $i \in \{1, 2, \dots, n\}$ $\bar{t}_i(s^*) = D'_i$. Если $\bar{t}_i(\tilde{s}) \leq D'_i$, $i = 1, \dots, n$, то $L_{\max}(\tilde{s}) \leq l^*$, т.е. $\tilde{s}$ — оптимальное расписание.

Покажем, что для существования допустимого (относительно директивных сроков) расписания необходимо и достаточно, чтобы было допустимым λ -расписание, соответствующее списку $\lambda = ((i_1, q_1), (i_2, q_2), \dots, (i_R, q_R))$, где $v(i_j, q_j) \leq v(i_{j+1}, q_{j+1})$, $j = 1, \dots, R-1$.

Достаточность утверждения очевидна. Докажем необходимость.

Предположим, что при расписании $\tilde{s}$ существует хотя бы одно запаздывающее требование. Тогда из соотношения (2.1) следует, что найдется номер t , $1 \leq t \leq R$, единичного временного интервала, для которого $t > v(\tilde{\alpha}_t)$ или $t > v(\tilde{\beta}_t)$. Среди всех таких значений t выберем минимальное значение t^0 . Для определенности пусть $t^0 > v(\tilde{\alpha}_{t^0})$.

Докажем справедливость следующих соотношений:

$$\begin{aligned} \tilde{\alpha}_t \neq \gamma, \quad v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_{t^0}) \quad \text{при } t = 2, \dots, t^0 - 1, \\ v(\tilde{\alpha}_1) \leq v(\tilde{\alpha}_{t^0}) \quad \text{при } \tilde{\alpha}_1 \neq \gamma. \end{aligned} \quad (2.3)$$

Если $\tilde{\alpha}_{t^0} = (i, 1)$, то $\tilde{\alpha}_t \neq \gamma$ при $1 \leq t \leq t^0 - 1$ и из неравенства вида $v(\tilde{\alpha}_t) > v(\tilde{\alpha}_{t^0})$ следует, что при расписании $\tilde{s}$ операция $\tilde{\alpha}_{t^0}$ выполняется раньше операции $\tilde{\alpha}_t$, т.е. $t^0 < t$.

Пусть $\tilde{\alpha}_{t^0} = (i, q)$ и $q > 1$. Тогда $(i, q - 1) = \tilde{\beta}_h$, где $h < t^0 - 1$. Из определения $v(i, q)$ и равенства $h = t^0 - 1$ следовало бы соотношение $v(\tilde{\beta}_h) = v(\tilde{\alpha}_{t^0}) - 1 < t^0 - 1 = h$, что противоречит минимальности выбранного значения t^0 .

Отсюда следует, что $\tilde{\alpha}_t \neq \gamma$ и $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_{t^0})$ при $h + 1 \leq t \leq t^0 - 1$, так как в противном случае операция $\tilde{\alpha}_{t^0}$ выполнялась бы во временном интервале, номер которого меньше t^0 .

Кроме того, должно выполняться и неравенство $v(\tilde{\alpha}_h) \leq v(\tilde{\alpha}_{h+1})$. В противном случае операция $\tilde{\alpha}_{h+1}$ выполнялась бы во временном интервале, номер которого меньше $h + 1$, поскольку $\tilde{\alpha}_{h+1}$ и $\tilde{\beta}_h$ — операции по обслуживанию разных требований.

Таким образом, при $h = 1$ соотношения (2.3) выполняются.

Пусть $h > 1$ и $\tilde{\alpha}_h = (i', q')$. Если $q' = 1$, то, как уже было показано выше, $\tilde{\alpha}_t \neq \gamma$ и $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_h)$ при $t = 1, \dots, h - 1$, и соотношения (2.3) имеют место.

Пусть $q' > 1$. Если $(i', q' - 1) = \tilde{\beta}_p$, где $p < h - 1$, то, повторяя приведенные выше рассуждения, получаем $\tilde{\alpha}_t \neq \gamma$ и $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_h)$ при $t = p, \dots, h - 1$. Если $(i', q' - 1) = \tilde{\beta}_{h-1}$, то найдем наименьший номер p' такого временного интервала, что $\tilde{\beta}_t$ и $\tilde{\alpha}_{t+1}$ — операции по обслуживанию требования i' при $t = p', \dots, h - 1$. Тогда $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_h)$ при $t = p', \dots, h - 1$. Поскольку $\tilde{\alpha}_{h+1}$ является операцией по обслуживанию требования, отличного от i' , то $v(\tilde{\alpha}_h) \leq v(\tilde{\alpha}_{h+1})$. Следовательно, при $q' > 1$ выполняются неравенства $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_{h+1})$ при $t = p', \dots, h$.

Если $p' = 1$, то соотношения (2.3) имеют место. В противном случае описанные рассуждения можно повторить.

Поскольку число R временных интервалов конечно, то за конечное число шагов приходим к заключению о справедливости соотношений (2.3).

Покажем, что из этих соотношений следует существование хотя бы одного запаздывающего требования при любом расписании $s \in S$.

Пусть $\tilde{\alpha}_1 = \gamma$, тогда неравенства $v(\tilde{\alpha}_t) \leq v(\tilde{\alpha}_{t^0}) < t^0$ справедливы при $t = 1, \dots, t^0$. Следовательно, для того чтобы при расписании s не было запаздывающих требований, необходимо, чтобы при этом расписании операции $\tilde{\alpha}_1, \tilde{\alpha}_2, \dots, \tilde{\alpha}_{t^0}$ (единичной длительности каждая) выполнялись прибором A в течение интервала времени $(0, t^0 - 1]$, что невозможно.

Пусть $\tilde{\alpha}_1 = \gamma$, тогда все операции вида $(i, 1)$, $i \in N$, должны выполняться прибором B . Поскольку $v(\tilde{\alpha}_t) < t^0$ при $t = 2, \dots, t^0$, то для того чтобы не было запаздывающих требований, необходимо операции $\tilde{\alpha}_2, \tilde{\alpha}_3, \dots, \tilde{\alpha}_{t^0}$ выполнить прибором A в течение интервала времени $(1, t^0 - 1]$, что также невозможно.

Итак, из предположения о том, что при расписании $\tilde{s}$ существует хотя бы одно запаздывающее требование, следует, что и любое другое расписание $s \in S$ не может быть допустимым относительно заданных директивных сроков. Утверждение доказано.

Таким образом, λ -расписанию $\tilde{s}$, построенному согласно алгоритму из п. 2.2, соответствует наименьшее значение $L_{\max}(s)$, $s \in S$. В частности, если существует допустимое относительно директивных сроков расписание, то $\tilde{s}$ является одним из таких расписаний.

2.5. Оценим *временную сложность* описанного в п. 2.2 алгоритма построения оптимального расписания.

Трудоемкость построения последовательностей $\tilde{\alpha}$ и $\tilde{\beta}$ по известному списку λ определяется, по существу, трудоемкостью нахождения значений t' . Поскольку при поиске этих значений на всех шагах алгоритма ни одна из компонент последовательностей α и β не просматривается более одного раза, то последовательности $\tilde{\alpha}$ и $\tilde{\beta}$ строятся за $O(R)$ элементарных действий.

Список λ можно сформировать за $O(R \log R)$ элементарных действий или, учитывая, что $v(i, q + 1) = v(i, q) + 1$, $q = 1, \dots, r_i - 1$, $i \in N$, — за $O(R \log n)$ действий.

Если все D_i , $i = 1, \dots, n$, — целые числа, то список λ можно сформировать за $O(R)$ действий.

Действительно, обозначим через l нижнюю оценку значения $L_{\max}(s^*)$. В качестве l может быть выбрано, например, значение $\max\{r_i - D_i \mid i \in N\}$. Через N' обозначим множество всех требований $i \in N$, для которых $R - D_i \leq l$.

Для всех $i \in N \setminus N'$ и $1 \leq q \leq r_i$ имеет место соотношение $-l \leq v(i, q) \leq 2R - l$. Поскольку $v(i, q)$ — целые числа, то для упорядочения по неубыванию всех чисел $v(i, q)$, $i \in N \setminus N'$, $1 \leq q \leq r_i$, достаточно $O(R)$ действий.

Нетрудно видеть, что при любом расписании $s \in S$ для всех $i \in N'$ выполняются соотношения $\tilde{t}_i(s) - D_i \leq R - D_i \leq l \leq l^*$. Поэтому для каждого $i \in N'$ и $1 \leq q \leq r_i$ можно положить $v(i, q) = W + q$, где W — достаточно большое число, например, $W > 2R - l$, и упорядочить по неубыванию все числа $v(i, q)$, $1 \leq q \leq r_i$, $i \in N'$, за $O(R)$ действий.

Т а б л и ц а 2.1

i	L^i	r_i	D_i
1	(A, B, A, B, A, B, A, B)	8	10
2	$(A, B, A, B, A, B, A, B, A, B)$	10	18
3	(A, B, A, B, A, B)	6	10
4	(B, A, B, A, B)	5	20
5	$(B, A, B, A, B, A, B, A, B)$	9	20

Таким образом, для построения оптимального расписания достаточно выполнить $O(R \log n)$ действий, а при условии целочисленности директивных сроков — $O(R)$ действий.

2.6. П р и м е р. Группа "летающих" лыжников N проводит тренировку по прыжкам с трамплина. В распоряжении спортсменов имеется подъемник на одного человека (прибор A) и трамплин (прибор B). Время подъема на трамплин равно единице и равно времени, отведенному на прыжок с трамплина. Для каждого лыжника $i \in N$ задан директивный срок D_i , к которому желательно закончить тренировку.

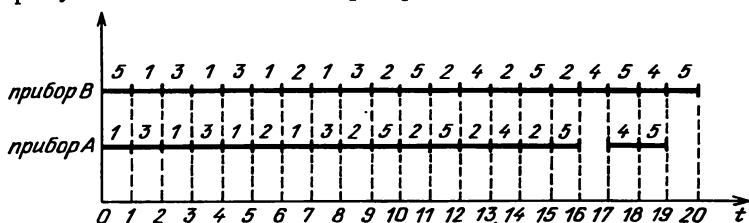


Рис. 2.2

В задаче требуется составить такое расписание s^* тренировки спортсменов, при котором принимает наименьшее значение величина $L_{\max}(s) = \max\{\bar{t}_i(s) - D_i \mid i \in N\}$ ($\bar{t}_i(s) - D_i$ — временное смещение момента $\bar{t}_i(s)$ завершения тренировки спортсмена $i \in N$ относительно заданного директивного срока D_i). Исходные данные задачи приведены в табл. 2.1.

По формуле (2.1) вычисляем значения весов $v(i, q)$, $1 \leq q \leq r_i$, $i = 1, 2, 3, 4, 5$, и упорядочиваем все операции по неубыванию полученных значений: $\lambda = ((1,1), (1,2), (1,3), (3,1), (1,4), (3,2), (1,5), (3,3), (1,6), (3,4), (1,7), (2,1), (3,5), (1,8), (2,2), (3,6), (2,3), (2,4), (5,1), (2,5), (5,2), (2,6), (5,3), (2,7), (5,4), (2,8), (4,1), (5,5), (2,9), (4,2), (5,6), (2,10), (4,3), (5,7), (4,4), (5,8), (4,5), (5,9))$.

Применяя описанный в п. 2.2 алгоритм, получаем λ -расписание $\tilde{s}$, представленное на рис. 2.2.

λ -расписание является оптимальным. Поскольку значение целевого функционала $L_{\max}(\tilde{s}) = \max\{8-10, 16-18, 9-10, 19-20, 20-20\} = 0$, то расписание $\tilde{s}$ является допустимым относительно заданных директивных сроков.

§ 3. Общее время обслуживания. Одинаковые длительности

В данном параграфе рассматривается задача построения оптимальных по быстродействию расписаний (без прерываний) обслуживания требований с одинаковыми длительностями всех операций.

В системе, состоящей из M приборов, обслуживается множество требований $N = \{1, 2, \dots, n\}$. Для каждого требования $i \in N$ задан момент времени $d_i \geq 0$ его поступления в систему и маршрут обслуживания $L^i = (L_1^i, L_2^i, \dots, L_{r_i}^i)$, где $L_q^i \in \mathcal{M} = \{1, 2, \dots, M\}$, $q = 1, \dots, r_i$, и $L_q^i \neq L_{q+1}^i$, $q = 1, \dots, r_i - 1$. Для каждой операции (i, L, q) по обслу-

живанию требования $i \in N$ прибором $L = L_q^i$ на стадии q известна длительность ее выполнения $t_{iLq} > 0$. Почти всюду в этом параграфе длительности выполнения операций предполагаются одинаковыми (без потери общности в таких случаях их можно считать единичными). Прерывания в процессе выполнения каждой операции не допускаются. На множестве требований может быть задано отношение $\rightarrow$ строгого порядка: если $i \rightarrow j$ ($i, j \in N$), то должно выполняться неравенство $t_j^0(s) \geq \bar{t}_i(s)$, где $t_j^0(s)$ (соответственно $\bar{t}_i(s)$) — момент начала (завершения) обслуживания требования j (требования i) при расписании s .

Необходимо построить расписание s^* , при котором общее время обслуживания требований $\bar{t}_{\max}(s) = \max\{\bar{t}_i(s) | i \in N\}$ является минимальным. Расписание s^* принято называть *оптимальным по быстродействию*.

3.1. Покажем, что построение оптимального по быстродействию расписания при произвольных значениях $d_i \geq 0$ сводится к построению расписания с наименьшим значением максимального временного смещения при $d_i = 0$, $i = 1, \dots, n$.

Рассмотрим две задачи построения расписания обслуживания требований $N = \{1, 2, \dots, n\}$ с наименьшим значением величины $L_{\max}(s) = \max\{\bar{t}_i(s) - D_i | i \in N\}$.

В первой из этих задач все требования поступают в систему одновременно ($d_i' = 0$, $i = 1, \dots, n$), известен маршрут $L^i = (L_1^i, L_2^i, \dots, L_{r_i}^i)$ обслуживания каждого требования $i \in N$ и директивный срок D_i' . На множестве N может быть задано отношение строгого порядка $\rightarrow$.

Во второй задаче требование $i \in N$ поступает в систему в момент времени $d_i'' = D - D_i'$, маршрут $(L_{r_i}^i, L_{r_i-1}^i, \dots, L_1^i)$ обслуживания требования $i \in N$ является обратным маршруту L^i и директивный срок $D_i'' = D$, где $D = \max\{D_i' | i \in N\}$. На множестве N задано отношение строгого порядка $\Rightarrow$, *обратного порядку* $\rightarrow$ (т.е. $i \Rightarrow j$ тогда и только тогда, когда $j \rightarrow i$).

Длительности t_{iLq}^i операций (i, L, q) первой задачи и длительности t_{iLq}'' операций (i, L, q) второй задачи связаны соотношениями

$$t_{iLq}^i = t_{iL_{r_i-q+1}}'', \quad q = 1, \dots, r_i, \quad i = 1, \dots, n.$$

Эти задачи будем называть *сопряженными*.

Пусть s' — расписание для первой задачи, при котором максимальное временное смещение принимает наименьшее значение $L_{\max}(s') = l'$. Обозначим через $\bar{s}$ совокупность M кусочно-постоянных функций $\{\bar{s}_1(t), \bar{s}_2(t), \dots, \bar{s}_M(t)\}$ таких, что $s_L(t) = s'_L(t)$, $L \in \mathcal{M}$, если t не является точкой разрыва функции $s'_L(t)$. В точках t разрыва функции $s'_L(t)$ положим $\bar{s}_L(t) = \bar{s}_L(t + \epsilon)$ для достаточно малых $\epsilon > 0$. Таким образом, в отличие от $s'_L(t)$ функция $\bar{s}_L(t)$ непрерывна справа, а не слева. Для каждого $L \in \mathcal{M}$, полагая $s_L''(t) = \bar{s}_L(D + L' - t)$ при $t \in (0, D + L']$ и $s_L''(t) = 0$ при $t > D + L'$, получим допустимое расписание s'' для второй задачи.

Нетрудно видеть, что $L_{\max}(s'') = \max\{\bar{t}_i(s'') - D_i'' | i \in N\} = l'$ и s'' — оптимальное расписание для второй задачи. Поскольку $D_i'' = D$, $i \in N$, то расписание s'' является оптимальным и по быстродействию, причем $\bar{t}_{\max}(s'') = l' + D$.

Следовательно, для решения задачи построения оптимального по быстродействию расписания при произвольных значениях d_i достаточно решить сопряженную ей задачу построения расписания с наименьшим значением $L_{\max}(s)$ при $d_i = 0$, $D'_i = D - d_i$, $D = \max\{d_i | i \in N\}$, $i = 1, \dots, n$.

Поскольку приведенные рассуждения обратимы, то имеет место и обратное сведение задачи построения расписания с наименьшим значением максимального временного смещения при $d_i = 0$, $i \in N$, к задаче построения оптимального по быстродействию расписания при произвольных значениях $d_i \geq 0$.

3.2. Рассмотрим задачу построения оптимального по быстродействию расписания обслуживания n требований двумя приборами A и B . Требование $i \in N$ поступает в момент времени $d_i \geq 0$. Длительности выполнения всех операций равны единице.

Как показано в предыдущем параграфе, сопряженная данной задаче построения расписания с наименьшим значением $L_{\max}(s)$ при $d_i = 0$, $D'_i = D - d_i$, $D = \max\{d_i | i \in N\}$, $i = 1, \dots, n$, относится к классу полиномиально разрешимых.

Таким образом, задача построения оптимального по быстродействию расписания при $M = 2$ и $t_{iLq} = 1$, $q = 1, \dots, r_i$, $i = 1, \dots, n$, также полиномиально разрешима, и для ее решения можно воспользоваться алгоритмом, описанным в п. 2.2. Напомним, что трудоемкость этого алгоритма не превосходит $O(R \log n)$, а при условии целочисленности значений

d_i , $i \in N$, — величину $O(R)$, где $R = \sum_{i=1}^n r_i$.

3.3. П р и м е р. Группа парашютистов проводит тренировку по прыжкам с вышки. В их распоряжении имеется лифт на одного человека (прибор A) и парашют для прыжков с вышки (прибор B). Время подъема на вышку и время прыжка с вышки равны между собой. Для каждого парашютиста известны момент времени d_i готовности к началу тренировки и число $r_i/2$ прыжков, которые нужно совершить.

Требуется построить оптимальное по быстродействию расписание тренировки парашютистов при исходных данных, приведенных в табл. 3.1.

Исходные данные для сопряженной задачи приведены в табл. 3.2

В соответствии с алгоритмом, описанным в п. 2.2, сформулируем список λ операций (i, q) , $q = 1, \dots, r_i$, $i = 1, 2, 3, 4$, упорядочив их по неубыванию весов $v(i, q) = D_i - r_i + q$: $\lambda = ((1,1), (4,1), (1,2), (4,2), (1,3), (4,3))$.

Т а б л и ц а 3.1

i	L^i	r_i	d_i
1	(A, B, A, B, A, B)	6	7
2	(A, B, A, B)	4	0
3	(A, B)	2	3
4	(A, B, A, B, A, B, A, B)	8	5

Таблица 3.2

i	L_i	r_i	d_i
1	(B, A, B, A, B, A)	6	0
2	(B, A, B, A)	4	7
3	(B, A)	2	4
4	(B, A, B, A, B, A, B, A)	8	2

(1,4), (4,4), (1,5), (4,5), (1,6), (4,6), (4,7), (4,8), (3,1), (2,1), (3,2), (2,2), (2,3), (2,4)). λ -расписание $\tilde{s}$, представленное на рис. 3.1, является оптимальным расписанием для сопряженной задачи и $L_{\max}(\tilde{s}) = 7$.

Расписание $\tilde{s}$ однозначно определяет оптимальное по быстродействию расписание s^* для исходной задачи, которое представлено на рис. 3.2. Значение $\bar{t}_{\max}(s^*) = L_{\max}(\tilde{s}) + \max\{D_i \mid i = 1, 2, 3, 4\} = 7 + 7 = 14$.

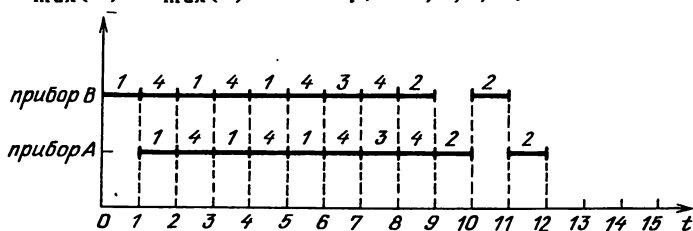


Рис. 3.1

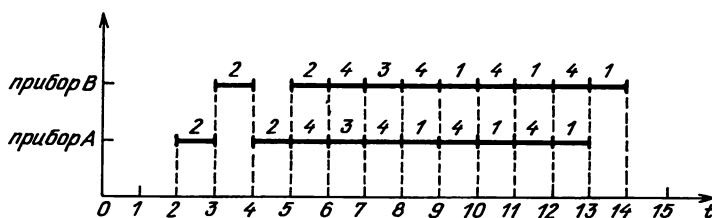


Рис. 3.2

3.4. Покажем, что при $M = 3$, $d_i = 0$, $i \in N$, и единичных длительностях всех операций задача построения оптимального по быстродействию расписания является NP-трудной в сильном смысле.

Сформулируем соответствующую задачу распознавания.

В обслуживающую систему, состоящую из трех приборов А, В и С, в момент времени $d = 0$ поступает множество требований $N = \{1, 2, \dots, n\}$. Для каждого требования $i \in N$ задан маршрут обслуживания L^i . Длительность выполнения каждой операции равна единице. Необходимо определить, существует ли расписание s^0 (без прерываний операций) такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного значения y .

Построим псевдополиномиальное сведение задачи о 3-разбиении (см. вводную часть данной главы) к сформулированной задаче распознавания.

Положим $n = 3n_0 + 2$, $y = n_0(2E + 9)$, $L^i = (A, C, [A, B]^{e_i}, C)$, $i = 1, \dots, 3n_0$, $L^{n-1} = ([B, C, B, A, B, A, [B, C]^E, A, B, A]^{n_0})$, $L^n =$

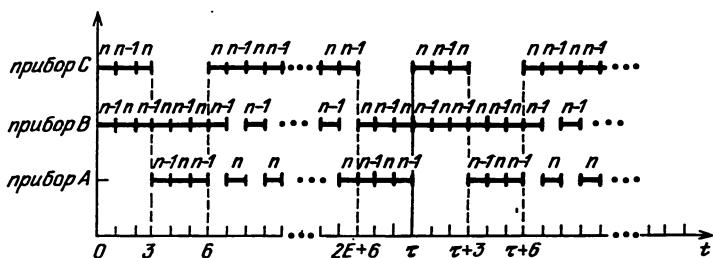


Рис. 3.3

$= ([C, B, C, B, A, B, [C, A]^E, B, A, B])^{n_0}$). Здесь и далее $[\cdot]^h$ — последовательность, состоящая из h повторений стоящего в скобках выражения. Например, $[A, B]^2 = A, B, A, B$.

Заметим, что число операций по обслуживанию как требования $n - 1$, так и требования n равно y . Следовательно, можно считать, что при расписании s^0 операция $(n - 1, L_q^{n-1}, q)$ и операция (n, L_q^n, q) выполняются во временном интервале $(q - 1, q]$, т.е. $t_{n-1, L_q^{n-1}}^0 = t_{n, L_q^n}^0 = q - 1$.

На рис. 3.3 представлен фрагмент расписания обслуживания требований $n - 1$ и n . Отметим, что это расписание является периодическим, т.е. график расписания состоит из n_0 одинаковых частей. На рис. 3.3 первая такая часть отделена сплошной вертикальной чертой, а период расписания обозначен через τ и равен $2E + 9$.

Как нетрудно заметить, для обслуживания требований $i = 1, \dots, 3n_0$ остается последовательность непрерывно следующих один за другим временных интервалов, начинающаяся в момент $t = 0$ и заканчивающаяся в момент $t = y$. Используя принятые обозначения, эту последовательность можно записать в виде $([A]^3, [C]^3, [A, B]^E, [C]^3)^{n_0}$.

Покажем, что требования $i = 1, \dots, 3n_0$ могут быть обслужены в соответствии с указанной последовательностью (и, следовательно, существует расписание, при котором общее время обслуживания требований равно y) тогда и только тогда, когда задача о 3-разбиении имеет решение.

Предположим, что задача о 3-разбиении имеет решение, т.е. существует разбиение множества N^0 на n_0 подмножеств $N_1^0, N_2^0, \dots, N_{n_0}^0$ таких, что $\sum_{i \in N_j^0} e_i = E$ при $j = 1, \dots, n_0$. Тогда, обслуживая каждое требование $i \in N_j^0$ от начала до конца в течение интервала времени $(\tau(j - 1), \tau j]$, $j = 1, \dots, n_0$, получаем расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$.

Предположим, что существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$, и докажем по индукции относительно j , что при расписании s^0 во временном интервале $(\tau(j - 1), \tau j]$, $1 \leq j \leq n_0$, обслуживаются (причем полностью) ровно три требования из множества $\{1, 2, \dots, 3n_0\}$.

Очевидно, что такие требования будут представлять собой трехэлементное подмножество N_j^0 , для которого $\sum_{i \in N_j^0} e_i = E$.

Рассмотрим случай $j = 1$. Если при расписании s^0 обслуживание некоторого требования i , $1 \leq i \leq 3n_0$, начинается во временном интервале $(0, \tau]$, то последовательность операций по его обслуживанию, выполнение которых заканчивается в этом же интервале, можно отнести к одному из следующих четырех типов:

- 1) (A) ;
- 2) $(A, C, [A, B]^h)$, $0 \leq h \leq e_i$;
- 3) $(A, C, [A, B]^h, A)$, $0 \leq h < e_i$;
- 4) $(A, C, [A, B]^{e_i}, C)$.

Через n_k обозначим число требований i , $1 \leq i \leq 3n_0$, обслуживание которых начинается в интервале $(0, \tau]$ и указанные последовательности операций относятся к типу k , $k = 1, \dots, n$. Через m_k обозначим суммарное число операций по обслуживанию этих требований прибором B во временном интервале $(0, \tau]$.

Нетрудно видеть, что суммарное число операций (единичной длительности), которые необходимо выполнить каждому из приборов, равно u . Следовательно, при любом расписании с общим временем обслуживания требований, равным u , каждый из приборов A , B и C во временном интервале $(0, n_0\tau]$ не простаивает. Соответственно получаем следующие уравнения относительно n_k и m_k :

$$n_1 + n_2 + m_2 + 2n_3 + m_3 + n_4 + m_4 = E + 3, \quad (3.1)$$

$$m_2 + m_3 + m_4 = E, \quad (3.2)$$

$$n_2 + n_3 + 2n_4 = 6. \quad (3.3)$$

Вычитая уравнение (3.1) из суммы уравнений (3.2) и (3.3), получаем $n_1 - n_3 + n_4 = 3$. Отсюда следует, что $n_4 \geq 3$, а из уравнения (3.3) получаем $n_4 \leq 3$. Следовательно, $n_4 = 3$. Из уравнения (3.3) получаем $n_2 = n_3 = 0$. Тогда по определению $m_2 = m_3 = 0$. Из уравнения (3.2) получаем $m_4 = E$, а из уравнения (3.1) — $n_1 = 0$.

Таким образом, при расписании s^0 в интервале $(0, \tau]$ обслуживаются (причем от начала до конца) точно три требования из множества $\{1, 2, \dots, 3n_0\}$. Эти требования, очевидно, составляют искомое трехэлементное подмножество N_1^0 , для которого $\sum_{i \in N_1^0} e_i = E$.

Пусть при расписании s^0 в каждом интервале $(\tau(j-1), \tau j]$ для всех $j < j_0 \leq n$ начинается и заканчивается обслуживание требований множества $N_j^0 \subseteq N^0$, $|N_j^0| = 3$. Тогда, повторяя рассуждения, аналогичные приведенным для интервала времени $(0, \tau]$, можно показать, что при любом расписании с общим временем обслуживания требований, равным u , в интервале $(\tau(j-1), \tau j_0]$ должно начинаться и заканчиваться обслуживание точно трех ранее не обслуженных требований из множества N^0 .

Таким образом, расписание s^0 , при котором $\bar{t}_{\max}(s^0) = u$, однозначно определяет требуемое разбиение множества N^0 на трехэлементные подмножества.

Нетрудно видеть, что для реализации описанного сведения достаточно выполнить $O(n_0 E)$ элементарных действий. Поскольку задача о 3-разбие-

нии NP -полна в сильном смысле, то задача построения оптимального по быстродействию расписания обслуживания тремя приборами n одновременно поступающих требований с единичными длительностями всех операций является NP -трудной в сильном смысле.

3.5. Покажем, что если на множестве требований N задано отношение строгого порядка, то задача построения оптимального по быстродействию расписания является NP -трудной в сильном смысле при $M = 2$ и $d_i = 0$, $i = 1, \dots, n$. Этот результат справедлив даже в том случае, когда каждая компонента связности графа редукции заданного отношения строгого порядка является цепью.

Сформулируем соответствующую задачу распознавания.

В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество требований $N = \{1, 2, \dots, n\}$. На множестве N задано отношение $\rightarrow$ строгого порядка. Каждая компонента связности графа редукции отношения $\rightarrow$ является цепью. Длительность выполнения каждой операции равна единице. Требуется определить, существует ли расписание s^0 обслуживания требований множества N такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного значения y .

Построим псевдополиномиальное сведение задачи о 3-разбиении к сформулированной задаче распознавания.

Положим $n = 4n_0E$, $y = 2n_0E$, $N = \sum_{k=1}^{3n_0+1} N_k$, где $N_1 = \{1, 2, \dots, 2e_1\}$, $N_k = \{2 \sum_{l=1}^{k-1} e_l + 1, 2 \sum_{l=1}^{k-1} e_l + 2, \dots, 2 \sum_{l=1}^k e_l\}$, $k = 2, \dots, 3n_0$, и $N_{3n_0+1} = \{2n_0E + 1, 2n_0E + 2, \dots, 4n_0E\}$.

На множестве требований N зададим отношение $\rightarrow$ строгого порядка, полагая $i \rightarrow j$, тогда и только тогда, когда $i \in N_k$, $j \in N_k$, $1 \leq k \leq 3n_0 + 1$, и $i < j$. Каждая компонента связности графа редукции этого отношения, очевидно, является цепью.

Пусть процесс обслуживания каждого требования $i \in N$ состоит в выполнении единственной операции. Для требований из множества N_1 положим $L^l = (B)$ при $l = 1, \dots, e_1$ и $L^l = (A)$ при $l = e_1 + 1, \dots, 2e_1$. Для каждого k , $2 \leq k \leq 3n_0$, положим

$$\begin{aligned} \bigcup_{l=1}^{k-1} e_l + 1 &= (B) \text{ при } l = 1, \dots, e_k, \\ \bigcup_{l=1}^{k-1} e_l + 2 &= (A) \text{ при } l = e_k + 1, \dots, 2e_k. \end{aligned}$$

Для требований из множества N_{3n_0+1} положим $L^{2E(n_0+l)+1} = L^{2E(n_0+l)+2} = \dots = L^{2E(n_0+l)+E} = (A)$, $L^{2E(n_0+l)+E+1} = L^{2E(n_0+l)+E+2} = \dots = L^{2E(n_0+l+1)} = (B)$, $l = 0, \dots, n_0 - 1$.

Поскольку множество N_{3n_0+1} является линейно упорядоченным и число его элементов равно $2n_0E = y$, то при любом расписании с общим временем обслуживания требований, равным y , требования множества N_{3n_0+1} должны обслуживаться непрерывно одно за другим начиная с момента времени $t = 0$.

На рис. 3.4 представлен фрагмент расписания обслуживания требований множества N_{3n_0+1} , при котором общее время обслуживания требований равно y . Это расписание является *периодическим* с периодом $\tau = 2E$.

Таким образом, обслуживание требований множества $N \setminus N_{3n_0+1}$ при любом расписании с общим временем обслуживания требований, равным y , может быть осуществлено только в тех единичных интервалах, которые определяются последовательностью $([B]^E, [A]^E)_{n_0}$.

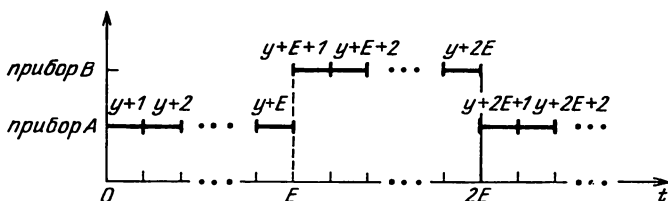


Рис. 3.4

Покажем, что расписание s^0 существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Пусть существует разбиение множества N^0 на n_0 трехэлементных подмножеств $N_1^0, N_2^0, \dots, N_{n_0}^0$ таких, что $\sum_{i \in N_j^0} e_i = E$ при $j = 1, \dots, n_0$. Тогда,

обслуживая множество требований $\bigcup_{k \in N_j^0} N_k$ в интервале $(\tau(j-1), \tau j]$,

$j = 1, \dots, n_0$, в соответствии с отношением $\rightarrow$, получаем расписание с общим временем обслуживания требований, равным y .

Пусть существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$. Поскольку суммарная длительность обслуживания всех требований прибором А (прибором В) равна y , то при расписании s^0 как прибор А, так и прибор В в интервале $(0, n_0 \tau]$ не простаивают.

По условию обслуживание каждого требования $i \in N_k, k \in N^0 = \{1, 2, \dots, 3n_0\}$, для которого $L^i = (B)$, должно предшествовать обслуживанию каждого требования $j \in N_k$, для которого $L^j = (A)$. Во множестве $N_k^A = \{i \mid i \in N_k, L^i = (A)\}$ содержится столько же элементов, сколько и во множестве $N_k^B = \{i \mid i \in N_k, L^i = (B)\}$. Для того чтобы прибор А не простаивал в интервале $(E, 2E]$, необходимо и достаточно, чтобы множество всех требований, которые обслуживаются прибором А в указанном интервале, можно было представить в виде $\bigcup_{l=1}^m N_{k_l}^A$, где $k_l \in N^0$,

$l = 1, \dots, m$, причем в интервале $(0, E]$ прибором В обслуживались бы все требования множества $\bigcup_{l=1}^m N_{k_l}^B$ и только они. При этом из неравенств

$E/4 < e_k < E/2, k = 1, \dots, 3n_0$, следует, что $m = 3$.

Таким образом, получаем подмножество $N_1^0 = \{k_1, k_2, k_3\}$ множества N^0 , для которого $\sum_{i \in N_1^0} e_i = \sum_{i \in N_1^0} |N_i^A| = E$.

Повторяя аналогичные рассуждения, можно показать, что при расписании s^0 в каждом интервале $(\tau(h-1), \tau h]$, $h = 2, \dots, n^0$, обслуживаются все требования трех множеств вида N_k , $k \in N^0$, и только они. В результате получаем разбиение множества N^0 на трехэлементные подмножества $N_1^0, N_2^0, \dots, N_{n^0}^0$ такие, что $\sum_{i \in N_j^0} e_i = E$, $1 \leq j \leq n^0$.

Для реализации этого сведения задачи о 3-разбиении к сформулированной задаче распознавания достаточно выполнить $O(n^0 E)$ элементарных действий.

Таким образом, задача построения оптимального по быстродействию расписания обслуживания двумя приборами частично упорядоченного множества одновременно поступающих в систему требований с операциями одинаковой длительности является NP -трудной в сильном смысле.

3.6. Если длительности выполнения операций принимают значения из множества $\{1, 2\}$ то задача построения оптимального по быстродействию расписания обслуживания n одновременно поступающих в систему требований является NP -трудной в сильном смысле при $M = 2$.

Сформулируем соответствующую задачу распознавания.

В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество требований $N = \{1, 2, \dots, n\}$. Предполагается, что длительности выполнения операций $t_{iLq} \in \{1, 2\}$. Необходимо определить, существует ли расписание s^0 обслуживания требований множества N такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного значения y .

Построим псевдополиномиальное сведение задачи о 3-разбиении к сформулированной задаче распознавания.

Положим $n = 3n_0 + 1$, $y = 2n_0(2E + 9)$, $L^i = (2A, B, [2A, 2B]^E, A, 2B)$, $i = 1, \dots, 3n_0$, $L^n = ([2B, A, 2B, A, 2B, A, [2B, 2A]^E, B, 2A, B, 2A, B, 2A]^n)$. Здесь используются следующие обозначения: если $L_q^i = 2A$ или $L_q^i = 2B$, то длительность t_{iLq} выполнения операции (i, L_q^i, q) равна двум; если $L_q^i = A$ или $L_q^i = B$, то $t_{iLq} = 1$.

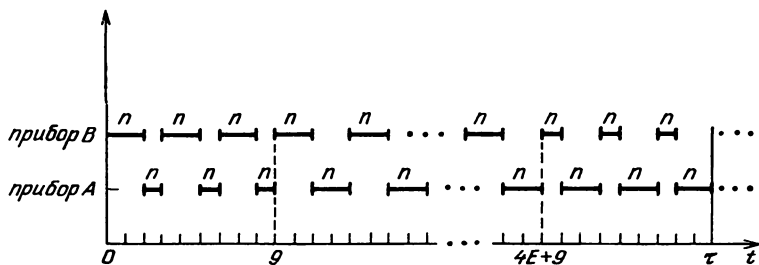


Рис. 3.5

Поскольку число операций по обслуживанию требования n равно y , то при любом расписании с общим временем обслуживания требований, равным y , эти операции должны выполняться непрерывно одна за другой, начиная с момента $t = 0$.

На рис. 3.5 представлен график обслуживания требований во временном интервале $(0, \tau]$, где $\tau = 2(2E + 9)$ — период расписания.

Таким образом, обслуживание требований множества $N^0 = \{1, 2, \dots, 3n_0\}$ при любом расписании с общим временем обслуживания требований, равным y , может быть осуществлено только в тех временных интервалах, которые определяются последовательностью $([2A, B, 2A, B, 2A, B, [2A, 2B]^E, A, 2B, A, 2B, A, 2B]^{n_0})$.

Покажем, что расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Предположим, что существует разбиение множества N^0 на n_0 трехэлементных подмножеств $N_1^0, N_2^0, \dots, N_{n_0}^0$ таких, что $\sum_{i \in N_j^0} e_i = E$ при

$j = 1, \dots, n_0$. Тогда, обслуживая полностью каждое требование $i \in N_j^0$ в интервале $(\tau(j-1), \tau j]$, $j = 1, \dots, n_0$, получаем расписание с общим временем обслуживания требований, равным y .

Предположим, что существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$. Покажем, что при этом расписании точно три требования из множества N^0 полностью обслуживаются в интервале $(0, \tau]$.

Поскольку суммарная длительность всех операций, выполняемых прибором A (прибором B), равна $9n_0 + 2 \sum_{i=1}^{3n_0} e_i + n_0(9 + 2E) = y$, то при любом расписании с общим временем обслуживания требований, равным y , как прибор A , так и прибор B не простаивают во временном интервале $(0, n_0\tau]$.

Для того чтобы прибор B в интервале $(0, 9]$ не простаивал, необходимо и достаточно, чтобы в этом интервале начиналось обслуживание точно трех требований из множества N^0 . Аналогично, для того чтобы прибор A в интервале $(4E + 9, 4E + 18]$ не простаивал, необходимо и достаточно, чтобы в этом интервале заканчивалось обслуживание точно трех требований из множества N^0 . Очевидно, что это те же требования, обслуживание которых начинается в интервале $(0, 9]$. Эти три требования и составляют подмножество $N_1^0 \subseteq N^0$, для которого $\sum_{i \in N_1^0} e_i = E$.

Далее индукцией по j можно показать, что при расписании s^0 обслуживание точно трех требований из множества $N^0 \setminus N_1^0$ начинается и завершается в интервале $(\tau(j-1), \tau j]$, $1 < j \leq n_0$. В результате получим разбиение множества N^0 на трехэлементные подмножества $N_1^0, N_2^0, \dots, N_{n_0}^0$ такие, что $\sum_{i \in N_j^0} e_i = E$, $1 \leq j \leq n_0$.

Нетрудно видеть, что для реализации описанного сведения задачи о 3-разбиении к сформулированной задаче распознавания достаточно выполнить $O(n_0 E)$ элементарных действий.

Следовательно, задача построения оптимального по быстродействию расписания обслуживания двумя приборами n одновременно поступающих требований со значениями длительностей операций из множества $\{1, 2\}$ является NP -трудной в сильном смысле.

3.7. Изложенные в п. 3.4–3.6 результаты, очевидно, справедливы и в случае неодинаковых значений d_i . В силу эквивалентности сопряженных задач (см. п.3.2) задача построения расписания, которому соответствует наименьшее значение максимального временного смещения $L_{\max}(s)$

при заданных директивных сроках D_i , является NP-трудной в сильном смысле в следующих случаях:

- а) $M = 3, d_i = 0, t_{iLq} = 1, L = L_q^i, q = 1, \dots, r_i, i = 1, \dots, n$;
- б) $M = 2, d_i = 0, t_{iLq} = 1, L = L_q^i, q = 1, \dots, r_i, i \in N = \{1, 2, \dots, n\}$,

на множестве N задано отношение $\rightarrow$ строгого порядка (достаточно, чтобы все компоненты связности графа редукции этого отношения были цепями);

- в) $M = 2, d_i = 0, t_{iLq} \in \{1, 2\}, L = L_q^i, q = 1, \dots, r_i, i = 1, \dots, n$.

§ 4. Общее время обслуживания. Различные длительности

Рассматривается задача построения оптимального по быстродействию расписания обслуживания n требований двумя приборами. В отличие от предыдущего параграфа длительности выполнения операций представляют собой произвольные действительные числа и могут допускаться прерывания в процессе выполнения операций.

4.1. В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Маршрут L^i обслуживания требования $i \in N$ представляет собой чередующуюся последовательность приборов $(L_1^i, L_2^i, \dots, L_{r_i}^i), L_q^i \in \{A, B\}, q = 1, \dots, r_i$, где $L_q^i \neq L_{q+1}^i, q = 1, \dots, r_i - 1$. Известны длительности $t_{iLq} > 0$ выполнения всех операций $(i, L, q), i \in N, L = L_q^i \in \{A, B\}, q = 1, \dots, r_i$.

Необходимо построить расписание s^* , при котором общее время обслуживания всех требований $t_{\max}(s) = \max \{t_i(s) \mid i \in N\}$ принимает наименьшее значение.

4.2. Опишем эффективный алгоритм построения оптимального по быстродействию расписания (без прерываний) обслуживания n требований двумя приборами при $r_i \leq 2, i = 1, \dots, n$.

В этом случае множество требований N можно разбить на четыре попарно непересекающихся (необязательно непустых) подмножества: $N = N_{AB} \cup N_{BA} \cup N_A \cup N_B$. Здесь $N_{AB}(N_{BA})$ — множество требований $i \in N$, обслуживаемых приборами A и B в последовательности $L^i = (A, B)$ (соответственно $L^i = (B, A)$). Если $L^i = (A)$, то $i \in N_A$. Аналогично, если $L^i = (B)$, то $i \in N_B$.

Поскольку при $r_i \leq 2$ требование $i \in N$ обслуживается каждым прибором $L \in \{A, B\}$ не более одного раза, то на протяжении этого пункта во всех обозначениях будем опускать номер стадии q , полагая $(i, L, q) = (i, L), t_{iLq} = t_{iL}, t_{iLq}^0 = t_{iL}^0, t_{iLq} = t_{iL}$.

Напомним, что в § 1 гл. 1 описан алгоритм трудоемкости $O(n \log n)$ построения оптимального по быстродействию расписания обслуживания двумя приборами n требований с одинаковыми маршрутами обслуживания.

В случае, когда $N = N_{AB}$, оптимальным по быстродействию будет расписание, при котором требования множества N обслуживаются обоими приборами в одной и той же последовательности $\pi = (i_1, i_2, \dots, i_n)$, удовлетворяющей условию

$$\min \{t_{i_1A}, t_{i_1+1B}\} \leq \min \{t_{i_1+1A}, t_{i_1B}\} \quad (4.1)$$

для всех $l = 1, \dots, n - 1$. В частности, каждому требованию $i \in N$ можно приписать вес

$$\omega(i) = [W - \min \{t_{iA}, t_{iB}\}] \operatorname{sign}(t_{iA} - t_{iB})$$

и упорядочить требования в порядке неубывания их весов. Здесь W — достаточно большое число ($W > \max_{1 \leq i \leq n} \min \{t_{iA}, t_{iB}\}$).

Аналогично при $N = N_{BA}$ наименьшее значение величины $\bar{t}_{\max}(s)$ достигается для расписания s , при котором требования множества N обслуживаются приборами A и B в одной и той же последовательности $\pi = (i_1, i_2, \dots, i_n)$, удовлетворяющей условию

$$\min \{t_{i_l B}, t_{i_{l+1} A}\} \leq \min \{t_{i_{l+1} B}, t_{i_l A}\} \quad (4.2)$$

для всех $l = 1, \dots, n - 1$.

Если $N_{AB} = \emptyset$ или $N_{BA} = \emptyset$ и хотя бы одно из множеств требований N_A или N_B непусто, то все требования множества $N_A \cup N_B$ можно отнести ко множеству N_{AB} или ко множеству N_{BA} , полагая $t_{iB} = 0$ при $i \in N_A$ и $t_{iA} = 0$ при $i \in N_B$. В результате получим множество требований с одинаковыми маршрутами обслуживания.

Таким образом, построение оптимального по быстродействию расписания обслуживания требований множества N двумя приборами не вызывает особых затруднений, если $N_{AB} = \emptyset$ или $N_{BA} = \emptyset$.

Пусть $N_{AB} \neq \emptyset$ и $N_{BA} \neq \emptyset$. Каждое расписание s для рассматриваемой задачи однозначно определяет некоторые последовательности $\pi = (i_1, i_2, \dots, i_{n_A})$ и $\pi' = (j_1, j_2, \dots, j_{n_B})$ обслуживания требований множества N прибором A и прибором B соответственно. Здесь $n_A = |N \setminus N_B|$ и $n_B = |N \setminus N_A|$.

Обозначим $N'_{AB} = N_{AB} \cup N_A$ и $N'_{BA} = N_{BA} \cup N_B$. Нетрудно видеть, что при расписании s имеют место следующие соотношения. Если $i_1 \in N'_{AB}$, то $t_{i_1 A}^0 \geq 0$, в противном случае $t_{i_1 A}^0 \geq t_{i_1 B}$. Если $j_1 \in N'_{BA}$, то $t_{j_1 B}^0 \geq 0$, в противном случае $t_{j_1 B}^0 \geq t_{j_1 A}$. Для всех $k = 2, \dots, n_A$ значение $t_{i_k A}^0 \geq \bar{t}_{i_{k-1} A}$, если $i_k \in N'_{AB}$, и $t_{i_k A}^0 \geq \max \{\bar{t}_{i_{k-1} A}, \bar{t}_{i_{k-1} B}\}$, если $i_k \in N_{BA}$.

Аналогично для всех $k = 2, \dots, n_B$ значение $t_{j_k B}^0 \geq \bar{t}_{j_{k-1} B}$, если $j_k \in N'_{BA}$, и $t_{j_k B}^0 \geq \max \{\bar{t}_{j_{k-1} B}, \bar{t}_{j_{k-1} A}\}$, если $j_k \in N_{AB}$.

Кроме того, поскольку прерывания в процессе выполнения операций не допускаются, то $\bar{t}_{i_k A} = t_{i_k A}^0 + t_{i_k A}$ для всех $k = 1, \dots, n_A$ и $\bar{t}_{j_k B} = t_{j_k B}^0 + t_{j_k B}$ для всех $k = 1, \dots, n_B$.

Очевидно, что существует бесконечное множество наборов значений $t_{i_k A}^0, k = 1, \dots, n_A$ и $t_{j_k B}^0, k = 1, \dots, n_B$, удовлетворяющих перечисленным соотношениям, и каждый из таких наборов определяет некоторое расписание s . Если при расписании s набор значений $t_{i_k A}^0$ и $t_{j_k B}^0$ удовлетворяет этим соотношениям как равенствам, то общее время обслуживания требований $\max \{\bar{t}_{i_{n_A} A}, \bar{t}_{i_{n_B} B}\}$ достигает наименьшего (при фиксированных π и π') значения, которое будем обозначать через $t(\pi, \pi')$.

Пусть в последовательности $\pi = (i_1, i_2, \dots, i_{n_A})$ требование $i_l \in N_{BA}$ и требование $i_{l+1} \in N'_{AB}$, $1 \leq l < n_A$. Покажем, что $t(\hat{\pi}, \pi') \leq t(\pi, \pi')$, где $\hat{\pi} = (i_1, i_2, \dots, i_{l-1}, i_{l+1}, i_l, i_{l+2}, \dots, i_{n_A})$.

Если требования множества N обслуживаются в последовательности $\hat{\pi}$ прибором A и в последовательности π' прибором B , то моменты начала и окончания выполнения операции (i, L) будем обозначать через τ_{iL}^0 и $\bar{\tau}_{iL}$ соответственно.

Нетрудно видеть, что $\tau_{i_{l+1}A}^0 = t_{i_{l+1}A}^0$, $\tau_{i_{l+1}A}^0 = t_{i_{l+1}A}^0 + t_{i_{l+1}A}$, а значения моментов начала выполнения остальных операций сохраняются неизменными:

$$\tau_{i_kA}^0 = t_{i_kA}^0, \quad 1 \leq k \leq n_A, k \neq l, k \neq l+1,$$

$$\tau_{i_kB}^0 = t_{i_kB}^0, \quad 1 \leq k \leq n_B.$$

Поскольку $\bar{\tau}_{i_{l+1}A} < \bar{\tau}_{i_{l+1}A} \leq t_{i_{l+1}B}^0$ и $\tau_{i_{l+1}A}^0 > t_{i_{l+1}A}^0 \geq \bar{\tau}_{i_{l+1}B}$, то значения $\tau_{i_k}^0$, $1 \leq k \leq n_A$, и значения $\tau_{j_k}^0$, $1 \leq k \leq n_B$, определяют некоторое расписание s' (без прерываний) и $\bar{t}_{\max}(s') \leq t(\pi, \pi')$.

Аналогично можно показать, что если в последовательности $\pi' = (j_1, j_2, \dots, j_{n_B})$ $j_l \in N_{AB}$ и $j_{l+1} \in N'_{BA}$, то в результате обслуживания требований множества N прибором B в последовательности $\hat{\pi}' = (j_1, j_2, \dots, j_{l-1}, j_{l+1}, j_l, j_{l+2}, \dots, j_{n_B})$ общее время обслуживания требований при соответствующем расписании может только уменьшиться по сравнению со значением $\bar{t}_{\max}(s)$, т.е. $t(\pi, \hat{\pi}') \leq t(\pi, \pi')$.

Таким образом, достаточно рассматривать только такие последовательности π и π' обслуживания требований, при которых прибор A сначала обслуживает требования множества N'_{AB} , а затем — требования множества N_{BA} , а прибор B сначала обслуживает требования множества N'_{BA} , затем — требования множества N_{AB} . При этом, поскольку $N'_{AB} \cap N'_{BA} = \emptyset$, требования множества N'_{AB} можно обслуживать прибором A непрерывно одно за другим начиная с момента $t = 0$, а требования множества N'_{BA} аналогичным образом можно обслуживать прибором B .

Покажем, что общее время обслуживания всех требований не увеличится, если потребовать, чтобы прибор A (прибор B) непрерывно обслуживал требования множества N_{BA} (множества N_{AB}) начиная с некоторого момента времени $d_A(\pi, \pi')$ (соответственно $d_B(\pi, \pi')$).

Пусть требования множества N_{BA} обслуживаются прибором A в последовательности $(i_k, i_{k+1}, \dots, i_{n_A})$, $k = |N'_{AB}| + 1$, причем $t_{i_{k+1}A}^0 = \bar{t}_{i_kA}$, $t_{i_{k+2}A}^0 = \bar{t}_{i_{k+1}A}$, $\dots$, $t_{i_{l+1}A}^0 = \bar{t}_{i_{l-1}A}$ и $t_{i_{l+1}A}^0 = \bar{t}_{i_{l+1}A} + h$, где $h > 0$ и $l > k$.

Положим $\tau_{i_pA}^0 = t_{i_pA}^0 + h$ и $\bar{\tau}_{i_pA} = \bar{t}_{i_pA} + h$ для всех $p = k, \dots, l$.

Поскольку $\tau_{i_kA}^0 = t_{i_kA}^0 + h \geq \bar{t}_{i_kB} + h > \bar{t}_{i_kB}$ и $\tau_{i_pA}^0 = t_{i_pA}^0 + h \geq \max\{\bar{t}_{i_{p-1}A}, \bar{t}_{i_{pB}}\} + h \geq \max\{\bar{\tau}_{i_{p-1}A}, \bar{t}_{i_{pB}}\}$ для всех $p = k+1, \dots, l$, то требования $i_k, i_{k+1}, \dots, i_l$ можно начинать обслуживать прибором A в моменты времени $\tau_{i_kA}^0, \tau_{i_{k+1}A}^0, \dots, \tau_{i_lA}^0$ соответственно. При этом значение $t_{i_{l+1}A}^0$ и, следовательно, общее время обслуживания требований не изменятся.

Применяя описанное преобразование конечное число раз, получим искомые значения $t_{i_pA}^0$, $p = 1, \dots, n_A$.

Очевидно, $d_A(\pi, \pi') \geq \sum_{k \in N'_{AB}} t_{kA}$, $d_B(\pi, \pi') \geq \sum_{k \in N'_{BA}} t_{kB}$ и общее время обслуживания требований $t(\pi, \pi') = \max \{d_A(\pi, \pi') + \sum_{k \in N_{AB}} t_{kA}, d_B(\pi, \pi') + \sum_{k \in N_{BA}} t_{kB}\}$.

Таким образом, решение исходной задачи сводится к построению оптимальных по быстродействию расписаний для двух однотипных задач с одинаковыми маршрутами обслуживания требований. В первой необходимо определить порядок обслуживания требований множества $N' = N'_{AB} \cup \{n+1\}$, полагая $L^{n+1} = (A, B)$, $t_{n+1,A} = 0$, $t_{n+1,B} = \sum_{k \in N'_{BA}} t_{kB}$ и $t_{iB} = 0$, если $i \in N_A$, а во второй — требований множества $N'' = N'_{BA} \cup \{n+2\}$, полагая $L^{n+2} = (B, A)$, $t_{n+2,B} = 0$, $t_{n+2,A} = \sum_{k \in N'_{AB}} t_{kA}$ и $t_{jA} = 0$, если $j \in N_B$.

В соответствии с алгоритмом, описанным в § 1 главы 1, получаем следующий результат. *Общее время обслуживания требований множества N достигает наименьшего значения для расписания, при котором прибор A обслуживает требования в последовательности N_{AB} , N_A , N_{BA} , а прибор B — в последовательности N_{BA} , N_B , N_{AB} . При этом требования множества N_{AB} (множества N_{BA}) обслуживаются прибором A и прибором B в одной и той же последовательности, удовлетворяющей условию (4.1) (условию (4.2)), а требования множеств N_A и N_B обслуживаются в произвольном порядке.*

Следовательно, оптимальное расписание может быть построено в результате выполнения не более чем $O(m \log m)$ элементарных действий, где $m = \max \{|N_{AB}|, |N_{BA}|\}$.

4.3. Покажем, что без увеличения общего времени обслуживания требований оптимальное (по быстродействию) расписание s^* можно преобразовать в расписание, при котором каждый прибор функционирует без простоев, т.е. обслуживает требования непрерывно одно за другим.

Действительно, как показано в предыдущем пункте, требования множества N'_{AB} (множества N'_{BA}) обслуживаются прибором A (прибором B) непрерывно одно за другим начиная с момента $t = 0$, а требования множества N_{BA} (множества N_{AB}) обслуживаются прибором A (прибором B) непрерывно одно за другим начиная с момента $d_A(\pi, \pi')$ (с момента $d_B(\pi, \pi')$).

Если $d_A(\pi, \pi') = \sum_{k \in N'_{AB}} t_{kA} + h$, $h > 0$, то $d_B(\pi, \pi') = \sum_{k \in N'_{BA}} t_{kB} \geq d_A(\pi, \pi')$ и можно увеличить моменты начала обслуживания требований множества N_{AB} прибором A на величину h без изменения общего времени обслуживания требований. Аналогично, если $d_B(\pi, \pi') = \sum_{k \in N'_{BA}} t_{kB} + h$, $h > 0$, то $d_A(\pi, \pi') = \sum_{k \in N'_{AB}} t_{kA} \geq d_B(\pi, \pi')$ и без изменения общего вре-

мени обслуживания требований можно увеличить на величину h моменты начала обслуживания требований множества N_{BA} прибором B . В обоих случаях каждый прибор будет обслуживать требования непрерывно одно за другим: один прибор — начиная с момента $t = 0$, а другой — начиная с момента $t = h$.

	N_{AB}				N_{BA}			N_A		N_B	
k	1	2	3	4	5	6	7	8	9	10	11
t_{kA}	3	2	1	1	2	4	3	1	2	—	—
t_{kB}	2	1	2	1	4	8	9	—	—	2	1

4.4. П р и м е р. При подготовке радиопередачи требуется прослушать и записать на магнитную ленту n музыкальных произведений. Для этого можно использовать аппаратуру воспроизведения A и аппаратуру звукозаписи B . Музыкальные произведения N_{AB} необходимо вначале прослушать на аппаратуре A , а затем дополнить комментариями на аппаратуре B . Музыкальные произведения N_{BA} необходимо записать на аппаратуре B , а затем прослушать на аппаратуре A . Музыкальные произведения N_A достаточно только прослушать, а N_B — только записать.

Необходимо составить оптимальное по быстродействию расписание подготовки радиопередачи. Исходные данные задачи приведены в табл. 4.1.

Упорядочим требования множества N_{AB} так, чтобы полученная последовательность π_{AB} удовлетворяла условию (4.1): $\pi_{AB} = (3, 4, 1, 2)$. Требования множества N_{BA} упорядочим так, чтобы полученная последовательность π_{BA} удовлетворяла условию (4.2): $\pi_{BA} = (6, 7, 5)$.

Таким образом, при расписании s^* музыкальные произведения прослушиваются на аппаратуре A в порядке $\pi = (3, 4, 1, 2, 8, 9, 6, 7, 5)$ и записываются на аппаратуре B в порядке $\pi' = (6, 7, 5, 10, 11, 3, 4, 1, 2)$.

На рис. 4.1 представлено расписание, соответствующее последовательностям π и π' .

Для того чтобы аппаратура работала непрерывно, достаточно увеличить моменты времени начала воспроизведения музыкальных произведений на аппаратуре A , как показано на рис. 4.2.

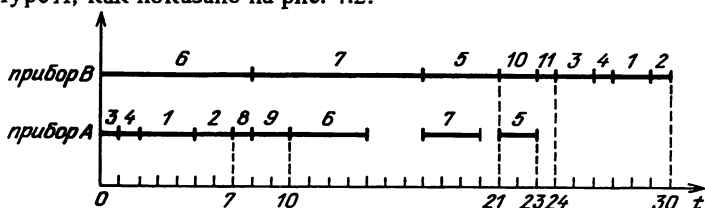


Рис. 4.1

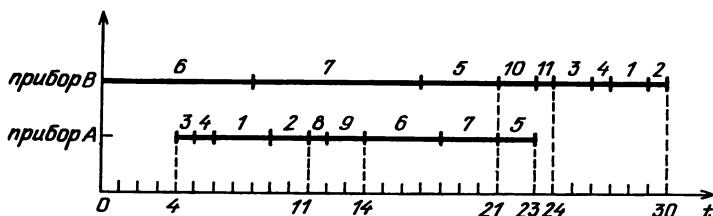


Рис. 4.2

4.5. Покажем, что задача построения оптимального по быстрдействию расписания обслуживания множества N одновременно поступающих требований двумя приборами является NP -трудной в случае, когда значения $r_i = 3$ хотя бы для двух требований из N .

Рассмотрим вначале случай, когда в процессе выполнения операций допускаются прерывания. Сформулируем соответствующую задачу распознавания.

В обслуживающую систему, состоящую из приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Для каждого требования $i \in N$ задан маршрут обслуживания L^i ; причем число стадий обслуживания $r_i \leq 3$. Для каждой операции (i, L, q) , $i \in N, L \in \{A, B\}$, $1 \leq q \leq r_i$, известна длительность $t_{iLq} > 0$ ее выполнения прибором $L = L^i_q$. Длительности выполнения операций представляют собой целые числа. Необходимо определить, существует ли расписание s^0 (возможно, с прерываниями в процессе выполнения операций) обслуживания требований множества N такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Построим полиномиальное сведение задачи о разбиении (см. вводную часть данной главы) к сформулированной задаче распознавания.

Положим $n = n_0 + 2$, $y = 10E$, $L^i = (B, A)$, $t_{iB1} = t_{iA2} = e_i$ для всех требований $i = 1, \dots, n_0$;

$$L^{n+1} = (A, B, A), t_{n+1, A1} = t_{n+1, B2} = E, t_{n+1, A3} = 6E;$$

$$L^{n+2} = (B, A, B), t_{n+2, A2} = t_{n+2, B3} = E, t_{n+2, B1} = 6E.$$

Покажем, что для построенной задачи расписание s^0 существует тогда и только тогда, когда существует разбиение множества N^0 на подмножества N^0_1 и N^0_2 , для которых $E_1 = E_2$.

Если указанное разбиение множества N^0 существует, то существует и расписание с общим временем обслуживания требований, равным y . Одно такое расписание (без прерываний в процессе выполнения операций) приведено на рис. 4.3.

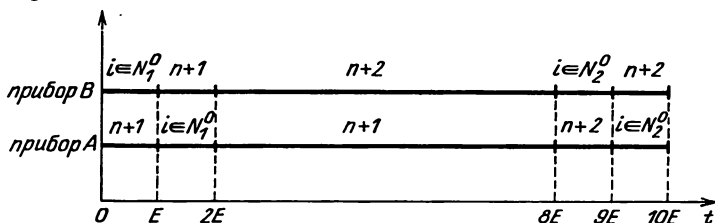


Рис. 4.3

Если задача о разбиении не имеет решения, то общее время обслуживания требований при любом расписании (в том числе и с прерываниями в процессе выполнения операций) больше y .

Действительно, предположим, что существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y$. Поскольку суммарная длительность всех операций, выполняемых как прибором A , так и прибором B , равна $10E = y$, то при расписании s^0 приборы A и B во временном интервале $(0, 10E]$ не простаивают.

Обозначим через N_1^0 множество требований $i \in N, 1 \leq i \leq n_0$, которые при расписании s^0 обслуживаются (возможно, частично) прибором A в интервале $(0, 2E]$. Кроме операций вида $(i, A, 2), i \in N_1^0$, прибор A в указанном интервале может выполнять только операцию $(n+1, A, 1)$. Следовательно,

$$\sum_{i \in N_1^0} t_{iA2} \geq 2E - t_{n+1, A1} = E.$$

Поскольку $L^i = (B, A)$ для всех $i = 1, \dots, n_0$, то множество N_1^0 может содержать только такие требования i , для которых операции $(i, B, 1)$ выполняются полностью во временном интервале $(0, 2E]$. Таким образом, имеет место соотношение

$$\sum_{i \in N_1^0} t_{iB1} \geq \sum_{i \in N_1^0} t_{iA2} \geq E. \quad (4.3)$$

Заметим, что при любом расписании с общим временем обслуживания требований, равным u , выполнение операции $(n+1, B, 2)$ должно быть завершено не позднее момента времени $t = 4E$, а выполнение операции $(n+2, B, 1)$ — не позднее момента времени $t = 8E$. Следовательно, для обслуживания требований из множества $N^0 = \{1, 2, \dots, n_0\}$ прибором B может быть использовано не более чем $8E - 6E - E = E$ единиц времени во временном интервале $(0, 8E]$ (и тем более в интервале $(0, 2E]$).

Так как $N_1^0 \subseteq N^0$, то $\sum_{i \in N_1^0} t_{iB1} \leq E$ и, учитывая (4.3), получаем

$\sum_{i \in N_1^0} t_{iB1} = \sum_{i \in N_1^0} e_i = E$, что противоречит предположению о том, что задача о разбиении не имеет решения.

Нетрудно видеть, что для реализации описанного сведения требуется не более $O(n_0)$ элементарных действий.

Таким образом, задача построения оптимального по быстродействию расписания (с прерываниями) является NP -трудной в случае, когда хотя бы для двух требований множества N значения $r_i = 3$, а для остальных требований $r_i = 2$.

4.6. Если прерывания в процессе выполнения каждой операции запрещены, то задача построения оптимального по быстродействию расписания оказывается NP -трудной уже в случае, когда для одного требования из N значение $r_i = 3$, а для остальных требований $r_i = 1$.

Построим полиномиальное сведение задачи о разбиении к соответствующей задаче распознавания.

Положим $n = n_0 + 1, u = 2E + 1, L^i = (A), t_{iA1} = e_i$ для требований $i = 1, \dots, n_0$; $L^n = (B, A, B), t_{nB1} = t_{nB3} = E, t_{nA2} = 1$.

Покажем, что для этой задачи расписание s^0 (без прерываний в процессе выполнения операций), при котором $\bar{t}_{\max}(s^0) = u$, существует тогда и только тогда, когда существует разбиение множества $N^0 = \{1, 2, \dots, n_0\}$ на подмножества N_1^0 и N_2^0 , для которых $\sum_{i \in N_1^0} e_i = \sum_{i \in N_2^0} e_i = E$.

Если указанное разбиение множества N^0 существует, то существует и расписание s^0 , для которого $\bar{t}_{\max}(s^0) = u$ (рис. 4.4).

Если задача о разбиении не имеет решения, то докажем от противного, что общее время обслуживания требований при любом расписании без прерываний в процессе выполнения операций должно быть больше u .

Предположим, что существует расписание s^0 (без прерываний), для которого $\bar{t}_{\max}(s^0) \leq y$.

Поскольку суммарная длительность операций по обслуживанию требования n равна y , то при любом расписании (без прерываний) с общим временем обслуживания требований, равным y , операция $(n, B, 2)$ единичной длины должна выполняться во временном интервале $(E, E+1]$.

Так как процесс обслуживания каждого требования $i \in N^0$ состоит из единственной операции, $L^i = (B)$, то при расписании s^0 нет прерываний в

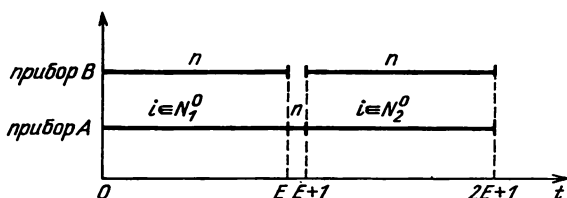


Рис. 4.4

процессе обслуживания требований $i \in N^0$. Обозначим через N_1^0 подмножество требований из N^0 , которые обслуживаются во временном интервале $(0, E]$.

Поскольку суммарная длительность всех операций, выполняемых прибором B , равна y , то при любом расписании с общим временем обслуживания требований, равным y , этот прибор в интервале $(0, y]$ не простаивает. Получаем $\sum_{i \in N_1^0} e_i = E$, что противоречит предположению о том, что

задача о разбиении не имеет решения.

Для реализации описанного сведения требуется $O(n_0)$ элементарных действий.

4.7. В п. 3.6 предыдущего параграфа показано, что при $M = 2, d_i = 0, i \in N, t_{iLq} \in \{1, 2\}$ задача построения оптимального по быстродействию расписания (без прерываний) является NP -трудной в сильном смысле.

Покажем, что аналогичный результат имеет место и в случае, когда $M = 2, d_i = 0, i \in N, t_{iLq}$ — целые числа и разрешены прерывания. Сформулируем соответствующую задачу распознавания.

В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество требований $N = \{1, 2, \dots, n\}$. Для каждого требования $i \in N$ заданы маршрут обслуживания $L^i = (L_1^i, L_2^i, \dots, L_{r_i}^i)$ и целочисленные длительности t_{iLq} выполнения всех операций (i, L, q) , $L = L^i, q = 1, \dots, r_i$. Необходимо определить, существует ли расписание s^0 (возможно, с прерываниями в процессе выполнения операций) обслуживания требований множества N такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Построим полиномиальное сведение задачи о 3-разбиении (см. вводную часть данной главы) к сформулированной задаче распознавания.

Положим $n = 3n_0 + 1, y = 2n_0E, L^i = (A, B), t_{iA1} = t_{iB2} = e_i$ для всех требований $i = 1, \dots, 3n_0; L^n = ([B, A]^{n_0}), t_{nLq} = E$ для всех $q = 1, \dots, 2n_0$. Напомним, что $[\cdot]^h$ означает последовательность, состоящую из h повторений стоящего в скобках выражения.

Покажем, что расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$ и возможны прерывания в процессе выполнения операций, существует тогда и только тогда, когда имеет решение задача о 3-разбиении,

Пусть существует разбиение множества $N^0 = \{1, 2, \dots, 3n_0\}$ на n_0 трехэлементных подмножеств $N_1^0, N_2^0, \dots, N_{n_0}^0$ таких, что $\sum_{i \in N_j^0} e_i = E$

при $j = 1, \dots, n_0$. Тогда существует и расписание с общим временем обслуживания требований, равным $y = 2n_0E$. Одно такое расписание (без прерываний) приведено на рис. 4.5.

Пусть существует расписание s^0 , при котором $\bar{t}_{\max}(s^0) = y$. Так как суммарная длительность выполнения всех операций прибором A (прибором B) равна

$$\sum_{i=1}^{3n_0} e_i + n_0E = 2n_0E = y,$$

то при расписании s^0 как прибор A , так и прибор B во временном интервале $(0, 2n_0E]$ не простаивает. Поскольку

$$\sum_{q=1}^{r_n} t_{nL}^n q = 2n_0E = y,$$

то при расписании s^0 операции по обслуживанию требования n должны выполняться непрерывно одна за другой начиная с момента времени $t = 0$, как показано на рис. 4.5. Из равенства $t_{iA1} = t_{iB2}$, $i = 1, \dots, 3n_0$, следует, что для того, чтобы при расписании с общим временем обслуживания требований, равным y , не было простоев прибора B в интервале $(E, 2E]$, необходимо и достаточно, чтобы в интервале $(0, E]$ были полностью выполнены операции вида $(i_1, A, 1)$, $(i_1 + 1, A, 1), \dots, (i_l, A, 1)$, где $i_k \in N^0$, $k = 1, \dots, l$, и только они, а в интервале $(E, 2E]$ были полностью выполнены операции $(i_1, B, 2)$, $(i_1 + 1, B, 2), \dots, (i_l, B, 2)$. В силу неравенств $E/4 < e_i < E/2$, $i = 1, \dots, 3n_0$, получаем $l = 3$.

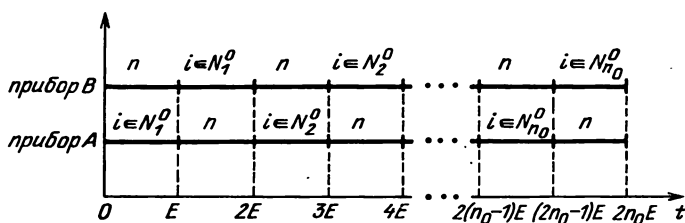


Рис. 4.5

Таким образом, выделено трехэлементное подмножество $N_1^0 = \{i_1, i_2, i_3\}$ множества N^0 , для которого

$$\sum_{i \in N_1^0} e_i = \sum_{i \in N_1^0} t_{iB2} = E.$$

Далее индукцией по j можно показать, что при расписании s^0 в интервале $(2E(j-1), 2Ej]$, $1 < j \leq n_0$, начинается и завершается обслуживание точно трех требований из множества N^0 . В результате получим разбиение мно-

жества N^0 на трехэлементные подмножества $N_1^0, N_2^0, \dots, N_{n_0}^0$ такие, что $\sum_{i \in N_j^0} e_i = E, 1 \leq j \leq n_0$.

Реализация описанного сведения требует выполнения $O(n_0)$ элементарных действий. Это сведение является как полиномиальным, так и псевдополиномиальным. Следовательно, задача построения оптимального по быстродействию расписания (как с прерываниями, так и без них) обслуживания двумя приборами n одновременно поступающих требований является NP -трудной в сильном смысле даже в том случае, когда для одного из требований множества N значение r_i не превосходит $2(n-1)$, а для каждого из остальных требований $r_i = 2$.

4.8. Поскольку задача с одинаковыми маршрутами L^i обслуживания всех требований $i \in N$ является частным случаем рассматриваемой в этом параграфе задачи, то все результаты относительно NP -трудности задач, приведенные в главе 1, справедливы и для соответствующих аналогов при различных маршрутах L^i обслуживания требований $i \in N$.

В частности, задача построения оптимального по быстродействию расписания (как с прерываниями, так и без прерываний) обслуживания требований $i \in N$, поступающих в моменты времени $d_i \geq 0$ в обслуживающую систему, состоящую из M приборов, NP -трудна в следующих случаях:

- а) $M = 3, d_i = 0, r_i \leq 2, i = 1, \dots, n$;
- б) $M = 2, d_i \in \{d, d'\}, r_i \leq 2, i = 1, \dots, n$,

и NP -трудна в сильном смысле в случае

- в) $M = 2, r_i \leq 2, i = 1, \dots, n$.

§ 5. Общее время обслуживания требований без задержек

В этом параграфе рассматривается задача построения оптимального по быстродействию расписания без задержек в процессе обслуживания каждого требования.

5.1. В систему, состоящую из двух приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Маршрут L^i обслуживания требования $i \in N$ представляет собой чередующуюся последовательность приборов $(L_1^i, L_2^i, \dots, L_{r_i}^i)$, в которой $L_q^i \in \{A, B\}, q = 1, \dots, r_i, L_q^i \neq L_{q+1}^i, q = 1, \dots, r_i - 1$. Длительности t_{iL_q} выполнения всех операций (i, L, q) предполагаются заданными. Каждое требование должно обслуживаться непрерывно и без задержек, т.е. оптимальное по быстродействию расписание s^* ищется в классе расписаний s , при которых $t_{iL_q}^0 = \bar{t}_{iL_q-1, q-1}, q = 2, \dots, r_i$, и $\bar{t}_i(s) = t_i^0(s) + \sum_{q=1}^{r_i} t_{iL_q}^i, i \in N$.

Как показано в § 3 гл. 1, если $r_i = 2, i \in N$, все L^i одинаковы и все $t_{iL_q} > 0$, то расписание s^* может быть построено за $O(n \log n)$ элементарных действий. Если же при этих условиях допускаются нулевые длительности выполнения операций, то задача построения расписания s^* является NP -трудной в сильном смысле.

Таким образом, если $r_i \leq 2, i \in N$, все $t_{iLq} > 0$ и допускаются различные маршруты прохождения приборов — $L^i \in \{(A, B), (B, A), (A), (B)\}$, то задача построения расписания s^* является NP-трудной в сильном смысле, поскольку NP-трудным в сильном смысле является ее частный случай при $L^i \in \{(A, B), (A), (B)\}$.

Покажем, что если $L^i \in \{(A, B), (B, A)\}$ и $t_{iLq} > 0$, то задача построения расписания s^* также NP-трудна в сильном смысле. Доказательство проведем в три этапа.

5.2. Введем в рассмотрение так называемую задачу о четырехэлементных суммах. Имеется четыре множества $U = \{u_1, u_2, \dots, u_p\}$, $V = \{v_1, v_2, \dots, v_p\}$, $W = \{w_1, w_2, \dots, w_p\}$ и $X = \{x_1, x_2, \dots, x_p\}$, элементами каждого из которых являются натуральные необязательно различные числа, и пусть $Q = \frac{1}{p} \sum_{i=1}^p (u_i + v_i + w_i + x_i)$ — натуральное число. Требуется определить, существует ли разбиение множества $U \cup V \cup W \cup X$ на четырехэлементные подмножества R_m , $1 \leq m \leq p$, такие, что в каждое подмножество R_m входит по одному элементу из каждого множества U, V, W и X , причем их сумма равна Q .

Покажем, что эта задача является NP-трудной в сильном смысле. Рассмотрим известную задачу о трехмерном паросочетании (см. вводную часть данной главы).

Число упорядоченных троек в E , m -я компонента которых равна $i \in N^0$, обозначим через $\mu_m(i)$, $1 \leq m \leq 3$. Можно считать, что все $\mu_m(i) \geq 1$, так как в противном случае E не содержит требуемого паросочетания. Пусть $h = n_0 + 1$ и $H = 1 + h + h^2 + h^3$. Для каждой тройки $(i, j, k) \in E$, компоненты которой необязательно попарно различны, положим $U(i, j, k) = H - (i + jh + kh^2)$ и определим множество U как совокупность всех $U(i, j, k)$. Для каждого $i \in N^0$ положим

$$V(i, 1) = i + 3h^4, \quad V(i, l) = i + 5h^4, \quad 2 \leq l \leq \mu_1(i),$$

$$W(i, 1) = ih + 3h^4, \quad W(i, l) = ih + 2h^4, \quad 2 \leq l \leq \mu_2(i),$$

$$X(i, 1) = ih^2 + 3h^4, \quad X(i, l) = ih^2 + 2h^4, \quad 2 \leq l \leq \mu_3(i),$$

и определим множество V как совокупность всех $V(i, 1)$ и $V(i, l)$, множество W — как совокупность всех $W(i, 1)$ и $W(i, l)$ и множество X — как совокупность всех $X(i, 1)$ и $X(i, l)$.

Заметим, что $\sum_{i \in N^0} \mu_m(i) = |E|$, $1 \leq m \leq 3$, и поэтому $|U| = |V| = |W| = |X| = |E|$. Очевидно, $Q = H + 9h^4$. Максимальный элемент в построенных множествах ограничен полиномом четвертой степени, зависящим от $h \leq |E|$. Число элементарных действий, необходимое для построения этих множеств, не превосходит $O(h^4 |E|)$.

Пусть множество троек E содержит требуемое паросочетание E' . Тогда имеет решение и построенная задача о четырехэлементных суммах, причем подмножества R_m , $1 \leq m \leq |E|$, могут быть найдены следующим образом. Если тройка $(i, j, k) \in E'$, то одно из искоемых множеств R_m состоит из элементов $U(i, j, k), V(i, 1), W(i, 1), X(i, 1)$. Если же тройка (i, j, k) не вошла в паросочетание E' , то можно сформировать множество R_m , состоящее из 10^*

элементов $U(i, j, k)$, $V(i, l_1)$, $W(j, l_2)$, $X(k, l_3)$, где $2 \leq l_1 \leq \mu_1(i)$, $2 \leq l_2 \leq \mu_2(j)$, $2 \leq l_3 \leq \mu_3(k)$.

Пусть имеет решение построенная задача о четырехэлементных суммах и R_m — одно из найденных подмножеств, $1 \leq m \leq |E|$. Покажем, что в этом случае имеет решение и задача о трехмерном паросочетании.

Предварительно докажем, что если подмножество R_m содержит элемент $U(i, j, k)$, то оно содержит также элементы $V(i, l_1)$, $W(j, l_2)$ и $X(k, l_3)$. Рассмотрим величину $Q = H + 9h^4$ как число, представленное в системе счисления по основанию h . "Вклад" величины $H = 1 + h + h^2 + h^3$ в значение Q соответствует цифрам четырех младших разрядов, в то время как величина $9h^4$ определяет цифру более старшего разряда в записи числа Q в системе счисления по основанию $h = n_0 + 1$. Для того чтобы цифра старшего разряда оказалась равной 9, необходимо, чтобы либо $l_1 = l_2 = l_3 = 1$, либо $2 \leq l_1 \leq \mu_1(i)$, $2 \leq l_2 \leq \mu_2(j)$, $2 \leq l_3 \leq \mu_3(k)$.

Таким образом, среди подмножеств R_m , $1 \leq m \leq |E|$, имеется ровно n_0 таких, что $R_m = \{U(i, j, k), V(i, 1), W(j, 1), X(k, 1)\}$ для некоторой тройки $(i, j, k) \in E$.

Очевидно, множество таких троек образует искомое трехмерное паросочетание E' из n_0 троек.

5.3. Введем в рассмотрение задачу распознавания, которую в дальнейшем будем называть задачей о равных парах. Имеется четыре множества $U = \{u_1, u_2, \dots, u_p\}$, $V = \{v_1, v_2, \dots, v_p\}$, $W = \{w_1, w_2, \dots, w_p\}$ и $X = \{x_1, x_2, \dots, x_p\}$, каждое из которых состоит из натуральных необязательно различных чисел. Требуется определить, существует ли разбиение множества $U \cup V \cup W \cup X$ на четырехэлементные подмножества R_m , $1 \leq m \leq p$, такие, что в каждое из них входит по одному элементу из каждого множества U , V , W и X и для каждого $R_m = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$ имеет место равенство $u_{i_1} + v_{i_2} = w_{i_3} + x_{i_4}$.

Покажем, что к этой задаче полиномиально сводится задача о четырехэлементных суммах. Пусть $U' = \{u'_1, u'_2, \dots, u'_p\}$, $V' = \{v'_1, v'_2, \dots, v'_p\}$, $W' = \{w'_1, w'_2, \dots, w'_p\}$ и $X' = \{x'_1, x'_2, \dots, x'_p\}$ — множества, определяющие задачу о четырехэлементных суммах, и $Q' = \frac{1}{p} \sum_{i=1}^p (u'_i + v'_i + w'_i + x'_i)$.

Для каждого i , $1 \leq i \leq p$, положим $u_i = Q' + 2u'_i$, $v_i = Q' + 2v'_i$, $w_i = 2Q' - 2w'_i$, $x_i = 2Q' - 2x'_i$.

Предположим, что задача о четырехэлементных суммах имеет решение и R'_m , $1 \leq m \leq p$, — найденные подмножества. Тогда имеет решение и задача о равных парах, причем искомые подмножества R_m , $1 \leq m \leq p$, могут быть построены, например, следующим образом. Пусть $R'_m = \{u'_{i_1}, v'_{i_2}, w'_{i_3}, x'_{i_4}\}$. Положим $R_m = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$. Так как $u'_{i_1} + v'_{i_2} = Q' - w'_{i_3} - x'_{i_4}$, то $u_{i_1} + v_{i_2} = w_{i_3} + x_{i_4}$.

Пусть имеет решение задача о равных парах и R_m , $1 \leq m \leq p$, — найденные подмножества. Если $R_m = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$, то можно положить $R'_m = \{u'_{i_1}, v'_{i_2}, w'_{i_3}, x'_{i_4}\}$. Поскольку $u_{i_1} + v_{i_2} = w_{i_3} + x_{i_4}$, то, очевидно, $u'_{i_1} + v'_{i_2} + w'_{i_3} + x'_{i_4} = 0$.

Для реализации этого сведения достаточно выполнить $O(p)$ элементарных действий, и следовательно, задача о равных парах является NP-трудной в сильном смысле.

5.4. Рассмотрим задачу распознавания, соответствующую задаче построения оптимального по быстродействию расписания s^* без задержек в процессе обслуживания требований при условии, что допускаются только два возможных маршрута (A, B) и (B, A) обслуживания требований приборами. Необходимо определить, существует ли расписание s^0 обслуживания (без задержек) множества требований N такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче полиномиально сводится задача о равных парах. Без потери общности в задаче о равных парах можно предполагать, что $\sum_{i=1}^p u_i + \sum_{i=1}^p v_i = \sum_{i=1}^p w_i + \sum_{i=1}^p x_i$, так как в противном случае эта задача не имеет решения.

Положим $n = 4p$, $y = 2p + 3pz + 4 \sum_{i=1}^p (u_i + v_i)$, где $z = 8 \max_{1 \leq i \leq p} \{u_i, v_i, w_i, x_i\}$.

Введем в рассмотрение следующие четыре множества требований: множество $N_U = \{1, 2, \dots, p\}$, полагая $L^i = (A, B)$, $t_{iA1} = 1$ и $t_{iB2} = 4u_i$ для каждого требования $i \in N_U$; множество $N_V = \{p+1, p+2, \dots, 2p\}$, полагая $L^i = (B, A)$, $t_{iB1} = 4v_i + 3z$ и $t_{iA2} = 1$ для каждого $i \in N_V$; множество $N_W = \{2p+1, 2p+2, \dots, 3p\}$, полагая $L^i = (A, B)$, $t_{iA1} = 4w_i + z$ и $t_{iB2} = 1$ для каждого $i \in N_W$, и множество $N_X = \{3p+1, 3p+2, \dots, 4p\}$, полагая $L^i = (B, A)$, $t_{iB1} = 1$ и $t_{iA2} = 4x_i + 2z$ для каждого $i \in N_X$.

Покажем, что для построенной задачи расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y$ и нет задержек в процессе обслуживания требований множества $N = N_U \cup N_V \cup N_W \cup N_X$, существует тогда и только тогда, когда имеет решение задача о равных парах. Заметим, что поскольку величина y равна полусумме длительностей выполнения всех операций по обслуживанию требований множества N приборами A и B , то при любом расписании с общим временем обслуживания требований, равным y , ни один из приборов A и B не простаивает во временном интервале $(0, y]$, а расписания с общим временем обслуживания требований, меньшим y , не существует.

Пусть задача о равных парах имеет решение и R_m , $1 \leq m \leq p$, — найденные подмножества множества $U \cup V \cup W \cup X$. Расписание s^0 может быть построено следующим образом. Если $R_1 = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$, то во временном интервале $(0, \tau]$, где $\tau = 4(u_{i_1} + v_{i_2}) + 3z + 2$, прибор A обслуживает требования в последовательности $i_1, 3p + i_4, 2p + i_3, p + i_2$, а прибор B — в последовательности $3p + i_4, i_1, p + i_2, 2p + i_3$ (рис. 5.1).

Вслед за указанными требованиями аналогично обслуживаются требования, определяемые подмножеством R_2 , и т.д.

Пусть существует расписание s^0 (без задержек), при котором $\bar{t}_{\max}(s^0) \leq y$. Предположим, что при расписании s^0 прибором A первым обслуживается требование i , а прибором B — требование j . Требование i должно принадлежать множеству N_U или множеству N_W , так как в противном случае

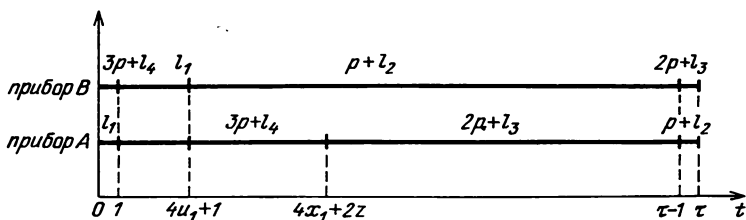


Рис. 5.1

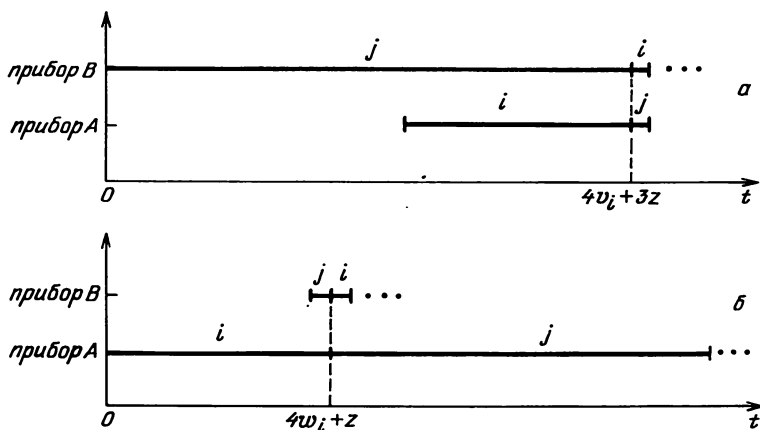


Рис. 5.2

прибор А будет простаивать, пока требование из множества $N_V \cup N_X$ будет обслуживаться прибором В. Аналогично $j \in N_V \cup N_X$.

Покажем, что если $i \in N_W$, то хотя бы один из приборов будет простаивать. Действительно, если $j \in N_V$, то простаивает прибор А, поскольку неравенство $t_{iA1} < t_{jB1}$ выполняется для требования $i \in N_W$ и каждого требования $j \in N_V$ и при расписании s^0 требования обслуживаются без задержки (рис. 5.2, а). Если же $j \in N_X$, то простаивает прибор В, поскольку $t_{jB1} = 1 < t_{iA1}$ при $i \in N_W$, $j \in N_X$ и требования обслуживаются без задержки (рис. 5.2, б).

Таким образом, $i \in N_U$. Отсюда следует, что при расписании s^0 имеем $j \in N_X$, так как в противном случае прибор В будет простаивать в интервале $(0, 1]$.

По определению величины z получаем, что $t_{jA2} - t_{iB2} > 1$ для всех $i \in N_U$ и всех $j \in N_X$. Следовательно, после выполнения операции $(i, B, 2)$ прибором В должно обслуживаться требование k из множества N_V (в противном случае либо прибор В будет простаивать, либо требование k будет обслуживаться с задержками).

На рис. 5.3 представлено расписание обслуживания без задержек требований i, j и k , принадлежащих множеству N_U, N_X и N_V соответственно.

По определению величины z получаем, что $t_{iB2} + t_{kB1} - t_{jA2} > 1$ для всех $i \in N_U$, $j \in N_X$ и $k \in N_V$, и для того, чтобы исключить простои прибо-

ров и задержки в обслуживании требований, необходимо, чтобы после выполнения операции $(j, A, 2)$ прибором A обслуживалось требование l из множества N_W и, кроме того, выполнялось равенство $t_{lB2} + t_{kB1} = t_{jA2} + t_{lA1}$.

Таким образом, элементы u_i, v_k, w_j и x_l множеств U, V, W и X соответственно образуют подмножество R_1 , для которого $u_i + v_k = w_j + x_l$. Рассуждая аналогичным образом, нетрудно построить и остальные подмножества $R_m, 2 \leq m \leq p$, составляющие решение задачи о равных парах.

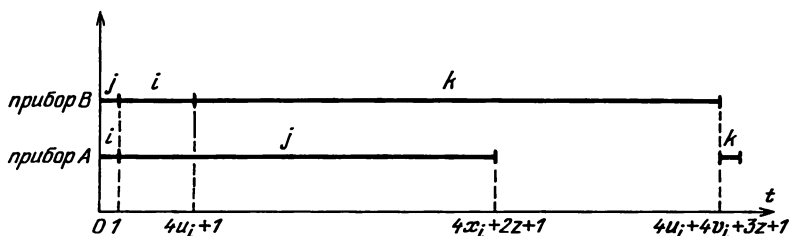


Рис. 5.3

Реализация описанного сведения требует выполнения $O(p)$ элементарных действий.

Таким образом, задача построения оптимального по быстродействию расписания без задержек в процессе обслуживания требований является NP-трудной в сильном смысле даже в том случае, когда система состоит из двух приборов A и B и допускаются два возможных маршрута (A, B) и (B, A) обслуживания требований.

5.5. Покажем, что при единичных длительностях всех операций задача построения оптимального по быстродействию расписания без задержек в процессе обслуживания каждого требования является NP-трудной. Сформулируем соответствующую задачу распознавания.

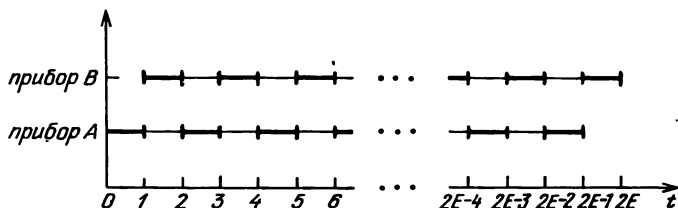


Рис. 5.4

В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительность $t_{lLq} = 1$ для всех $i \in N, L = L_q^i, q = 1, \dots, r_i$. Необходимо определить, существует ли расписание s^0 без задержек в процессе обслуживания требований N такое, что $\bar{t}_{\max}(s^0) \leq u$ для заданного числа u .

Построим полиномиальное сведение задачи о разбиении (см. вводную часть данной главы) к сформулированной задаче распознавания.

Положим $n = n_0$, $y = 2E + 1$, $L^i = ([A, B]^{e_i})$, $t_{iL}^{t_{q^q}} = 1$, $i \in N$, $q = 1, \dots, 2e_1$.

Нетрудно показать, что для построенной задачи расписание s^0 существует тогда и только тогда, когда существует разбиение множества N^0 на подмножества N_1^0 и N_2^0 , для которых $E_1 = E_2$.

Действительно, если указанное разбиение множества N^0 существует, то при расписании, приведенном на рис. 5.4, $\bar{t}_{\max}(s^0) = y$ и все требования обслуживаются без задержек (обслуживание требований множества N_1^0 изображено тонкой линией, а требований множества N_2^0 — жирной).

Если же указанного разбиения множества N^0 не существует, то очевидно, что при любом расписании s обслуживания требований N приборами A и B либо выполняется неравенство $\bar{t}_{\max}(s) > y$, либо хотя бы одно из требований не обслуживается непрерывно (без задержек).

Поскольку для реализации описанного сведения достаточно выполнить $O(n_0)$ элементарных действий, то задача построения расписания s^0 является NP -трудной.

§ 6. Библиографическая справка

Системы с различными маршрутами обслуживания требований наиболее широко распространены на практике. Как отмечали В. Дудек, М. Смит и С. Панвалкар [507, 724], в 57% из 300 обследованных промышленных компаний США маршруты обслуживания требований были различными.

6.1. Графический подход к построению оптимального по быстродействию расписания обслуживания двух требований M приборами предложен С.Акерсом и Дж.Фридманом [422, 423]. Основанные на этом подходе полиномиальные алгоритмы решения данной задачи в случае, когда прерывания запрещены, предложены В.Шварцем [794], В.В.Сервахом [281], Н.И.Глебовым [67], В. Хардгрэйвом и Г.Немхаузером [592]. Алгоритм с наилучшей оценкой трудоемкости $O(\omega \log \omega)$ описан В.В.Сервахом [281] (здесь ω — число особых прямоугольников в графическом представлении задачи). В [67] предложен полиномиальный алгоритм решения указанной задачи при разрешении прерываний. Оценки общего времени обслуживания двух требований M приборами получены В.Э.Хенкиным [381], Н.И.Глебовым и В.А.Перепелицей [68, 70].

Описанные в § 1 полиномиальные алгоритмы решения задачи оптимального обслуживания двух требований M приборами при произвольной неубывающей целевой функции предложены Ю.Н.Сотсковым [302]. Вопросы программной реализации этих алгоритмов рассматривались А.В.Тузиковым [360].

Попытки обобщения графического подхода на случай $M > 2$ предпринимались в [281, 688, 794, 813, 822]. В.В.Сервахом [281] предложен псевдополиномиальный алгоритм трудоемкости $O(MT \log M)$ для построения оптимального по быстродействию расписания обслуживания трех требований M приборами при условии, что число выполняемых каждым прибором операций ограничено, длительности всех операций целые числа и их сумма равна T .

6.2. В.Г.Тимковский [355], Н.Хифитц и И.Адири [595] предложили полиномиальные алгоритмы построения оптимального по быстродействию расписания обслуживания двумя приборами n одновременно поступающих требований с единичными длительностями операций. Трудоемкость этих алгоритмов — $O(R)$, где R — общее число выполняемых операций. При неодновременном поступлении требований оптимальное по быстродействию расписание может быть построено за $O(R \log R)$ действий [481, 595]. Для решения сопряженной задачи минимизации максимального временного смещения П.Брукер [482] предложил алгоритм трудоемкости $O(R)$ при условии целочисленности директивных сроков.

Приведенное в пп. 4.1–4.4 обобщение алгоритма С.Джонсона [612] получено Дж.Джексоном [608].

6.3. Описанный в § 8 гл. 1 подход к построению полиномиальных алгоритмов с априорными оценками абсолютной или относительной погрешности получаемого решения можно использовать и для систем с различными маршрутами обслуживания требований.

С.В.Севастьяновым [278] предложен алгоритм трудоемкости $O(n^2 M^2 r^2)$ построения расписания s^0 , при котором $\delta \leq r(M^2 r^2 + r - 2)t^*$, где $\delta = \bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*)$, $r = \max\{r_i | i \in N\}$ и $t^* = \max\{t_{iLq} | i \in N, L = L_i^q, q = 1, \dots, r_i\}$. Для случая, когда каждая операция может выполняться любым прибором из заранее выделенного множества приборов, в [278] предложен алгоритм трудоемкости $O(M^2 n^2 r^2 + (m + M)^3 \log m)$ построения расписания s^0 , при котором $\delta \leq (r^3 M^3 + r^2 - 2r + 2)t^*$.

Для случая $M = r = 3$ А.И.Бабушкин, А.И.Башта и И.С.Белов [17] предложили алгоритм трудоемкости $O(n^2)$ с оценкой $\delta \leq 3(3\sqrt{2} + 5)t^*$, а С.В.Севастьянов [277] — алгоритм той же трудоемкости с оценкой $\delta \leq 6t^*$.

Для обслуживающих систем, в которых часть требований обслуживается приборами в порядке $1, 2, \dots, M$, а остальные требования — в порядке $M, M - 1, \dots, 1$, С.В.Севастьянов [276] предложил алгоритм трудоемкости $O(n^2 M^2)$ построения расписания s^0 , при котором $\delta \leq (2M^2 - (M + 3)/2)t^*$. Эта же задача рассматривалась в [16], предложенный алгоритм имеет худшие оценки.

Т.Гонсалес и С.Сахни [552] показали, что для любого расписания s^0 , при котором в любой момент времени из интервала $(0, \bar{t}_{\max}(s^0)]$ функционирует хотя бы один прибор, справедливы оценки

$$\sum_{i=1}^n \bar{t}_i(s^0) / \sum_{i=1}^n \bar{t}_i(s^*) \leq n \text{ и } \bar{t}_{\max}(s^0) / \bar{t}_{\max}(s^*) \leq M, \text{ причем эти оценки}$$

достижимы при малых значениях M . В этой же работе предложен алгоритм трудоемкости $O(n \log n)$ построения расписания s^0 , при котором

$$\sum_{i=1}^n \bar{t}_i(s^0) / \sum_{i=1}^n \bar{t}_i(s^*) \leq M. \text{ В.Г.Тимковским [353] описан алгоритм трудо-}$$

емкости $O(R + Mr)$ построения расписания s^0 , при котором $\bar{t}_{\max}(s^0) / \bar{t}_{\max}(s^*) \leq \min\{M, r\}$.

6.4. Доказательству NP -трудности различных частных случаев задачи построения оптимального по быстродействию расписания обслуживания (без прерываний) n одновременно поступающих требований M приборами посвящен ряд публикаций.

Приведенное в п. 3.4 доказательство NP -трудности в сильном смысле этой задачи при $M = 3$ и $t_{iLq} = 1$ предложено Я.Ленстрой и А.Ринной Каном [652]. Ранее Я.Ленстра, А.Ринной Кан и П.Бруккер [657] доказали ее NP -трудность при целочисленных t_{iLq} . В [652] установлена NP -трудность в сильном смысле указанной задачи при $M = 2$ и $t_{iLq} \in \{1, 2\}$ (см. п. 3.6). В.Г.Тимковским [354] и Ю.Н.Сотсковым (см. п. 3.5) доказана ее NP -трудность в сильном смысле при $M = 2$, $t_{iLq} = 1$ и заданном отношении строгого порядка, граф редукции которого состоит из цепей.

Я. Ленстра, А. Ринной Кан и П. Бруккер [657] установили NP -трудность задачи в случае, когда $M = 2$, хотя бы одно $r_i = 3$, а остальные $r_i = 1$ (см. п. 4.6). М. Гэри, Д. Джонсон и Р. Сети [533] доказали NP -трудность в сильном смысле задачи в случае, когда $M = 2$, хотя бы одно $r_i = 2(n-1)$, а остальные $r_i = 2$ (см. п. 4.7). Т. Гонсалес и С. Сахни [552] показали, что при разрешении прерываний в процессе выполнения операций задача оказывается NP -трудной в случае, когда $M = 2$, $r_i = 3$ для двух требований и $r_i = 2$ для остальных требований (см. п. 4.5), и NP -трудной в сильном смысле в случае, когда $M = 2$, $r_i = n/3$ для одного требования и $r_i = 2$ для остальных требований.

С. Сахни и Ю. Чо [764] доказали NP -трудность в сильном смысле задачи построения оптимального по быстродействию расписания обслуживания без задержек n одновременно поступающих требований двумя приборами (см. § 5). В.Г. Тимковский [354] установил, что NP -трудной является задача построения оптимального по быстродействию расписания обслуживания без задержек n одновременно поступающих требований двумя приборами, когда длительности выполнения операций одинаковы.

Поскольку система с одинаковыми маршрутами обслуживания требований — частный случай рассматриваемой, то все результаты о NP -трудности задачи, приведенные в гл. 1, справедливы и для систем с различными маршрутами обслуживания требований.

После подготовки рукописи книги к изданию Ю.Н. Сотсковым была доказана NP -трудность задач оптимального по быстродействию обслуживания (с прерываниями или без прерываний операций) трех требований с фиксированными (одинаковыми или различными) маршрутами прохождения приборов. Аналогичные результаты установлены и для задач минимизации суммарного времени обслуживания трех требований.

6.5. Для решения NP -трудных задач построения оптимальных расписаний разработан ряд методов последовательного анализа вариантов, целочисленного линейного программирования, различные эвристические процедуры.

Целочисленные линейные модели описаны Е. Балашем [449], Е. Боуманом [475], Х. Вагнером [826], Х. Гринбергом [571], Дж. Данцигом [497], М. Фишером [522], А.Л. Лурье [194, 195], А. Манном [678], С. Ланзенауэром и Р. Химесом [641], Дж. Чарлтоном и С. Детом [491], Ю.А. Заком, [106], В.С. Солдастовым [289] и другими авторами. Систематическое исследование применимости методов линейного программирования для решения задач теории расписаний проведено Ю.В. Глебским и В.Н. Шевченко [71, 72, 397, 398]. Этим же вопросам посвящены работы [91, 117, 642].

Возможности использования теории двойственности при решении задач календарного планирования анализировались М. Фишером [522], В.К. Де-

миным и А.С. Чеботаревым [91], А.И. Куксой и Ю.П. Лаптиным [159 – 161], А.Л. Ильницким [113, 114] и другими.

Достаточно полное изложение теории двойственности применительно к задачам построения оптимальных расписаний содержится в монографии В.С. Михалевица и А.И. Куксы [215]. Изучение релаксированных задач, возникающих в результате замены нескольких ограничений линейной модели их линейной комбинацией, проведено М. Фишером, Б. Лагевеом, Я. Ленстрой, А. Ринной Каном [524].

Э. Андерсон [425], И.Н. Зимин и Ю.П. Иванилов [108] рассматривали возможности сведения некоторых задач теории расписаний к задачам оптимального управления.

Попытки применить декомпозиционный подход к решению задач большой размерности предприняты в [126, 429, 794, 822].

6.6. Методы построения различных классов расписаний рассматривались Дж. Геллером и Г. Логеманом [598], Б. Гиффлером и Дж. Томпсоном [540], Л. Шрейджем [768, 769], Л.П. Матюшковым и В.С. Танаевым [199], С.А. Канцдалом и О.Н. Малых [127, 129, 131], Ю.Н. Сотсковым и В.С. Танаевым [296, 297, 307, 308] и другими авторами. Исследование влияния правил предпочтения (приоритетов) на качество получаемых расписаний проведено в работах [30, 99, 199, 247, 251, 537].

Различные по сложности правила предпочтения предложены Э.Г. Иоффе [117], П.К. Выгнаном и Б.И. Кузиным [59], Н. Стейнхофом [786], В.В. Шафранским [388, 390], Дж. Джексоном [92], Н.Б. Мироносцем [260] и другими авторами. Вопросам выбора правил предпочтения, оптимальной их композиции и настройки посвящены работы В.В. Шкурбы [399], Г. Фишера и Г. Томпсона [373], У. Максвелла и М. Мехры [682], Т.П. Подчасовой [246], М.Х. Ангуладзе [7], Л.В. Кочнева [154], А.И. Семенова и В.М. Португала [251, 280], А.Б. Ароновича [9], А. Тизена [817]. См. также [131, 146, 192, 199, 222, 284, 292, 293, 437, 438, 782, 783].

Методы локальной оптимизации расписаний использовались в работах Л.И. Фейгина [370], С. Петерсона [737], Дж. Кинга и А. Спачиса [622, 782], С.А. Канцдала и О.Н. Малых [130].

Методы случайного поиска рассматривали Б. Гиффлер, Д. Томпсон и В. Ван-Нессе [66], Дж. Геллер и Г. Логеман [598], С.А. Канцдал и О.Н. Малых [127, 130, 131], В.В. Шкурба [400], П. Рандолф, Г. Суинсон и С. Эллингсен [746, 747], Г. Фишер и Г. Томпсон [373], С. Элмаграби [517], Н.Б. Мироносцкий [211] и другие. В [66, 145, 537] проведен сравнительный экспериментальный анализ разнообразных правил предпочтения. В обзоре Дж. Блекстона, Д. Филлипса и С. Хогга [467] рассмотрено 34 правила предпочтения. Авторы отмечают, что ни одно из этих правил не лучше остальных, однако можно выделить несколько правил, которые в совокупности дают хорошие результаты. В обзоре С. Панвалкара и В. Искандера [725] проведен анализ 113 правил предпочтения.

6.7. Систематическое исследование правил предпочтения проводится и в рамках изучения стохастических обслуживающих систем, в которых те или иные параметры (длительности операций, моменты поступления требований в систему, маршруты обслуживания требований и т.п.) являются случайными величинами с заданными или неизвестными законами

распределения. В работах [104, 105, 368, 369, 417, 498, 512, 515, 516, 553, 599, 600, 607, 614, 658, 693, 702, 756, 827, 836] используются методы статистического моделирования таких систем. В работах [546, 591, 825, 610] отмечаются преимущества расписаний, составленных в интерактивном режиме. Аналитические методы исследования предложены в [8, 164–166, 206, 218, 367–369, 501, 607, 693, 771]. Системы с нечеткими длительностями операций рассматривались в [371]. Обзоры работ по указанному направлению приведены в [501, 504, 535, 546, 725].

6.8. С. Гойал [558], С. Редди и С. Рамамурти [751], А. Хартвиг и Дж. Терно [594] предложили сведения задачи построения оптимального по быстродействию расписания обслуживания без задержек n требований M приборами к задаче о коммивояжере. Постановки некоторых задач теории расписаний в терминах задачи о коммивояжере и различных ее обобщений рассматривались, например, в [27, 259, 260, 335]. Метод ветвей и границ для решения задачи о коммивояжере впервые предложили Д. Литл, К. Мурти, Д. Суини и К. Кэрел [191]. В [611] приведен список работ, посвященных различным аспектам задачи о коммивояжере, который насчитывает около 600 наименований.

Одна из наиболее популярных тестовых задач построения оптимального по быстродействию расписания ($n = 10$, $M = 10$) описана в [373]. Лучшие из известных для нее расписаний получены методом ветвей и границ [524, 686]. Краткие сведения о публикациях, посвященных методам последовательного анализа вариантов, приведены в § 6 гл. 4.

Множество $N = \{1, 2, \dots, n\}$ требований обслуживается в системе, состоящей из M последовательных приборов. Требование $i \in N$ поступает в систему в момент времени $d_i \geq 0$. Для всех требований множества N маршрут обслуживания (порядок прохождения приборов) заранее не задан и может быть различным для разных требований. Длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$, предполагаются заданными. Запись $t_{iL} = 0$ означает, что требование i прибором L не обслуживается. В любой момент времени каждое требование обслуживается не более чем одним прибором и каждый прибор обслуживает не более одного требования. Такие обслуживающие системы называются *системами с нефиксированными маршрутами*.

Для доказательства NP -трудности задач построения оптимальных расписаний в данной главе используются следующие эталонные задачи.

Задача о разбиении. Дано множество $N^0 = \{1, 2, \dots, n_0\}$, каждому элементу i которого сопоставлено натуральное число e_i и $\sum_{i \in N^0} e_i = 2E$. Существует ли разбиение множества N^0 на два подмножества N_1^0 и N_2^0 таких, что $\sum_{i \in N_1^0} e_i = \sum_{i \in N_2^0} e_i = E$?

Задача о 3-разбиении. Дано множество $N^0 = \{1, 2, \dots, 3n_0\}$ и натуральное число E . Каждому элементу $i \in N^0$ сопоставлено натуральное число e_i такое, что $E/4 < e_i < E/2$ и $\sum_{i \in N^0} e_i = n_0 E$. Существует ли разбиение множества N^0 на n_0 трехэлементных подмножеств N_j^0 таких, что $\sum_{i \in N_j^0} e_i = E$, $j = 1, \dots, n_0$?

Задача о 3-раскраске 4-регулярного графа. Дан неориентированный граф $G = (X, U)$ без петель, степень каждой вершины которого равна четырем. Можно ли раскрасить вершины графа G в три цвета таким образом, что никакие две смежные вершины не окрашиваются в один цвет?

Задача о разбиении NP -полна [133], задача о 3-разбиении NP -полна в сильном смысле [531]. NP -полнота о 3-раскраске 4-регулярного графа доказана в [550].

В этой главе используются следующие обозначения. Пусть $(\alpha_1, \alpha_2, \dots, \alpha_r)$ — вектор с действительными компонентами и $R' \subseteq R = \{1, 2, \dots$

$\dots, r\}$, тогда $\alpha(R') = \sum_{i \in R'} \alpha_i$. Аналогично, если $\|\alpha_{ij}\|$ — матрица размерности $r \times q$ и $Q' \subseteq Q = \{1, 2, \dots, q\}$, то

$$\alpha_j(R') = \sum_{i \in R'} \alpha_{ij}, \quad 1 \leq j \leq q \quad \text{и} \quad \alpha_i(Q') = \sum_{j \in Q'} \alpha_{ij}, \quad 1 \leq i \leq r.$$

Если π — некоторая перестановка элементов конечного множества X , то через $\{\pi\}$ будем обозначать множество X , т.е. $\{\pi\} = X$.

§ 1. Общее время обслуживания. Два прибора

В этом параграфе рассматривается задача построения оптимального по быстродействию расписания обслуживания n требований M приборами в предположении, что все требования поступают в очередь на обслуживание в момент времени $d = 0$, каждое требование может обслуживаться приборами в произвольном порядке, прерывания процесса обслуживания каждого требования каждым прибором запрещены. Обсуждаются также некоторые обобщения указанной ситуации. Особое внимание уделено случаю, когда $M = 2$.

1.1. Имеется множество $N = \{1, 2, \dots, n\}$ требований и множество $\mathcal{M} = \{1, 2, \dots, M\}$ обслуживающих приборов. Известны длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L \in \mathcal{M}$. Если $t_{iL} > 0$, то требование i должно быть обслужено прибором L . Если $t_{iL} = 0$, то требование i прибором L не обслуживается. Порядок прохождения приборов заранее не задан и может быть различным для разных требований. Каждый прибор может обслуживать требования в произвольной последовательности, но одновременно не более одного требования. Каждое требование одновременно обслуживается не более чем одним прибором. Процесс обслуживания каждого требования $i \in N$ каждым прибором $L \in \mathcal{M}$ предполагается непрерывным, т.е. если обслуживание требования i прибором L начинается в момент времени t_{iL}^0 , то оно протекает без прерываний и завершается в момент $t_{iL} = t_{iL}^0 + t_{iL}$.

В рассматриваемом случае расписание s однозначно определяется заданием матрицы $\|t_{iL}^0\|$ моментов начала или матрицы $\|\bar{t}_{iL}\|$ моментов завершения обслуживания требований приборами. Общее время обслуживания при этом равно $\bar{t}_{\max}(s) = \max \{\bar{t}_{iL} \mid i \in N, L \in \mathcal{M}\}$. Необходимо построить расписание s , при котором величина $\bar{t}_{\max}(s)$ минимальна. Это расписание будем называть оптимальным (по быстродействию) и обозначать через s^* .

Отметим, что задача построения оптимального расписания обладает определенной симметрией: *приборы и требования можно поменять ролями без изменения существа задачи.*

Нетрудно видеть, что для любого расписания s общее время обслуживания

$$\bar{t}_{\max}(s) \geq \max \left\{ \max_{i \in N} t_i(\mathcal{M}), \max_{L \in \mathcal{M}} t_L(N) \right\}. \quad (1.1)$$

Ниже показано, что при $M = 2$ (или, что то же самое, при $n = 2$) для построения оптимального расписания требуется не более $O(n)$ (соответственно $O(M)$) операций. Описан полиномиальный алгоритм построения оптимального расписания в случае, когда часть требований имеет фиксиро-

ванные маршруты обслуживания и $M = 2$. В заключение параграфа доказана NP -трудность задачи построения оптимального расписания, если $M \geq 2$ и все требования обслуживаются без задержек.

1.2. Покажем, что при $M = 2$ оптимальное по быстродействию расписание может быть построено в результате выполнения не более чем $O(n)$ операций.

Пусть система содержит два прибора A и B и известны длительности a_i и b_i обслуживания требования $i \in N = \{1, 2, \dots, n\}$ приборами A и B соответственно.

В этом случае из (1.1) следует, что для любого расписания справедливо соотношение

$$\bar{t}_{\max}(s) \geq \max \{a(N), b(N), \max_{i \in N} \{a_i + b_i\}\}, \quad (1.2)$$

где $a(N) = \sum_{i \in N} a_i$, $b(N) = \sum_{i \in N} b_i$.

Покажем, что существует такое расписание s^* , что в соотношении (1.2) при $s = s^*$ имеет место равенство, и опишем способ его построения. Расписание s^* будем искать в классе таких расписаний, что для каждого прибора имеется не более двух временных интервалов, в каждом из которых этот прибор функционирует без простоев, обслуживая требования непрерывно одно за другим. Расписания этого класса однозначно определяются заданием двух наборов вида $(L; \pi_1; \pi_2; t_0; w_L)$, где L — наименование прибора, t_0 — момент начала непрерывного обслуживания требований множества $\{\pi_1\}$ в последовательности π_1 , w_L — длительность простоя прибора L после завершения обслуживания требований множества $\{\pi_1\}$ и до начала непрерывного обслуживания требований множества $\{\pi_2\}$ в последовательности π_2 .

Множество N требований разобьем на два подмножества $N_1 = \{i \in N \mid a_i \leq b_i\}$ и $N_2 = N \setminus N_1$. Предположим, что $N_1 \neq \emptyset$ и $N_2 \neq \emptyset$. Среди требований множества N_1 выберем такое требование k , что $b_k \geq \max \{a_i \mid i \in N_1\}$, а среди требований множества N_2 выберем требование r , для которого $a_r \geq \max \{b_i \mid i \in N_2\}$.

Рассмотрим пару наборов (рис. 1.1, а)

$$(A; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); 0; w_A); \quad (1.3)$$

$$(B; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); a_k, w_B),$$

где $\pi_{N'}$ — произвольная перестановка элементов множества $N' \subseteq N$, а в качестве w_A и w_B выбраны любые неотрицательные числа, удовлетворяющие условию

$$a(N) - a_k + w_A = b(N) - b_r + w_B. \quad (1.4)$$

Нетрудно проверить, что эта пара наборов определяет расписание, поскольку требования обслуживаются обоими приборами в одной и той же последовательности, причем к моменту начала обслуживания каждого требования прибором B оно уже обслужено прибором A .

Опишем процесс преобразования этого расписания в искомое оптимальное расписание.

1) Предположим, что $a(N) - a_k \geq b(N) - b_r$. Тогда из (1.4) следует, что $w_A \leq w_B$.

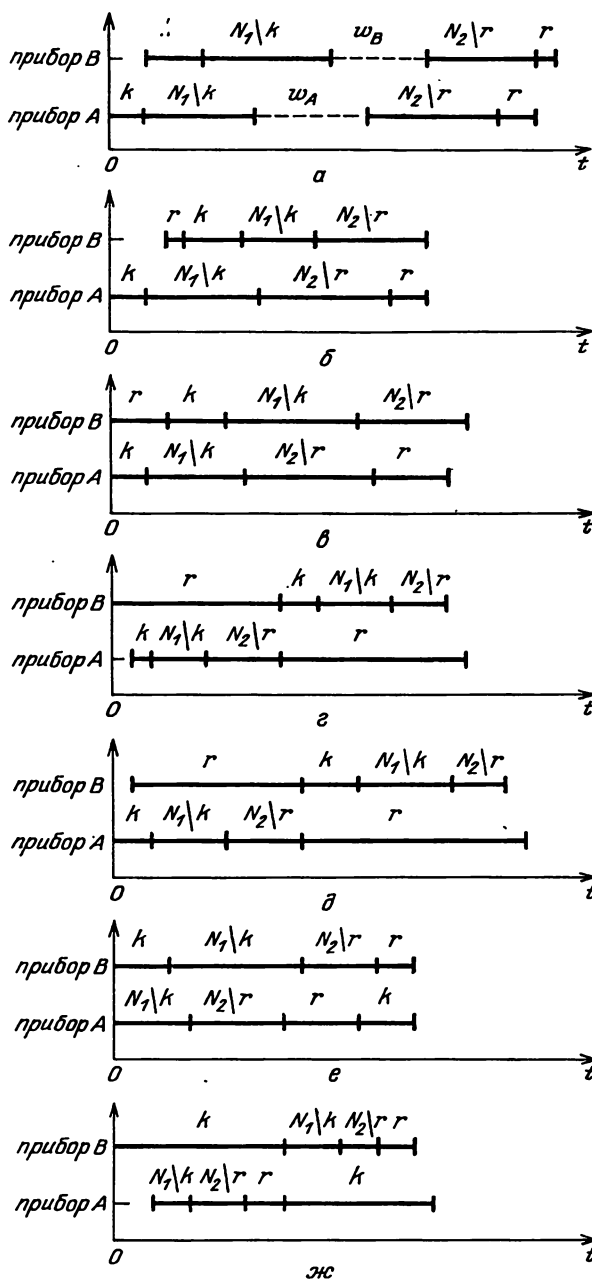


Рис. 1.1

Введем в рассмотрение пару наборов

$$(A; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); 0; 0); \quad (1.5)$$

$$(B; (r, k, \pi_{N_1 \setminus k}); \pi_{N_2 \setminus r}; \max \{a(N) - b(N), 0\}; 0).$$

а) Если имеет место неравенство

$$\max \{a(N) - b(N), 0\} + b_r \leq a(N) - a_r,$$

то (1.5) определяет расписание с общим временем обслуживания, равным $\max \{a(N), b(N)\}$ (см. рис. 1.1, б, в).

б) Пусть выполняется неравенство

$$\max \{a(N) - b(N), 0\} + b_r > a(N) - a_r. \quad (1.6)$$

При этом, очевидно, в некоторый момент времени требование r обслуживается обоими приборами одновременно, т.е. (1.5) не определяет расписания.

Заменим (1.5) парой наборов

$$(A; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); a_r + b_r - a(N); 0); \quad (1.7)$$

$$(B; (r, k, \pi_{N_1 \setminus k}); \pi_{N_2 \setminus r}; 0; 0).$$

Если $a(N) < b(N)$, то неравенство (1.6) равносильно неравенству $a_r + b_r > a(N)$ и (1.7) определяет расписание с общим временем обслуживания, равным $\max \{a_r + b_r, b(N)\}$ (см. рис. 1.1, г).

Пусть $a(N) \geq b(N)$. Тогда неравенство (1.6) равносильно неравенству $a_r + b_r > b(N)$.

Если $a_r + b_r > a(N)$, то (1.7) определяет расписание с общим временем обслуживания, равным $a_r + b_r$ (см. рис. 1.1, г).

Если $a_r + b_r \leq a(N)$, то нетрудно убедиться, что пара наборов

$$(A; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); 0; 0);$$

$$(B; (r, k, \pi_{N_1 \setminus k}); \pi_{N_2 \setminus r}, a(N) - a_r - b_r; 0)$$

определяет расписание с общим временем обслуживания, равным $a(N)$ (см. рис. 1.1, д).

2) Предположим, что $a(N) - a_k < b(N) - b_r$. Тогда из (1.4) следует, что $w_A > w_B$.

Введем в рассмотрение пару наборов

$$(A; \pi_{N_1 \setminus k}; (\pi_{N_2 \setminus r}, r, k); 0; 0); \quad (1.8)$$

$$(B; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); 0; 0).$$

Если $a_k + b_k \leq a(N)$, то (1.8) определяет расписание с общим временем обслуживания, равным $\max \{a(N), b(N)\}$ (см. рис. 1.1, е).

Если $a_k + b_k > a(N)$, то пара наборов (1.8) не определяет расписания, поскольку в некоторый момент времени требование k одновременно обслуживается двумя приборами.

Заменяем (1.8) парой наборов

$$(A; \pi_{N_1 \setminus k}; (\pi_{N_2 \setminus r}, r, k); a_k + b_k - a(N); 0);$$

$$(B; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r), 0; 0).$$

Нетрудно убедиться, что эта пара наборов определяет расписание с общим временем обслуживания, равным $\max\{a_k + b_k, b(N)\}$ (см. рис. 1.1, ж).

Таким образом, во всех рассмотренных случаях может быть построено расписание s^* такое, что в соотношении (1.2) при $s = s^*$ имеет место знак равенства.

Можно показать, что и в случаях, когда $N_1 = \phi$ или $N_2 = \phi$, также может быть построено расписание s^* , обладающее указанным свойством.

Если $N_1 = \phi$, то искомого оптимального расписания определяется парой наборов вида

$$(A; \pi_{N_2 \setminus r}; r; \max\{a_r + b_r - a(N), 0\}; 0);$$

$$(B; r; \pi_{N_2 \setminus r}; \max\{\min\{a(N) - b(N), a(N) - a_r - b_r\}, 0\}; 0).$$

Если $N_2 = \phi$, — то парой наборов вида

$$(A; \pi_{N_1 \setminus k}; k; \max\{a_k + b_k - a(N), 0\}; 0);$$

$$(B; k; \pi_{N_1 \setminus k}; 0; 0).$$

Если $N_1 \neq \phi$ и $N_2 \neq \phi$, то из приведенных выше рассуждений следует, что искомого оптимального расписания определяется наборами

$$(A; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); \max\{a_r + b_r - a(N), 0\}; 0);$$

$$(B; (r, k, \pi_{N_1 \setminus k}); \pi_{N_2 \setminus r}; \max\{\min\{a(N) - b(N), a(N) - a_r - b_r\}, 0\}; 0),$$

если $a(N) - a_k \geq b(N) - b_r$, и

$$(A; \pi_{N_1 \setminus k}; (\pi_{N_2 \setminus r}, r, k); \max\{a_k + b_k - a(N), 0\}; 0);$$

$$(B; (k, \pi_{N_1 \setminus k}); (\pi_{N_2 \setminus r}, r); 0; 0),$$

если $a(N) - a_k < b(N) - b_r$.

Отметим, что формирование множеств N_1 и (или) N_2 , выбор требуемых $k \in N_1$ и (или) $r \in N_2$, вычисление $a(N)$ и $b(N)$ требуют выполнения не более чем $O(n)$ элементарных операций. Трудоемкость последующего построения искомого пары наборов не зависит от n .

Таким образом, при $M = 2$ оптимальное по быстродействию расписание может быть построено в результате выполнения не более чем $O(n)$ операций. В силу симметрии задачи описанный алгоритм может быть использован и в случае, когда $M > 2$, но $n = 2$.

1.3. Пусть $m \neq r$. Имеются две разные книги A и B и шесть читателей, желающих прочесть эти книги. Длительности a_i и b_i (количество дней, затрачиваемых читателем на прочтение каждой из книг A и B) приведены в табл. 1.1.

Предполагается, что каждый читатель может одновременно читать только одну книгу и, раз начав ее читать, читает непрерывно до тех пор, пока не

Таблица 1.1

i	1	2	3	4	5	6
a_i	2	6	2	3	3	6
b_i	4	3	1	5	4	4

прочтет всю книгу. Каждая книга в каждый момент времени может находиться не более чем у одного читателя. Необходимо, чтобы все читатели прочли все книги в кратчайший срок.

В рассматриваемом случае $N = \{1, 2, 3, 4, 5, 6\}$, $N_1 = \{1, 4, 5\}$, $N_2 = \{2, 3, 6\}$, $k \in \{1, 4, 5\}$, $r \in \{2, 6\}$, для определенности положим $k = 4$, $r = 2$. Имеем $a(N) = 22$, $b(N) = 21$, $a(N) - a_k = 19 > b(N) - b_r = 18$.

Соответствующая пара наборов, определяющих одно из искоемых оптимальных расписаний s^* , имеет вид

(A ; (4, 1, 5); (3, 6, 2); 0; 0);

(B ; (2, 4, 1, 5); (3, 6); 1; 0).

Расписание s^* представлено на рис. 1.2.

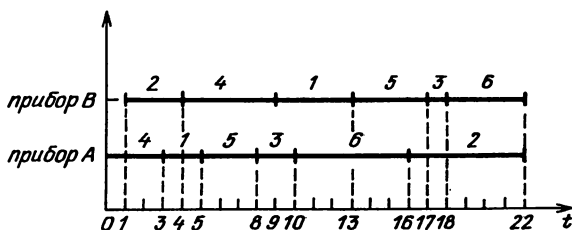


Рис. 1.2

1.4. Рассмотрим следующее обобщение ситуации, описанной в п. 1.2.

В обслуживающую систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительность обслуживания требования $i \in N$ прибором A равна $a_i \geq 0$ единицам времени и прибором B равна $b_i \geq 0$ единицам времени. Запись $a_i = 0$ или $b_i = 0$ означает, что требование не обслуживается прибором A или B соответственно.

Множество N требований состоит из трех попарно непересекающихся подмножеств N_{AB} , N_{BA} и N_O . Требования множества N_{AB} обслуживаются сначала прибором A , затем прибором B (маршрут обслуживания (A, B)); маршрутом обслуживания требований множества N_{BA} является (B, A) ; требования множества N_O могут обслуживаться в соответствии с любым из маршрутов (A, B) или (B, A) , причем маршруты заранее не заданы и могут быть различными для разных требований. Ко множеству N_O относятся также все требования, для которых либо $a_i = 0$, либо $b_i = 0$.

В любой момент времени каждый прибор обслуживает не более одного требования и каждое требование обслуживается не более чем одним прибором.

ром. Прерывания процесса обслуживания каждого требования каждым прибором запрещены.

Необходимо построить оптимальное (по быстродействию) расписание s^* обслуживания требований множества N . Если $N = N_O$, имеем ситуацию, рассмотренную в п. 1.2, если $N = N_{AB}$ (или $N = N_{BA}$) — ситуацию, рассмотренную в п. 1.3 гл. 1, и если $N_O = \phi$ — ситуацию, рассмотренную в п. 4.2 гл. 2.

Ниже описан алгоритм построения расписания s^* в случае, когда все три подмножества N_{AB} , N_{BA} и N_O могут быть не пустыми. Трудоемкость построения s^* не превосходит $O(n + \tilde{n} \log \tilde{n})$ операций, где $\tilde{n} = \max \{ |N_{AB}|, |N_{BA}| \}$.

Введем некоторые обозначения. Для множества $R \subseteq N$ положим $R^a = \{i \in R \mid a_i \leq b_i\}$ и $R^b = \{i \in R \mid b_i < a_i\}$. Через $\pi(R)$ обозначим произвольную перестановку элементов множества R .

Оптимальное расписание s^* будем искать в классе расписаний, которые однозначно определяются парой наборов вида $(L; H_1; H_2, \dots, H_{p_L})$, $p_L \geq 1, L \in \{A, B\}$. Элемент этого набора имеет вид $H_j = [t_L^0(j), \pi_j(N_j)]$, где $t_L^0(j)$ — момент времени, начиная с которого прибор L , функционируя без простоев, обслуживает без прерываний требования множества $N_j \subseteq N$ в последовательности $\pi_j(N_j)$.

Если в системе обслуживаются только требования множества N_{AB} , то оптимальное (по быстродействию) расписание будем обозначать через s_{AB}^* . Напомним способ построения такого расписания (см. п. 1.3 гл. 1). Требования множества N_{AB}^a упорядочим по неубыванию значений a_i , вслед за ними упорядочим требования множества N_{AB}^b по невозрастанию значений b_i . Полученную последовательность обозначим через $\varphi(N_{AB}) = (i_1, i_2, \dots, i_q)$, $q = |N_{AB}|$. Расписание s_{AB}^* определяется парой наборов

$$(A; [0, \varphi(N_{AB})]);$$

$$(B; [t_B^0, \varphi(N_{AB})]),$$

где $t_B^0 = \max \{a_{i_1}, \max \{ \sum_{j=1}^u a_{i_j} - \sum_{j=1}^{u-1} b_{i_j} \mid u = 2, \dots, n \} \}$. Трудоемкость построения расписания s_{AB}^* не превосходит $O(q \log q)$ операций.

Если в системе обслуживаются только требования множества N_O , то можно описать несколько способов построения расписаний, которые, вообще говоря, не являются оптимальными, но могут быть преобразованы в оптимальные в соответствии с п. 1.2.

Действительно, если $N_O^a \neq \phi$, выберем требование $k \in N_O^a$ такое, что $b_k \geq a_i$, $i \in N_O^a$, иначе положим $\{k\} = \phi$. Аналогично, если $N_O^b \neq \phi$, выберем требование $r \in N_O^b$ такое, что $a_r \geq b_i$, $i \in N_O^b$, иначе положим $\{r\} = \phi$. Построим последовательности $\psi(N_O) = (k, \pi(N_O^a \setminus \{k\}), \pi(N_O^b \setminus \{r\}), r)$ и $\psi'(N_O) = (r, \pi(N_O^b \setminus \{r\}), \pi(N_O^a \setminus \{k\}), k)$. Для произвольного требования $m \in N_O$ через $\psi(N_O \setminus \{m\})$ будем обозначать последовательность требований множества $N_O \setminus \{m\}$, расположенных в том же порядке, что и в последовательности $\psi(N_O)$.

Если имеет место неравенство $a(N_O \setminus \{k\}) \leq b(N_O \setminus \{r\})$, то можно построить расписание s_k обслуживания требований множества N_O , которое определяется парой наборов

$$(A; [0, \psi(N_O)]);$$

$$(B; [a_k, \psi(N_O)]),$$

и расписание $\tilde{s}_k$, которое определяется парой наборов

$$(A; [b(N_O) - a(N_O \setminus \{k\}), \psi'(N_O)]);$$

$$(B; [0, \psi'(N_O)]).$$

Нетрудно проверить, что $\bar{t}_{kA}(s_k) = t_{kB}^0(s_k)$ и $\bar{t}_{iA}(s_k) \leq t_{iB}^0(s_k)$, а также что $\bar{t}_{kB}(\tilde{s}_k) = t_{kA}^0(\tilde{s}_k)$ и $\bar{t}_{iB}(\tilde{s}_k) \leq t_{iA}^0(\tilde{s}_k)$, $i \in N_O \setminus \{k\}$. Расписания s_k и $\tilde{s}_k$ представлены на рис. 1.3 а и 1.3 б соответственно.

Если имеет место неравенство $a(N_O \setminus \{k\}) > b(N_O \setminus \{r\})$, то можно построить расписание s_r обслуживания требований множества N_O , кото-

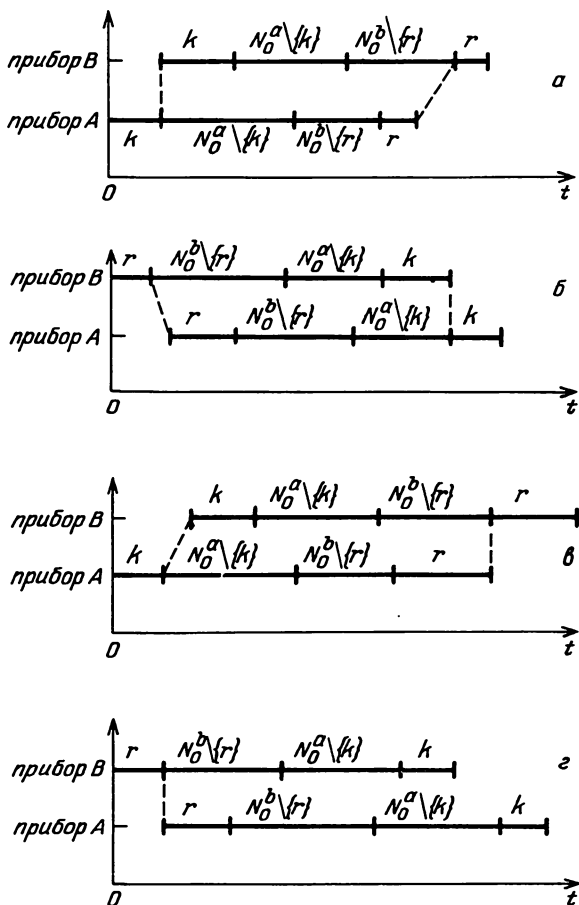


Рис. 1.3

рое определяется парой наборов

$$(A; [0, \psi(N_O)]);$$

$$(B; [a(N_O) - b(N_O \setminus \{r\}), \psi(N_O)]),$$

и расписание $\tilde{s}_r$, которое определяется парой наборов

$$(A; [b_r, \psi'(N_O)]);$$

$$(B; [0, \psi'(N_O)]).$$

Нетрудно проверить, что $\bar{t}_{rA}(s_r) = t_{rB}^0(s_r)$ и $\bar{t}_{iA}(s_r) \leq t_{iB}^0(s_r)$, $i \in N_O \setminus \{r\}$. Аналогично $\bar{t}_{rB}(\tilde{s}_r) = t_{rA}^0(\tilde{s}_r)$ и $\bar{t}_{iB}(\tilde{s}_r) \leq t_{iA}^0(\tilde{s}_r)$, $i \in N_O \setminus \{r\}$. Расписания s_r и $\tilde{s}_r$ представлены на рис. 1.3, в и 1.3, г соответственно.

Трудоемкость построения каждого из расписаний s_k , s_r , $\tilde{s}_k$ и $\tilde{s}_r$ не превосходит $O(|N_O|)$ операций.

1) Оценим величину $\bar{t}_{\max}(s)$ общего времени обслуживания требований множества $N = N_{AB} \cup N_{BA} \cup N_O$ при произвольном расписании s .

Пусть s_{AB}^* и s_{BA}^* — оптимальные расписания обслуживания только требований множества N_{AB} и только требований множества N_{BA} соответственно, а $t_{\max}(s_{AB}^*)$ и $t_{\max}(s_{BA}^*)$ — соответствующие этим расписаниям значения общего времени обслуживания.

Очевидно, имеет место соотношение $\bar{t}_{\max}(s) \geq T_1 = \max\{T, \bar{t}_{\max}(s_{AB}^*), \bar{t}_{\max}(s_{BA}^*)\}$, где $T = \max\{a(N), b(N)\}$. Кроме того, если $a_m + b_m = \max\{a_i + b_i | i \in N_O\} > T_1$, то $\bar{t}_{\max}(s) \geq a_m + b_m$, $m \in N_O$.

Покажем, что расписания s (без прерываний), при котором $\bar{t}_{\max}(s) = a_m + b_m > T_1$ не существует.

Действительно, при таком расписании в любой момент времени во временном интервале $(0, \bar{t}_{\max}(s)]$ требование m должно обслуживаться одним из приборов. Следовательно, за время обслуживания требования m прибором A (или B) все остальные требования должны быть обслужены прибором B (соответственно прибором A). При этом, однако, нарушится маршрут обслуживания всех требований одного из множеств N_{AB} или N_{BA} , что недопустимо.

Оценим величину $\bar{t}_{\max}(s)$, если $a_m + b_m > T_1$. Рассмотрим класс расписаний, при которых требованию m назначается маршрут обслуживания (A, B) . Построим расписание s_1 из этого класса, определяемое парой наборов

$$(A; [0, (\pi(N_{AB}^a), m, \pi(N_{AB}^b \cup N_{BA} \cup N_O \setminus \{m\}))]);$$

$$(B; [a(N_{AB}^a \cup \{m\}) - b(N_{AB}^a \cup N_{BA} \cup N_O \setminus \{m\}),$$

$$\pi(N_{AB}^a \cup N_{BA} \cup N_O \setminus \{m\}), m, \pi(N_{AB}^b)]).$$

$$\text{Тогда } \bar{t}_{\max}(s_1) = a_m + b_m + a(N_{AB}^a) + b(N_{AB}^b).$$

Это расписание будет оптимальным в рассматриваемом классе, поскольку

а) ни одно из требований множества N_{AB} не может быть обслужено обоими приборами в интервале обслуживания требования m ;

б) значение $a(N_{AB}^a) + b(N_{AB}^b) \leq a(N_{AB}'') + b(N_{AB}'')$ для любого разбиения множества N_{AB} на подмножества N_{AB}' и N_{AB}'' ;

в) обслуживание требований множества $N_{BA} \cup N_O \setminus \{m\}$ вне интервала обслуживания требования m не уменьшит общего времени обслуживания.

Рассуждая аналогичным образом, можно показать, что в классе расписаний, при которых требованию m назначается маршрут обслуживания (B, A) , оптимальным будет расписание s_2 такое, что $\bar{t}_{\max}(s_2) = a_m + b_m + a(N_{BA}^a) + b(N_{BA}^b)$.

Таким образом, для любого расписания s (без прерываний) имеем

$$\bar{t}_{\max}(s) \geq \max\{T, \bar{t}_{\max}(s_{AB}^*), \bar{t}_{\max}(s_{BA}^*),$$

$$\max\{a_i + b_i | i \in N_O\} + \min\{a(N_{AB}^a) + b(N_{AB}^b), a(N_{BA}^a) + b(N_{BA}^b)\}\} \}. \quad (1.9)$$

2) Опишем процесс построения расписания s , при котором в (1.9) имеет место знак равенства. Это расписание, очевидно, и является искомым оптимальным расписанием s^* .

Рассмотрим следующие пары наборов, каждая из которых при определенных условиях задает расписание s^* .

- I. $(A; [0, (\varphi(N_{AB}), \pi(N_{BA} \cup N_O))]);$
 $(B; [0, \pi(N_{BA} \cup N_O)]);$
 $[\max\{b(N_{BA} \cup N_O),$
 $\bar{t}_{\max}(s_{AB}^*) - b(N_{AB})\}, \varphi(N_{AB})]);$
- II. $(A; [0, (\pi(N_{AB}), \pi(N_{BA}), \psi'(N_O))]);$
 $(B; [0, (\pi(N_{BA}), \psi'(N_O), \pi(N_{AB}))]);$
- III. $(A; [0, (\psi(N_O \setminus \{k\}), \pi(N_{AB}), \pi(N_{BA}), k)]);$
 $(B; [0, (\pi(N_{BA}), k, \psi(N_O \setminus \{k\}), \pi(N_{AB}))]);$
- IV. $(A; [0, (\pi(N_{AB}), \pi(N_{BA}), \psi(N_O \setminus \{r\}), r)]);$
 $(B; [0, (\pi(N_{BA}), r, \pi(N_{AB}))]);$
 $[T - b(N_O \setminus \{r\}), \psi(N_O \setminus \{r\})];$
- V. $(A; [0, (\pi(N_{AB}), \pi(N_{BA} \cup N_O \setminus \{m\}))]); [b(N_{BA} \cup \{m\}), m]);$
 $(B; [0, (\pi(N_{BA}), m, \pi(N_{AB} \cup N_O \setminus \{m\}))]);$
- VI. $(A; [0, (\pi(N_{AB}), \pi(N_{BA}^b \cup N_O \setminus \{m\}))]);$
 $[\max\{a(N_{AB} \cup N_{BA}^b \cup N_O \setminus \{m\}),$
 $b(N_{BA}^b \cup \{m\})\}, (m, \pi(N_{BA}^a))]);$
 $(B; [0, (\pi(N_{BA}^b), m, \pi(N_{BA}^a), \pi(N_{AB} \cup N_O \setminus \{m\}))]);$
- VII. $(A; [0, (m, \pi(N_{AB}))]); [\max\{a(N_{AB} \cup \{m\}),$
 $b(N_{BA} \cup N_O \setminus \{m\})\}, \pi(N_{BA} \cup N_O \setminus \{m\})];$
 $(B; [0, (\pi(N_{BA}), \pi(N_O \setminus \{m\}), m, \pi(N_{AB}))]);$

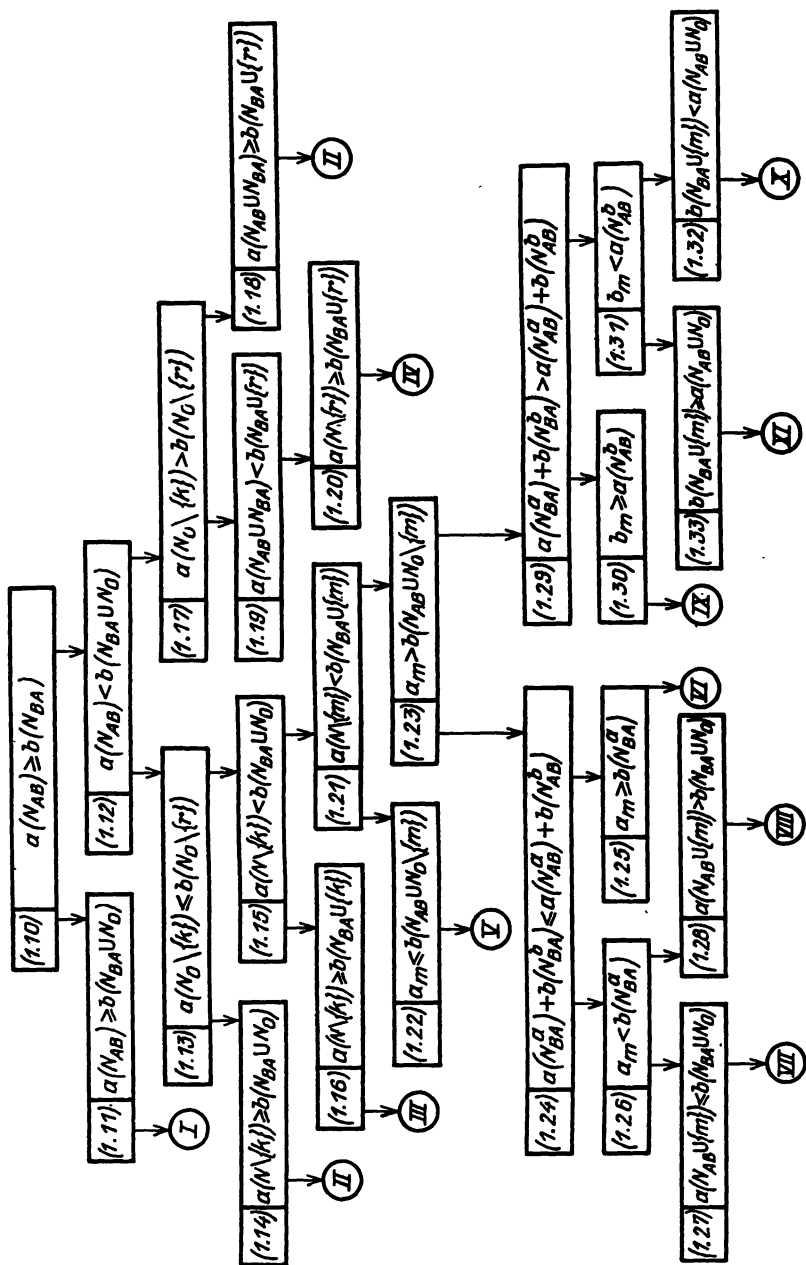


Рис. 1.4

- VIII. $(A; [0, (\pi(N_{AB}), \pi(N_O), \pi(N_{BA})))$;
 $(B; [0, (\pi(N_O), \pi(N_{BA}), \pi(N_{AB})))$);
- IX. $(A; [0, (\pi(N_{AB}^a), m, \pi(N_{AB}^b), \pi(N_{BA} \cup N_O \setminus \{m\})))$;
 $(B; [0, (\pi(N_{BA} \cup N_O \setminus \{m\}), \pi(N_{AB}^a)))$; $[\max\{a(N_{AB}^a \cup \{m\}),$
 $b(N_{BA} \cup N_{AB}^a \cup N_O \setminus \{m\}), (m, \pi(N_{AB}^b))]\}$);
- X. $(A; [0, (\pi(N_{AB} \cup N_O \setminus \{m\}), m, \pi(N_{BA})))$;
 $(B; [0, (m, \pi(N_{BA}), \pi(N_{AB} \cup N_O \setminus \{m\})))$);
- XI. $(A; [0, (\pi(N_O), \pi(N_{AB}), \pi(N_{BA})))$;
 $(B; [0, (\pi(N_{BA}), \pi(N_O), \pi(N_{AB})))$).

Алгоритм построения расписания s^* заключается в проверке выполнения ряда неравенств и выборе одной из перечисленных пар наборов. Порядок проверки выполнения неравенств указан на рис. 1.4. Цифры в круглых скобках слева от неравенства означают номер неравенства, на который далее в тексте имеются ссылки. Римские цифры в кружках означают номер пары наборов.

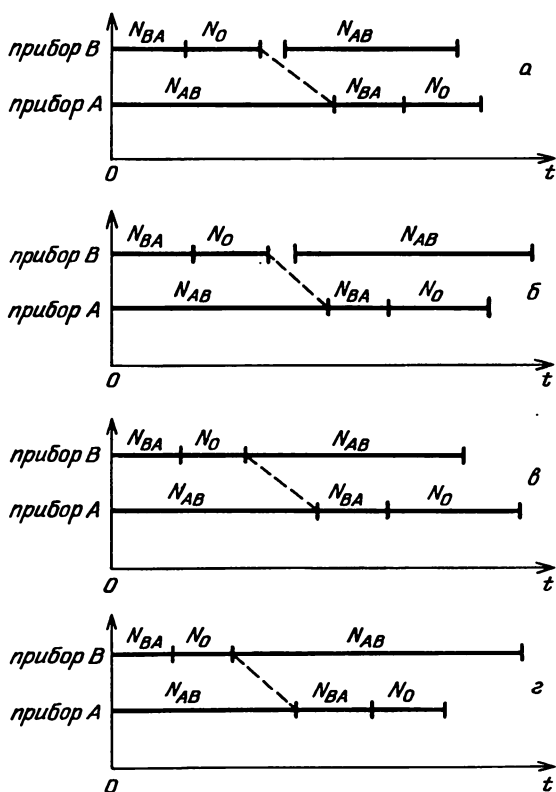


Рис. 1.5

Если неравенство $a(N_{AB}) \geq b(N_{BA})$ не имеет места, то можно поменять местами наименования приборов, построить расписание s^* , воспользовавшись описанной выше схемой, и затем восстановить исходные наименования. При каждом таком переименовании приборов меняются ролями длительности обслуживания a_i и b_i , требования k и r , множества каждой пары N_{AB} и N_{BA} , N_{AB}^a и N_{BA}^b , N_{AB}^b и N_{BA}^a , N_O^a и N_O^b , а также расписания каждой пары s_{AB}^* и s_{BA}^* , s_k и $\tilde{s}_r$, $\tilde{s}_k$ и s_r .

3) Покажем, что описанный алгоритм действительно приводит к построению оптимального расписания.

а) Пусть выполняются неравенства (1.10) и (1.11). Покажем, что оптимальное расписание s^* определяется парой наборов I.

Действительно, в силу (1.11) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA} \cup N_O\} = b(N_{BA} \cup N_O) \leq a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_{BA} \cup N_O\}$. Если $\bar{t}_{\max}(s_{AB}^*) \geq b(N)$ (рис. 1.5, а, б), то $t_{iL}^0(s^*) = t_{iL}^0(s_{AB}^*)$, $\bar{t}_{iL}(s^*) = \bar{t}_{iL}(s_{AB}^*)$, $i \in N_{AB}$, $L \in \{A, B\}$. Если же $b(N) > \bar{t}_{\max}(s_{AB}^*)$ (рис. 1.5, в, з), то $t_{iB}^0(s^*) - \bar{t}_{iA}(s^*) \geq t_{iB}^0(s_{AB}^*) - \bar{t}_{iA}(s_{AB}^*) \geq 0$, $i \in N_{AB}$. Кроме того, $\bar{t}_{\max}(s^*) = \max\{T, \bar{t}_{\max}(s_{AB}^*)\}$.

б) Пусть выполняются неравенства (1.10), (1.12)–(1.14). Покажем, что оптимальное расписание s^* определяется парой наборов II.

Действительно, в силу (1.10) $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\} = b(N_{BA}) \leq a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_{BA}\}$. В силу (1.12) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB}) < b(N_{BA} \cup N_O) = \min\{t_{iB}^0(s^*) | i \in N_{AB}\}$. Из неравенства (1.14) следует, что $t_{kA}^0(s^*) - \bar{t}_{kB}(s^*) = a(N \setminus \{k\}) - b(N_{BA} \cup N_O) \geq 0 = t_{kA}^0(\tilde{s}_k) - \bar{t}_{kB}(\tilde{s}_k)$ и, тем самым, $t_{iA}^0(s^*) - t_{iB}(s^*) \geq t_{iA}^0(\tilde{s}_k) - \bar{t}_{iB}(\tilde{s}_k) \geq 0$, $i \in N_O \setminus \{k\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.6.

в) Пусть выполняются неравенства (1.10), (1.12), (1.13), (1.15) и (1.16). Покажем, что в этом случае оптимальное расписание s^* определяется парой наборов III.

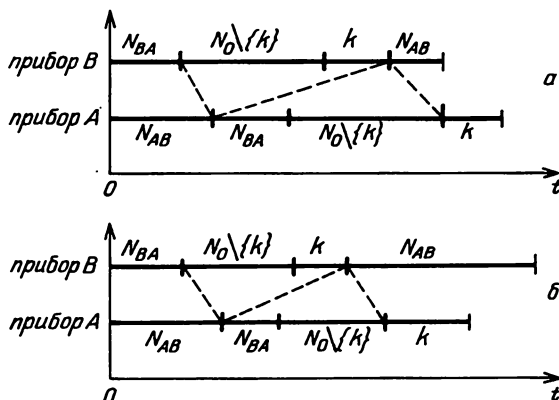


Рис. 1.6

В силу (1.10) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\} = b(N_{BA}) \leq a(N_{AB}) \leq a(N_{AB} \cup N_O \setminus \{k\}) = \min\{t_{iA}^0(s^*) | i \in N_{BA}\}$. Далее, в силу (1.15) $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB} \cup N_O \setminus \{k\}) < b(N_{BA} \cup N_O) = \min\{t_{iB}^0(s^*) | i \in N_{AB}\}$. Из неравенства (1.16) следует, что $t_{kA}^0(s^*) = a(N \setminus \{k\}) \geq b(N_{BA} \cup \{k\}) = \bar{t}_{kB}(s^*)$. Поскольку $b(N_{BA}) \geq 0$, то $t_{iB}^0(s^*) - \bar{t}_{iA}(s^*) \geq t_{iB}^0(s_k) - \bar{t}_{iA}(s_k) \geq 0$, $i \in N_O \setminus \{k\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.7.

г) Пусть выполняются неравенства (1.10), (1.12), (1.17) и (1.18). Покажем, что оптимальное расписание s^* определяется парой наборов II.

В силу (1.10) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\} = b(N_{BA}) \leq a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_{BA}\}$. В силу (1.12) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB}) < b(N_{BA} \cup N_O) = \min\{t_{iB}^0(s^*) | i \in N_{AB}\}$. Из неравенства (1.18) следует, что $\bar{t}_{rB}(s^*) = b(N_{BA} \cup \{r\}) \leq a(N_{AB} \cup N_{BA}) = t_{rA}^0(s^*)$

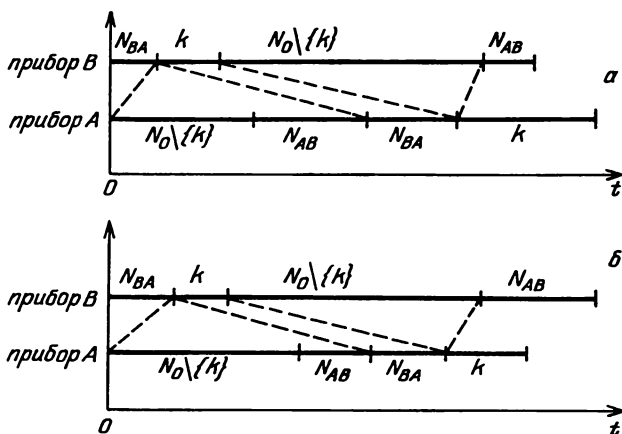


Рис. 1.7

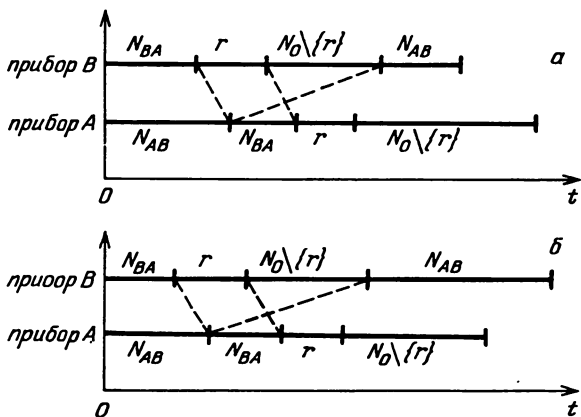


Рис. 1.8

и, тем самым, $t_{iA}^0(s^*) - \bar{t}_{iB}(s^*) \geq t_{iA}^0(\tilde{s}_r) - \bar{t}_{iB}(\tilde{s}_r) \geq 0$, $i \in N_O \setminus \{r\}$.

Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.8.

д) Пусть выполняются неравенства (1.10), (1.12), (1.17), (1.19) и (1.20). Покажем, что в этом случае оптимальное расписание s^* определяется парой наборов IV.

В силу (1.10) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\} \leq \min\{t_{iA}^0(s^*) | i \in N_{BA}\}$. Далее, в силу (1.19) $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB}) \leq a(N_{AB} \cup N_{BA}) < b(N_{BA} \cup \{r\}) = \min\{t_{iB}^0(s^*) | i \in N_{AB}\}$. Из неравенства (1.20) следует, что $\bar{t}_{rB}(s^*) = b(N_{BA} \cup \{r\}) \leq a(N \setminus \{r\}) = t_{rA}^0(s^*)$. Если $a(N) \geq b(N)$ (рис. 1.9, а), то $t_{iB}^0(s^*) - \bar{t}_{iA}(s^*) = t_{iB}^0(s_r) - \bar{t}_{iA}(s_r) \geq 0$, $i \in N_O \setminus \{r\}$. Если $a(N) < b(N)$ (рис. 1.9, б), то $t_{iB}^0(s^*) - \bar{t}_{iA}(s^*) \geq t_{iB}^0(s_r) - \bar{t}_{iA}(s_r) \geq 0$, $i \in N_O \setminus \{r\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$.

е) Положим $m = k$, если имеют место неравенства (1.13) и (1.15), и $m = r$, если имеют место неравенства (1.17) и (1.19). Пусть выполняются неравенства (1.10), (1.12), (1.21) и (1.22). Покажем, что оптимальное расписание s^* определяется парой наборов V.

В силу (1.10) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\} = b(N_{BA}) \leq a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_{BA}\}$. Далее, в силу (1.21) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\} = a(N \setminus \{m\}) < b(N_{BA} \cup \{m\}) = \min\{t_{iB}^0(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\}$. Кроме того, по построению $\bar{t}_{mB}(s^*) = t_{mA}^0(s^*)$. Не-

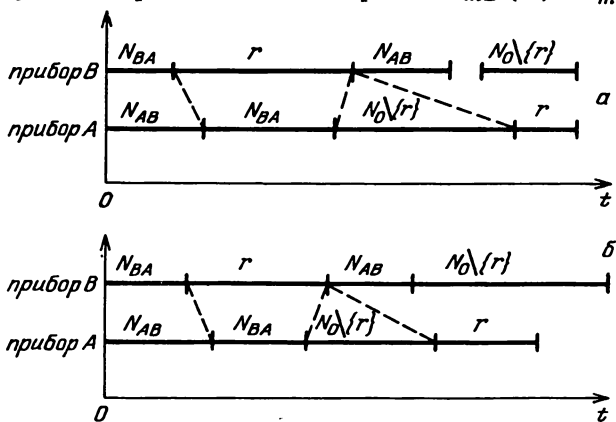


Рис. 1.9

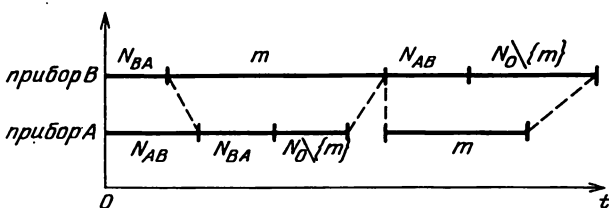


Рис. 1.10

равенство (1.22) означает, что $\bar{t}_{\max}(s^*) = b(N)$. Расписание s^* изображено на рис. 1.10.

ж) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23) – (1.25). Покажем, что оптимальное расписание s^* определяется парой наборов VI.

В силу (1.10) имеем $\max\{\bar{t}_{iB}(s^*) | i \in N_{BA}^b\} = b(N_{BA}^b) \leq a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_{BA}^b\}$. В силу (1.21) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\} \leq a(N_{AB} \cup N_{BA}^b \cup N_O \setminus \{m\}) \leq a(N \setminus \{m\}) < b(N_{BA} \cup \{m\}) = \min\{t_{iB}^0(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\}$. Если $b(N_{BA}^b \cup \{m\}) \geq a(N_{AB} \cup N_{BA}^b \cup N_O \setminus \{m\})$ (рис. 1.11, а, б), то $t_{mA}^0(s^*) = \bar{t}_{mB}(s^*)$ и в силу (1.25) имеем $\min\{t_{iA}^0(s^*) | i \in N_{BA}^a\} = b(N_{BA}^b \cup \{m\}) + a_m \geq b(N_{BA} \cup \{m\}) = \max\{\bar{t}_{iB}(s^*) | i \in N_{BA}^a\}$. Если же $b(N_{BA}^b \cup \{m\}) < a(N_{AB} \cup N_{BA}^b \cup N_O \setminus \{m\})$ (рис. 1.11, в, з), то $t_{mA}^0(s^*) > \bar{t}_{mB}(s^*)$ и в силу (1.25) имеем $\min\{t_{iA}^0(s^*) | i \in N_{BA}^a\} = a(N \setminus N_{BA}^a) = a_m + a(N_{AB} \cup N_{BA}^b \cup N_O \setminus \{m\}) > a_m + b(N_{BA}^b \cup \{m\}) \geq b(N_{BA} \cup \{m\}) =$

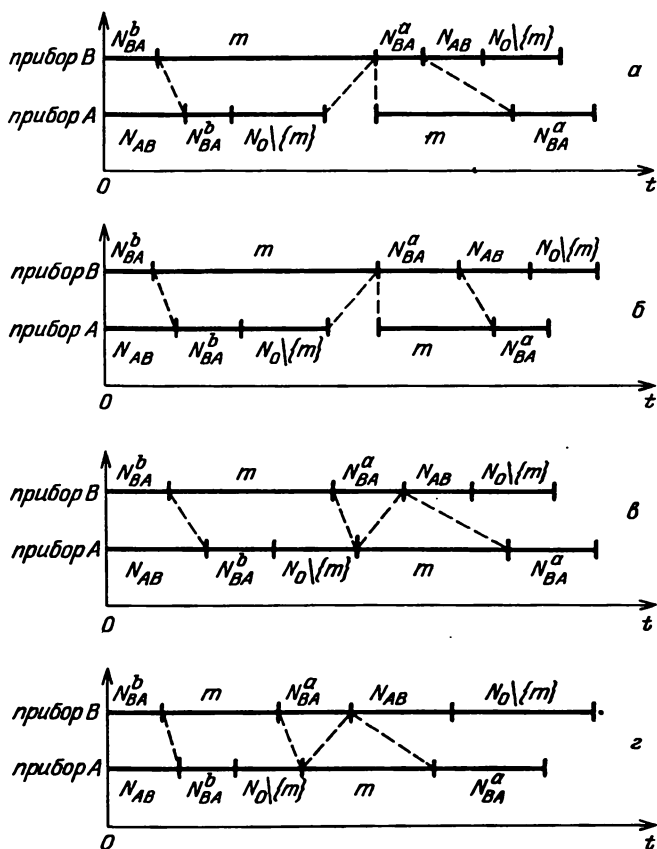


Рис. 1.11

$= \max\{\bar{t}_{iB}(s^*) | i \in N_{BA}^a\}$. Кроме того, $\bar{t}_{\max}(s^*) = \max\{T, a_m + b_m + a(N_{BA}^a) + b(N_{BA}^b)\}$.

з) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23), (1.24), (1.26) и (1.27). Покажем, что в этом случае оптимальное расписание s^* определяется парой наборов VII.

В силу (1.26) имеем $t_{mB}^0(s^*) = b(N_{BA} \cup N_O \setminus \{m\}) \geq b(N_{BA}) \geq b(N_{BA}^a) > a_m = \bar{t}_{mA}(s^*)$. В силу (1.27) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB} \cup \{m\}) \leq b(N_{BA} \cup N_O) = \min\{t_{iB}^0(s^*) | i \in N_{AB}\}$. Если $a(N_{AB} \cup \{m\}) \geq b(N_{BA} \cup N_O \setminus \{m\})$ (рис. 1.12а, б), то $\min\{t_{iA}^0(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\} \geq \max\{\bar{t}_{iB}(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\}$. Если же $a(N_{AB} \cup \{m\}) < b(N_{BA} \cup N_O \setminus \{m\})$ (рис. 1.12, в) то $\min\{t_{iA}^0(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\} = b(N_{BA} \cup N_O \setminus \{m\}) = \max\{\bar{t}_{iB}(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\}$. Отметим, что в последнем случае $\bar{t}_{mB}(s^*) > \max\{\bar{t}_{iA}(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\}$, поскольку, вычитая неравенство (1.10) из неравенства (1.21), имеем $a(N_{BA} \cup N_O \setminus \{m\}) < b_m$. При расписании s^* значение $\bar{t}_{\max}(s^*) = T$.

и) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23), (1.24), (1.26) и (1.28). Покажем, что в этом случае оптимальное расписание s^* определяется парой наборов VIII.

Действительно, вычитая из неравенства (1.28) неравенство $b(N_{BA}) > a_m$, которое следует из (1.26), получаем $b(N_O) < a(N_{AB})$. Это означает, что $\max\{\bar{t}_{iB}(s^*) | i \in N_O\} = b(N_O) < a(N_{AB}) = \min\{t_{iA}^0(s^*) | i \in N_O\}$. Далее, в силу (1.12) имеем $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB}\} = a(N_{AB}) < b(N_{BA} \cup$

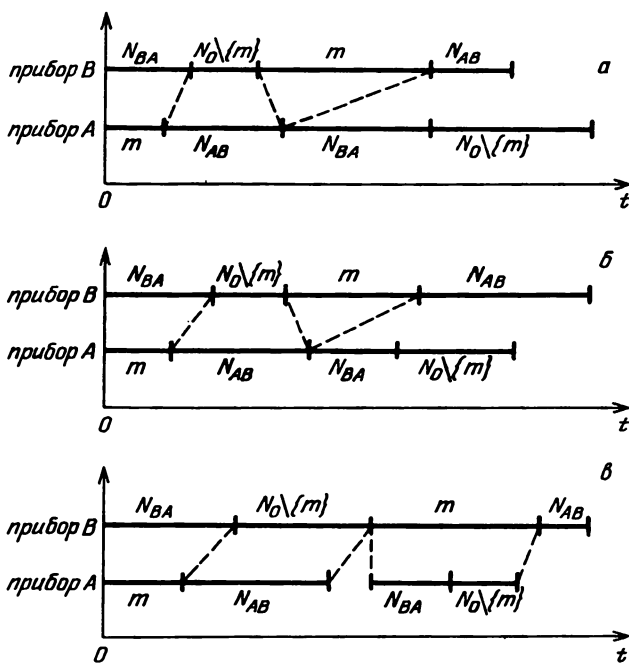


Рис. 1.12

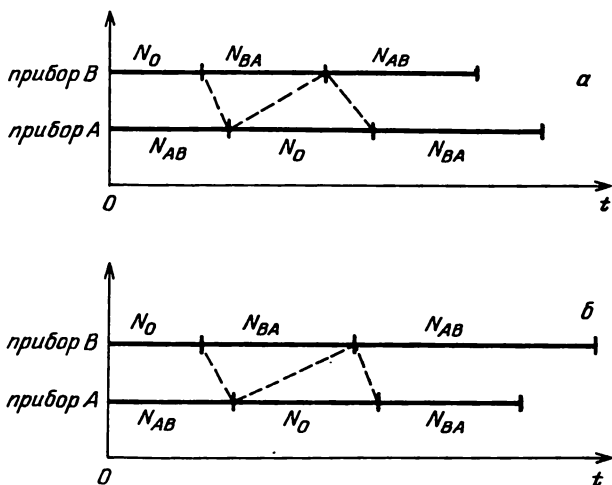


Рис. 1.13

$\cup N_O) = \min \{t_{iB}^0(s^*) | i \in N_{AB}\}$. Из неравенства (1.28) также следует, что $\max \{\bar{t}_{iB}(s^*) | i \in N_{BA}\} = b(N_O \cup N_{BA}) < a(N_{AB} \cup \{m\}) \leq a(N_{AB} \cup N_O) = \min \{t_{iA}^0(s^*) | i \in N_{BA}\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.13.

к) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23), (1.29) и (1.30). Покажем, что оптимальное расписание s^* определяется парой наборов IX.

Действительно, складывая неравенства (1.10) и (1.23), получаем неравенство $a(N_{AB} \cup \{m\}) > b(N \setminus \{m\})$, которое означает, что $\min \{t_{iA}^0(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\} = a(N_{AB} \cup \{m\}) > b(N \setminus \{m\}) = \max \{\bar{t}_{iB}(s^*) | i \in N_{BA} \cup N_O \setminus \{m\}\}$. Поскольку $b(N_{BA}) \geq a(N_{BA}^a) + b(N_{BA}^b)$, то $b(N_{BA}) \geq a(N_{AB}^a)$ и $\min \{t_{iB}^0(s^*) | i \in N_{AB}^a\} = b(N_{BA} \cup N_O \setminus \{m\}) \geq b(N_{BA}) \geq a(N_{AB}^a) = \max \{\bar{t}_{iA}(s^*) | i \in N_{AB}^a\}$. Если $a(N_{AB}^a \cup \{m\}) \geq b(N_{BA} \cup N_{AB}^a \cup N_O \setminus \{m\})$ (рис. 1.14, а, б), то $\bar{t}_{mA}(s^*) = t_{mB}^0(s^*) = \min \{t_{iA}^0(s^*) | i \in N_{AB}^b\}$. В силу (1.30) это означает, что $\min \{t_{iB}^0(s^*) | i \in N_{AB}^b\} = t_{mB}^0(s^*) + b_m \geq t_{mB}^0(s^*) + a(N_{AB}^b) = \max \{\bar{t}_{iA}(s^*) | i \in N_{AB}^b\}$. Если же $a(N_{AB}^a \cup \{m\}) < b(N_{BA} \cup N_{AB}^a \cup N_O \setminus \{m\})$ (рис. 1.14, в, г), то $\bar{t}_{mA}(s^*) < t_{mB}^0(s^*)$, и следовательно, в силу (1.30) значение $\max \{\bar{t}_{iA}(s^*) | i \in N_{AB}^b\} < \bar{t}_{mB}(s^*) = \min \{t_{iB}^0(s^*) | i \in N_{AB}^b\}$. Кроме того, $\bar{t}_{\max}(s^*) = \max \{T, a_m + b_m + a(N_{AB}^a) + b(N_{AB}^b)\}$.

л) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23), (1.29), (1.31) и (1.32). Покажем, что в этом случае оптимальное расписание s^* определяется парой наборов X.

В силу (1.31) имеем $t_{mA}^0(s^*) = a(N_{AB} \cup N_O \setminus \{m\}) \geq a(N_{AB}) > b_m = \bar{t}_{mB}(s^*)$. В силу (1.21) $\max \{\bar{t}_{iA}(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\} = a(N_{AB} \cup N_O \setminus \{m\}) \leq a(N \setminus \{m\}) < b(N_{BA} \cup \{m\}) = \min \{t_{iB}^0(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\}$. Из неравенства (1.32) следует, что $\max \{\bar{t}_{iB}(s^*) | i \in$

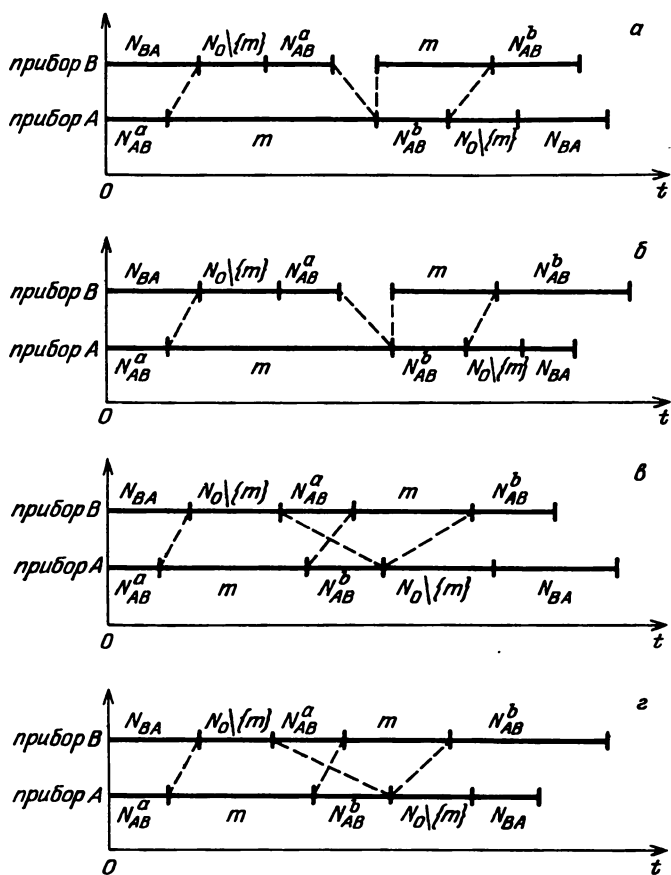


Рис. 1.14

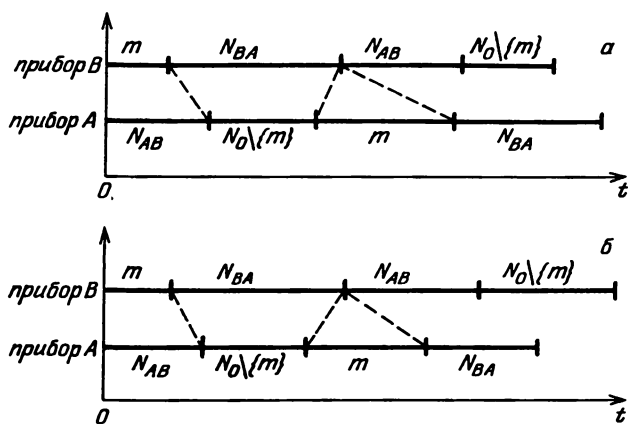


Рис. 1.15

$\in N_{BA}\} = b(N_{BA} \cup \{m\}) < a(N_{AB} \cup N_O) = \min \{t_{iA}^0(s^*) | i \in N_{BA}\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.15.

м) Пусть выполняются неравенства (1.10), (1.12), (1.21), (1.23), (1.29), (1.31) и (1.33). Покажем, что в этом случае расписание s^* определяется парой наборов XI.

Вычитая из неравенства (1.33) неравенство $a(N_{AB}) > b_m$, которое следует из (1.31), получаем $a(N_O) > b(N_{BA})$. Это означает, что $\max \{\bar{t}_{iA}(s^*) | i \in N_O\} = a(N_O) < b(N_{BA}) = \min \{t_{iB}^0(s^*) | i \in N_O\}$. В силу неравенства (1.32) $\min \{t_{iB}^0(s^*) | i \in N_{AB}\} = b(N_{BA} \cup N_O) \geq b(N_{BA} \cup \{m\}) \geq a(N_{AB} \cup N_O) = \max \{\bar{t}_{iA}(s^*) | i \in N_{AB}\}$. Из неравенства (1.10) следует, что $\min \{t_{iA}^0(s^*) | i \in N_{BA}\} = a(N_O \cup N_{AB}) \geq a(N_{AB}) \geq b(N_{BA}) = \max \{t_{iB}^0(s^*) | i \in N_{BA}\}$. Кроме того, $\bar{t}_{\max}(s^*) = T$. Расписание s^* изображено на рис. 1.16.

Расписание s^* , определенное парой наборов I, может быть построено в результате выполнения не более чем $O(n + |N_{AB}| \log |N_{AB}|)$ операций. Расписание s^* , определяемое любой из пар наборов II – XI, может быть построено не более чем за $O(n)$ операций. Следовательно, трудоемкость построения расписания s^* не превосходит $O(n + \tilde{n} \log \tilde{n})$ операций, где $\tilde{n} = \max \{|N_{AB}|, |N_{BA}|\}$.

1.5. В заключение покажем, что задача построения оптимального (по быстродействию) расписания s^* является NP-трудной в сильном смысле при $M \geq 2$, если потребовать, чтобы каждое требование обслуживалось без задержек.

Будем говорить, что процесс обслуживания требования $i \in N = \{1, 2, \dots, n\}$ в системе, состоящей из множества $\mathcal{M} = \{1, 2, \dots, M\}$ приборов, протекает без задержек, если этот процесс начинается в момент времени t_i^0 и завершается в момент времени $\bar{t}_i = t_i^0 + t_i(\mathcal{M})$. По-прежнему предполагается, что процесс обслуживания каждого требования каждым прибором протекает без прерываний.

Рассмотрим задачу распознавания, соответствующую задаче построения оптимального (по быстродействию) расписания без задержек в процессе

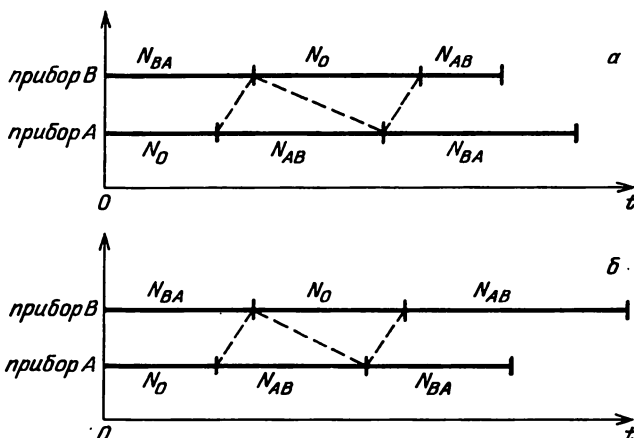


Рис. 1.16

обслуживания требований в случае, когда система состоит из двух приборов A и B . Требуется определить, существует ли расписание s^0 обслуживания (без задержек) множества N требований такое, что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y . Длительности обслуживания требования $i \in N$ приборами A и B соответственно равны a_i и b_i .

Покажем, что к этой задаче сводится задача о равных парах, NP -трудность которой (в сильном смысле) была доказана в п. 5.3 § 5 гл. 2. Напомним формулировку этой задачи. Имеется четыре множества $U = \{u_1, u_2, \dots, u_p\}$, $V = \{v_1, v_2, \dots, v_p\}$, $W = \{w_1, w_2, \dots, w_p\}$ и $X = \{x_1, x_2, \dots, x_p\}$, каждое из которых состоит из натуральных необязательно попарно различных чисел. Требуется определить, существует ли разбиение множества $U \cup V \cup W \cup X$ на четырехэлементные подмножества R_m , $1 \leq m \leq p$, такие, что в каждое из них входит по одному элементу из каждого множества U , V , W и X и для каждого $R_m = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$ имеет место равенство $u_{i_1} + v_{i_2} = w_{i_3} + x_{i_4}$.

Положим $n = 4p$, $N = \{1, 2, \dots, n\}$, $Z = 8 \max_{1 \leq i \leq p} \{u_i, v_i, w_i, x_i\}$. Множество N требований разобьем на четыре группы: U -требования, обозначаемые U_i , $1 \leq i \leq p$; V -требования, обозначаемые V_i , $1 \leq i \leq p$; W -требования, обозначаемые W_i , $1 \leq i \leq p$; X -требования, обозначаемые X_i , $1 \leq i \leq p$. Положим

$$a_{U_i} = 1, b_{U_i} = 4u_i, i = 1, \dots, p;$$

$$a_{V_i} = 1, b_{V_i} = 4v_i + 3Z, i = 1, \dots, p;$$

$$a_{W_i} = 4w_i + Z, b_{W_i} = 1, i = 1, \dots, p;$$

$$a_{X_i} = 4x_i + 2Z, b_{X_i} = 1, i = 1, \dots, p.$$

Покажем, что в построенной задаче расписание s^0 такое, что $\bar{t}_{\max}(s^0) \leq y = 2p + 3pZ + 4 \sum_{i=1}^p (u_i + v_i)$, существует тогда и только тогда, когда имеет место решение задачи о равных парах. Заметим, что величина y равна полусумме длительностей обслуживания всех требований множества N обоими приборами. При задании значения y учитывается тот факт, что $\sum_{i=1}^p (u_i + v_i) = \sum_{i=1}^p (w_i + x_i)$, ибо в противном случае задача о равных парах, очевидно, не имеет решения.

Пусть задача о равных парах имеет решение и R_m , $1 \leq m \leq p$, — найденные подмножества. Расписание s^0 может быть построено следующим образом. Пусть $R_1 = \{u_{i_1}, v_{i_2}, w_{i_3}, x_{i_4}\}$. Тогда начиная с момента времени $d = 0$ прибор A обслуживает требования $U_{i_1}, X_{i_4}, W_{i_3}, V_{i_2}$ в указанной последовательности, а прибор B обслуживает эти же требования в последовательности $X_{i_4}, U_{i_1}, V_{i_2}, W_{i_3}$, причем в интервале $(0, 4u_{i_1} + 4v_{i_2} + 3Z + 2]$ оба прибора A и B функционируют без простоев. Вслед за указанными требованиями аналогичным образом обслуживаются требования, определяемые подмножеством R_2 и т.д.

Пусть существует расписание s^0 . Поскольку $a(N) = b(N) = y$, то при расписании s^0 оба прибора A и B функционируют без простоев. Предполо-

жим, что для требования $j \in N$ выполняется $t_{jA}^0 = 0$, и это требование является либо W -требованием, либо X -требованием. В этом случае нельзя указать такое требование $k \in N$, $k \neq j$, что $t_{kB}^0 = 0$ и требование k обслуживается без задержек. Отсюда следует, что требование j является либо U -требованием, либо V -требованием. Для определенности предположим, что требование j является U -требованием (случай, когда требование j является V -требованием, аналогичен рассматриваемому). Поскольку требование j является U -требованием, то $\bar{t}_{jA} = t_{jB}^0 = 1$ и требование k такое, что $t_{kB}^0 = 0$, может быть либо W -требованием, либо X -требованием (так как в этом случае $\bar{t}_{kB} = 1$ и требование k может обслуживаться без задержек). Пусть l — требование, которое обслуживается прибором B после завершения обслуживания требования j . Очевидно, требование l не может быть ни U -требованием, ни W -требованием, ни X -требованием, так как в этом случае $\bar{t}_{lB} < \bar{t}_{kA}$ и требование l не может обслуживаться без задержек. Следовательно, требование l должно быть V -требованием. Для того чтобы прибор A функционировал без простоев, необходимо, чтобы требование q , обслуживаемое прибором A вслед за требованием k , было либо W -требованием, либо X -требованием, но не одного вида с требованием k . При этом должно выполняться условие $a_k + a_q = b_j + b_l$, так как в противном случае требование q не может быть обслужено без задержек. Элементы множеств U , V , W и X , определяющие длительность обслуживания требований j , l , k и q , образуют подмножество R_1 в решении задачи о равных парах. Рассуждая аналогичным образом, нетрудно построить остальные подмножества R_m , $m = 2, \dots, p$.

Тем самым за $O(p)$ операций реализовано сведение задачи о равных парах к рассматриваемой задаче распознавания.

§ 2. Общее время обслуживания. Три и более приборов

Рассматривается задача построения оптимального по быстродействию расписания s^* обслуживания множества $N = \{1, 2, \dots, n\}$ требований в системе с нефиксированными маршрутами, состоящей из множества $\mathcal{M} = \{1, 2, \dots, M\}$ приборов. Длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L \in \mathcal{M}$ предполагаются заданными. Все требования поступают в систему в момент времени $d = 0$. Прерывания процесса обслуживания каждого требования каждым прибором запрещены.

Напомним, что при любом расписании s значение

$$\bar{t}_{\max}(s) \geq \max \left\{ \max_{i \in N} t_i(\mathcal{M}), \max_{L \in \mathcal{M}} t_L(N) \right\}. \quad (2.1)$$

В этом параграфе показано, что задача построения оптимального расписания s^* является NP -трудной при $M \geq 3$. Приводятся алгоритмы построения s^* , трудоемкость которых не превосходит $O(nM)$ операций, при наличии в системе так называемых доминирующих приборов. Описан алгоритм трудоемкости $O(n^2 M^3)$ операций построения s^* , если $\max_{L \in \mathcal{M}} t_L(N) \geq (M^2 + 2M - 1)t^*$, где $t^* = \max\{t_{iL} \mid i \in N, L \in \mathcal{M}\}$.

2.1. Покажем, что при $M \geq 3$ задача построения оптимального по быстродействию расписания является NP -трудной. Этой задаче соответствует следующая задача распознавания: определить, существует ли такое расписание s^0 обслуживания требований множества N , что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что при $M = 3$ к этой задаче сводится задача о разбиении (см. вводную часть этой главы).

Положим $n = n_0 + 1$, $t_{iL} = e_i$, $i = 1, \dots, n_0$; $t_{nL} = E$, $L = 1, 2, 3$, $y = 3E$. Покажем, что в построенной задаче расписание s^0 существует тогда и только тогда, когда имеет решение задача о разбиении.

Если существует указанное разбиение множества N^0 на подмножества N_1^0 и N_2^0 , то искомое расписание s^0 можно построить, например, следующим образом. Каждый прибор должен обслуживать требования непрерывно одно за другим, начиная с момента времени $d = 0$; первый прибор обслуживает требования в последовательности $(n, \pi_{N_2^0}, \pi_{N_1^0})$, второй — в последовательности $(\pi_{N_1^0}, n, \pi_{N_2^0})$, третий — в последовательности $(\pi_{N_2^0}, \pi_{N_1^0}, n)$.

Пусть существует расписание s^0 . Поскольку $t_L(N) = 3E$, $L = 1, 2, 3$, то в силу (2.1) $\bar{t}_{\max}(s^0) = 3E$ и каждый прибор во временном интервале $(0, 3E]$ функционирует без простоев. Так как $t_n(\mathcal{M}) = 3E$, то в любой момент времени $t \in (0, 3E]$ требование n обслуживается одним из приборов. Следовательно, найдется прибор L , $1 \leq L \leq 3$, такой, что требование n обслуживается во временном интервале $(E, 2E]$, а требования множества N^0 — в интервалах $(0, E]$ и $(2E, 3E]$. Обозначая через N_1^0 и N_2^0 множества требований, обслуживаемых в интервалах $(0, E]$ и $(2E, 3E]$ соответственно, получаем решение задачи о разбиении.

Реализация сведения задачи о разбиении к задаче о существовании расписания s^0 требует выполнения не более $O(n_0)$ операций.

Таким образом, задача построения оптимального по быстродействию расписания в случае, когда $M \geq 3$, является NP -трудной.

2.2. Рассмотрим систему, содержащую $M \geq 3$ приборов, среди которых имеется один *доминирующий прибор*, и покажем, что *оптимальное по быстродействию расписание обслуживания n требований ($n \geq M$) в этой системе может быть построено в результате выполнения не более чем $O(nM)$ элементарных операций.*

Будем говорить, что прибор Q доминирует прибор H , $Q, H \in \mathcal{M}$, $Q \neq H$, если $t_{iQ} \geq t_{iH}$ для всех $i, j \in N$, $i \neq j$.

Аналогично требование i доминирует требование j , $i, j \in N$, $i \neq j$, если $t_{iQ} \geq t_{jH}$ для всех $Q, H \in \mathcal{M}$, $Q \neq H$.

Прибор (требование) назовем *доминирующим*, если он (оно) доминирует все остальные приборы (требования).

Не нарушая общности, будем считать, что доминирующим является прибор 1.

Введем в рассмотрение матрицу $\|\lambda_{Li}\|$ размерности $M \times n$, каждая строка которой представляет собой некоторую перестановку элементов множества N . Если $\lambda_{Li} = k$, то содержательно это означает, что прибор L обслуживает требование k i -м по порядку.

Зададим матрицу $\|\lambda_{Li}\|$, полагая

$$\lambda_{1i} = i, \quad i = 1, \dots, n;$$

$$\lambda_{L1} = \lambda_{L-1,n}, \quad \lambda_{Li} = \lambda_{L-1,i-1}, \quad i = 2, \dots, n, \quad L = 2, \dots, M.$$

Эта матрица представляет собой латинский прямоугольник, построенный на множестве N . Напомним, что *латинским прямоугольником (квадратом)*, построенным на конечном множестве S , называется прямоугольная (квад-

ратная) матрица, каждая строка которой представляет собой некоторую перестановку элементов множества S и каждый элемент S встречается не более одного раза (ровно один раз) в каждом столбце матрицы.

Положим

$$\bar{t}_{11} = t_{11}, \quad \bar{t}_{i1} = \bar{t}_{i-1,1} + t_{i1}, \quad i = 2, \dots, n;$$

$$\bar{\lambda}_{LiL} = \bar{t}_{i1}, \quad i = 1, \dots, n, \quad L = 2, \dots, M.$$

Нетрудно убедиться, что построенная матрица $\|\bar{t}_{iL}\|$ определяет расписание с общим временем обслуживания, равным

$$t_1(N) = \max \{ \max_{L \in \mathcal{M}} t_L(N), \max_{i \in N} t_i(\mathcal{M}) \}.$$

Из (2.1) следует, что это расписание является оптимальным. Трудоемкость построения каждой из матриц $\|\lambda_{Li}\|$ и $\|\bar{t}_{iL}\|$ не превосходит $O(nM)$ операций.

В силу симметрии рассматриваемой задачи можно заключить (см. п.1.1 § 1 этой главы), что построение оптимального по быстродействию расписания в случае, когда одно из обслуживаемых требований является доминирующим и $n \leq M$, также может быть осуществлено за $O(nM)$ элементарных операций.

Условие $n \geq M$ при наличии доминирующего прибора (или, что то же, условие $n \leq M$ при наличии доминирующего требования) является существенным. Если это условие не выполняется, то *задача построения оптимального расписания является NP-трудной* (в п. 2.1 при сведении задачи о разбиении к задаче построения оптимального расписания при $M = 3$ требование $n = n_0 + 1$ является доминирующим).

2.3. Пусть система, содержащая M приборов, обслуживает n требований ($n \geq M$) и среди приборов имеются два прибора, *доминирующие все остальные приборы, но не друг друга*. Покажем, что и в этом случае построение оптимального по быстродействию расписания требует выполнения не более $O(nM)$ операций.

Не нарушая общности, можно считать, что указанными приборами являются приборы с номерами 1 и 2. Приборы, отличные от приборов 1 и 2, назовем *недоминирующими*.

При помощи алгоритма, описанного в п. 1.2 § 1 этой главы, найдем оптимальное по быстродействию расписание обслуживания всех требований приборами 1 и 2. Для удобства дальнейших рассуждений будем полагать, что в построенном расписании последовательности обслуживания требований приборами 1 и 2 определяются соотношениями

$$\lambda_{11} = 1, \quad \lambda_{1i} = i, \quad \lambda_{21} = n, \quad \lambda_{2i} = \lambda_{1,i-1}, \quad i = 2, \dots, n \quad (2.2)$$

(в противном случае требования можно соответствующим образом переименовать).

В силу соотношения (1.2) § 1 этой главы общее время обслуживания требований приборами 1 и 2 равно $\max\{t_1(N), t_2(N), t_{n1} + t_{n2}\}$.

1) Предположим, что

$$t_{n1} + t_{n2} \geq \max\{t_1(N), t_2(N)\}. \quad (2.3)$$

Тогда общее время обслуживания в исходной задаче будет не меньше $t_n(\mathcal{M})$.

Пусть первые две строки матрицы $\|\lambda_{Li}\|$ заполнены в соответствии с (2.2). Положим

$$\lambda_{L1} = \lambda_{L-1,n-1}, \quad \lambda_{Li} = \lambda_{L-1,i-1}, \quad L = 2, \dots, M, \quad i = 2, \dots, n-1.$$

Из алгоритма, описанного в п. 1.2 § 1 этой главы, следует, что

$$\bar{t}_{11} = t_{n1} + t_{n2} - t_1(N) + t_{11};$$

$$\bar{t}_{i1} = \bar{t}_{i-1,1} + t_{i1}, \quad i = 2, \dots, n;$$

$$\bar{t}_{n2} = t_{n2}, \quad \bar{t}_{12} = \bar{t}_{n2} + t_{12};$$

$$\bar{t}_{i2} = \bar{t}_{i-1,2} + t_{i2}, \quad i = 2, \dots, n.$$

Положим

$$\bar{t}_{\lambda_{Li}L} = \bar{t}_{i1}, \quad L = 2, \dots, M, \quad i = 1, \dots, n-1.$$

Проведенные построения означают, что за время обслуживания требования n прибором 2 все остальные требования будут заведомо обслужены всеми остальными приборами. Затем, во время обслуживания требования n прибором 1, все остальные требования обслуживаются прибором 2, а недоминирующие приборы простаивают.

Положим

$$\lambda_{Ln} = n, \quad L = 3, \dots, M;$$

$$\bar{t}_{n3} = \bar{t}_{n1} + t_{n3};$$

$$\bar{t}_{nL} = \bar{t}_{n,L-1} + t_{nL}, \quad L = 4, \dots, M.$$

В результате получаем расписание с общим временем обслуживания, равным $t_n(M)$.

2) Пусть неравенство (2.3) не имеет места. Как и выше, полагаем, что первые две строки матрицы $\|\lambda_{Li}\|$ заполнены согласно (2.2). Остальные строки этой матрицы заполним таким образом, чтобы строка L имела вид $(n-L+2, n-L+3, \dots, n, 1, 2, \dots, n-L+1)$, $L = 3, \dots, M$. Отметим, что построенная матрица $\|\lambda_{Li}\|$ представляет собой латинский прямоугольник, построенный на множестве N .

Моменты $\bar{t}_{i1}, \bar{t}_{i2}, i = 1, \dots, n$, завершения обслуживания требований приборами 1 и 2 могут быть найдены из расписания, построенного по алгоритму, описанному в п. 1.2 § 1 этой главы. Положим

$$\bar{t}_{\lambda_{Li}L} = \bar{t}_{i1}, \quad L = 3, \dots, M, \quad i = 1, \dots, L-2,$$

$$\bar{t}_{\lambda_{Li}L} = \bar{t}_{i-1,2}, \quad L = 3, \dots, M, \quad i = L, \dots, n.$$

Первое из этих соотношений означает, что каждое требование с номером $j, j = 2, \dots, n-1$, перед его обслуживанием прибором 1 будет обслужено недоминирующими приборами в порядке возрастания номеров от $L_j = n-j+2$ ($L_j \leq M$) до M . Из второго соотношения следует, что каждое требование с номером $k, k = 1, \dots, n-2$, после завершения его обслуживания прибором 2 будет обслужено недоминирующими приборами в порядке возрастания номеров от 3 до $L_k = n-k+1$ ($L_k \leq M$). Таким образом, построено расписание обслуживания всех требований множества $N \setminus \{n\}$.

Для требования n положим $\bar{t}_{n3} = \bar{t}_{n2} + t_{n3}$ и $\bar{t}_{nL} = \bar{t}_{n,L-1} + t_{n,L-1}$, $L = 4, \dots, M$.

Если $t_1(N) \geq t_n(M)$, то построенная матрица $\|\bar{t}_{iL}\|$ определяет расписание с общим временем обслуживания, равным $\max\{t_1(N), t_2(N)\}$. В противном случае положим $\bar{t}_{n1} = \bar{t}_{nM} + t_{n1}$. В результате получаем расписание с общим временем обслуживания, равным $\max\{t_2(N), t_n(N)\}$.

Таким образом, в любом случае построено оптимальное расписание s^* . Трудоемкость построения каждой из матриц $\|\lambda_{Li}\|$ и $\|\bar{t}_{iL}\|$ не превосходит $O(nM)$ операций.

В силу симметрии рассматриваемой задачи можно заключить, что построение оптимального по быстродействию расписания в случае, когда $n \leq M$ и среди требований имеются два требования, доминирующие все остальные требования, но не друг друга, также может быть осуществлено за $O(nM)$ операций.

Указанные соотношения между n и M являются существенными. Если эти соотношения не выполняются, то соответствующая задача оказывается NP -трудной.

Действительно, рассмотрим задачу о разбиении. Положим $n = n_0 + 2$, $M = 3$; $t_{i3} = 0$, $t_{i1} = t_{i2} = e_i$, $i = 1, \dots, n_0$; $t_{n_0+1,1} = t_{n_0+1,2} = t_{n_0+2,1} = t_{n_0+2,2} = E$; $t_{n_0+1,3} = t_{n_0+2,3} = 2E$. Заметим, что требования $n_0 + 1$ и $n_0 + 2$ доминируют остальные требования, но не друг друга, причем $n > M$.

Нетрудно показать, что задача о разбиении имеет решение тогда и только тогда, когда при указанных n , M и t_{iL} существует расписание s^0 такое, что $\bar{t}_{\max}(s^0) \leq y = 4E$.

2.4. П р и м е р. Построим оптимальное по быстродействию расписание обслуживания пяти требований четырьмя приборами. Длительности t_{iL} приведены в табл. 2.1.

Нетрудно убедиться, что приборы 1 и 2 доминируют остальные приборы, но не друг друга.

Поскольку неравенство (2.3) не выполняется ($t_{51} + t_{52} = 43$, $t_1(N) = 45$, $t_2(N) = 63$), то матрицы $\|\lambda_{Li}\|$ и $\|\bar{t}_{iL}\|$ строим так, как это описано в п.2.3, 2).

Используя алгоритм п.1.2 § 1 этой главы, строим оптимальное расписание обслуживания всех требований двумя приборами 1 и 2. Таким распи-

Т а б л и ц а 2.1

$i \backslash L$	1	2	3	4
1	7	12	1	3
2	7	8	5	2
3	10	6	4	7
4	8	7	3	3
5	13	30	5	2

$i \backslash L$	1	2	$i \backslash L$	1	2
1	7	42	4	32	63
2	14	50	5	45	30
3	24	56			

санием является, в частности, расписание, определяемое моментами $\bar{t}_{iL}$, $L = 1, 2$, завершения обслуживания требований, приведенными в табл. 2.2.

В рассматриваемом случае исходная нумерация требований такова, что выполняется (2.2) и матрица $\|\lambda_{Li}\|$ имеет вид

$$\begin{vmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 3 & 4 \\ 4 & 5 & 1 & 2 & 3 \\ 3 & 4 & 5 & 1 & 2 \end{vmatrix}.$$

Учитывая, что $t_1(N) = 45 < t_5(\mathcal{M}) = 50$, получаем матрицу

$$\|\bar{t}_{iL}\| = \begin{vmatrix} 7 & 42 & 50 & 56 \\ 14 & 50 & 56 & 63 \\ 24 & 56 & 63 & 7 \\ 32 & 63 & 7 & 14 \\ 50 & 30 & 35 & 37 \end{vmatrix},$$

определяющую оптимальное расписание s^* . Общее время обслуживания равно $\bar{t}_{\max}(s^*) = t_2(N) = 63$.

2.5. Опишем алгоритм трудоемкости $O(n^2 M^3)$ операций построения оптимального (по быстрдействию) расписания s^* в предположении, что $t^*(N) = \max_{L \in \mathcal{M}} t_L(N) \geq (M^2 + 2M - 1)t^*$, где $t^* = \max\{t_{iL} \mid i \in N, L \in \mathcal{M}\}$.

Далее везде будем предполагать $t_L(N) = t^*(N)$, $L = 1, \dots, M$. В противном случае исходные длительности обслуживания t_{iL} можно было бы соответствующим образом "удлинить", не увеличивая значения t^* .

Введем в рассмотрение множество V , состоящее из n векторов v вида $(t_{i1} - t_{i2}, t_{i2} - t_{i3}, \dots, t_{iM-1} - t_{iM}, t_{iM} - t_{i1})$, $i \in N$. Очевидно, $\sum_{v \in V} v = 0$, и если в качестве нормы $\|v\|$ вектора v выбрать максимум модуля его компонент, то $\|v\| \leq t^*$, $v \in V$.

Нумеруя векторы v множества V согласно алгоритму, описанному в п.8.2 § 8 гл. 1, получаем $V = \{v^1, v^2, \dots, v^n\}$. Перенумеруем (если это необходимо) требования множества N так, чтобы $v^i = (t_{i1} - t_{i2}, t_{i2} - t_{i3}, \dots, t_{iM-1} - t_{iM}, t_{iM} - t_{i1})$.

Из утверждения, доказанного в п.8.2 § 8 гл. 1, следует неравенство

$$-Mt^* \leq \sum_{i=1}^k t_{iL} - \sum_{i=1}^k t_{i, L+1} \leq Mt^*, \quad L = 1, \dots, M-1, \quad (2.4)$$

$$k = 1, \dots, n,$$

а также неравенство

$$-Mt^* \leq \sum_{i=1}^k t_{iM} - \sum_{i=1}^k t_{i1} \leq Mt^*, \quad k = 1, \dots, n. \quad (2.5)$$

Расписание s^* построим следующим образом. Все приборы функционируют во временном интервале $(0, t^*(N)]$ без простоев и обслуживают требования непрерывно одно за другим. Первый прибор обслуживает требования в последовательности $1, 2, \dots, n$; прибор $L, 2 \leq L \leq M$, — в последовательности $i_L + 1, i_L + 2, \dots, n, 1, 2, \dots, i_L$.

Опишем процедуру выбора требований $i_2, i_3, \dots, i_n$.

В качестве i_2 выберем наибольшее значение j , для которого

$$\sum_{i=j+1}^n t_{i2} \geq (M+1)t^*.$$

Положим $t_{12}^0 = \sum_{i=i_2+1}^n t_{i2}$.

Пусть выбраны требования $i_2, i_3, \dots, i_{Q-1}$ и вычислены моменты времени $t_{1L}^0 = \sum_{i=i_L+1}^n t_{iL}, L = 2, \dots, Q-1, 2 \leq Q \leq M$.

В качестве i_Q выберем наибольшее значение j , для которого

$$\sum_{i=j+1}^n t_{iQ} \geq t_{1,Q-1}^0 + (M+1)t^*.$$

Положим $t_{1Q}^0 = \sum_{i=i_Q+1}^n t_{iQ}$.

Этот процесс продолжается до тех пор, пока не будет выбрано требование i_M и вычислено значение $t_{1M}^0 = \sum_{i=i_M+1}^n t_{iM}$.

Реализуемость описанной процедуры выбора требований $i_2, i_3, \dots, i_M$ можно обосновать следующим образом.

Нетрудно видеть, что $t_{12}^0 < (M+2)t^*$, так как противоположное неравенство противоречило бы правилу выбора требования i_2 . Так как $(M+2)t^* < (M^2 + 2M - 1)t^* = t^*(N)$ для всех $M \geq 2$, то требование i_2 действительно может быть выбрано.

Далее, из аналогичных рассуждений следует, что $t_{1L}^0 - t_{1,L-1}^0 < (M+2)t^*, L = 2, \dots, M$. Отсюда получаем $t_{1L}^0 \leq (L-1)(M+2)t^*$. Так как даже при $L = M$ имеем $(M-1)(M+2)t^* = (M^2 + M - 2)t^* < (M^2 + 2M - 1)t^*$ для всех $M > 0$, то все требования $i_L, L = 2, \dots, M$ могут быть выбраны. Кроме того,

$$t^*(N) > t_{1M}^0 + (M+1)t^*, \quad (2.6)$$

так как $t_{1M}^0 < (M-1)(M+2)t^*$.

Покажем, что при обслуживании требований в указанных последовательностях каждый прибор может функционировать без простоев во времен-

ном интервале $(0, t^*(N)]$. Для этого достаточно показать, что ни одно из требований не будет обслуживаться несколькими приборами одновременно.

Пусть $j \in N$ — произвольное требование. Обозначим через Q такое максимальное значение L , $1 \leq L \leq M$, для которого $t_{1L}^0 + \sum_{i=1}^j t_{iL} \leq t^*(N)$. Это

означает, что требование j обслуживается приборами L , $L = 2, \dots, Q$, после момента времени t_{1L}^0 , а приборами $L, L = Q+1, \dots, M$ (если $Q < M$) — до момента времени t_{1L}^0 .

Чтобы требование j одновременно не обслуживалось несколькими приборами, должны выполняться неравенства

$$t_{jL}^0 \geq \bar{t}_{j, L-1}, \quad L = 2, \dots, M, \quad L \neq Q+1; \quad (2.7)$$

$$t_{jQ}^0 \geq \bar{t}_{j, Q+1}, \quad Q \neq M; \quad (2.8)$$

$$t_{j1}^0 \geq \bar{t}_{jM}, \quad Q \neq M. \quad (2.9)$$

Пусть $L \leq Q$. Тогда

$$t_{jL}^0 - \bar{t}_{j, L-1} = t_{1L}^0 + \sum_{i=1}^{j-1} t_{iL} - (t_{1, L-1}^0 + \sum_{i=1}^{j-1} t_{i, L-1} + t_{j, L-1}).$$

Поскольку по построению $t_{1L}^0 - t_{1, L-1}^0 \geq (M+1)t^*$ и в силу (2.4) $\sum_{i=1}^{j-1} t_{iL} - \sum_{i=1}^{j-1} t_{i, L-1} \geq -Mt^*$, то $t_{jL}^0 - \bar{t}_{j, L-1} \geq (M+1)t^* - Mt^* - t_{j, L-1} = t^* - t_{j, L-1} \geq 0$.

Аналогично, при $Q+2 < L \leq M$ имеем

$$t_{jL}^0 - \bar{t}_{j, L-1} = \sum_{i=i_L+1}^{j-1} t_{iL} - \sum_{i=i_L-1+1}^{j-1} t_{i, L-1} - t_{j, L-1}.$$

Нетрудно убедиться, что

$$\sum_{i=i_L+1}^{j-1} t_{i, L-1} = t_{1L}^0 - t^*(N) + \sum_{i=1}^{j-1} t_{iL},$$

$$\sum_{i=i_L-1+1}^{j-1} t_{iL} = t_{1, L-1}^0 - t^*(N) + \sum_{i=1}^{j-1} t_{i, L-1}.$$

Тогда

$$t_{jL}^0 - \bar{t}_{j, L-1} = t_{1L}^0 - t_{1, L-1}^0 + \sum_{i=1}^{j-1} t_{iL} - \sum_{i=1}^{j-1} t_{i, L-1} - t_{j, L-1} \geq 0.$$

Таким образом, неравенство (2.7) выполняется.

Нетрудно проверить, что $t_{jQ}^0 = t_{1Q}^0 + \sum_{i=1}^{j-1} t_{iQ}$ и

$$\bar{t}_{j, Q+1} = \sum_{i=i_{Q+1}+1}^j t_{i, Q+1} = t_{1, Q+1}^0 - t^*(N) + \sum_{i=1}^{j-1} t_{i, Q+1} + t_{j, Q+1},$$

поскольку, кроме того, $t_{1Q}^0 - t_{1, Q+1}^0 > -(M+2)t^*$, то с учетом (2.4) имеем

$$t_{jQ}^0 - \bar{t}_{j, Q+1} > -(M+2)t^* - Mt^* + t^*(N) - t_{j, Q+1} \geq (M^2 - 4)t^*,$$

т.е. при $M \geq 2$ неравенство (2.8) также выполняется.

Нетрудно проверить, что

$$t_{jM}^0 = \sum_{i=t_M+1}^{j-1} t_{iM} = t_{1M}^0 - t^*(N) + \sum_{i=1}^{j-1} t_{iM}.$$

В силу (2.5)

$$\sum_{i=1}^{j-1} t_{iM} \leq \sum_{i=1}^{j-1} t_{i1} + Mt^*.$$

Кроме того, из (2.6) следует, что

$$t_{1M}^0 - t^*(N) + Mt^* < -t^*.$$

Таким образом,

$$t_{jM}^0 \leq \sum_{i=1}^{j-1} t_{i1} + t_{1M}^0 - t^*(N) + Mt^* < \sum_{i=1}^{j-1} t_{i1} - t^* = t_{1j}^0 - t^*,$$

откуда вытекает неравенство (2.9).

Нетрудно видеть, что оптимальное расписание s^* может быть построено в результате выполнения не более чем $O(n^2 M^3)$ операций, поскольку наиболее трудоемким этапом при его построении является этап перенумерации требований множества N (нумерации векторов v множества V (см. § 8 гл. 1)).

§ 3. Общее время обслуживания. Прерывания

В этом параграфе рассматривается задача построения оптимального по быстродействию расписания функционирования систем с нефиксированными маршрутами при условии, что в процессе обслуживания каждого требования каждым прибором допускаются прерывания.

3.1. В систему, содержащую M последовательных приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Известны длительности $t_{iL} \geq 0$ обслуживания каждого требования $i \in N$ каждым прибором $L \in \mathcal{M} = \{1, 2, \dots, M\}$. Порядок прохождения приборов каждым требованием может быть произвольным и необязательно одинаковым для разных требований. По-прежнему предполагается, что каждый прибор одновременно обслуживает не более одного требования, а каждое требование одновременно обслуживается не более чем одним прибором.

В процессе обслуживания требований допускаются *прерывания*: каждое требование i может обслуживаться каждым прибором L в несколько "приемов" при сохранении условия, что суммарная длительность этого обслуживания равна t_{iL} (если $t_{iL} > 0$). Процесс обслуживания требования i прибором L может быть прерван в любой момент времени. После прерывания прибор L может обслуживать любое другое требование, а требование

i — обслуживаться любым другим прибором. Процесс обслуживания требования i прибором L может быть возобновлен в любой момент времени. Число прерываний может быть произвольным, но конечным.

Если $\bar{t}_{iL}$ — момент (полного) завершения обслуживания требования i прибором L при некотором расписании s и $\bar{t}_{\max}(s) = \max \{ \bar{t}_{iL} \mid i \in N, L \in \mathcal{M} \}$, то, как и прежде (см. (1.1)),

$$\bar{t}_{\max}(s) \geq T, \quad T = \max \{ \max_{i \in N} t_i(\mathcal{M}), \max_{L \in \mathcal{M}} t_L(N) \}. \quad (3.1)$$

Как следует из результатов, приведенных в § 1 этой главы, при $M = 2$ существует расписание s^* без прерываний, при котором $\bar{t}_{\max}(s^*) = T$, и это расписание может быть построено за $O(n)$ операций.

Ниже описываются полиномиальные алгоритмы построения расписания s^* (с прерываниями), при котором $\bar{t}_{\max}(s^*) = T$ для произвольного M . В силу (3.1) это расписание является оптимальным по быстродействию.

В п. 3.6 рассматриваемая ситуация обобщается на случай, когда каждый прибор L представляет собой систему, содержащую несколько параллельных (необязательно идентичных) приборов.

Задача, рассматриваемая в п. 3.8, отличается от той, которая была описана в п. 1.4 этой главы, только тем, что в процессе обслуживания каждого требования каждым прибором допускаются прерывания.

В заключение параграфа устанавливается NP -трудность задачи построения оптимального по быстродействию расписания в случае, когда после прерывания процесса обслуживания требования i любым прибором L это требование не может обслуживаться другими приборами. Устанавливается также NP -трудность построения оптимального расписания (с прерываниями), когда в процессе обслуживания каждого требования запрещены задержки.

3.2. Пусть все t_{iL} — целые числа. Покажем, что существует расписание s^* такое, что $\bar{t}_{\max}(s^*) = T$ и прерывания процесса обслуживания имеют место только в целочисленные моменты времени.

Построим двудольный неориентированный мультиграф $\Gamma = (N, \mathcal{M}; U)$, где $N = \{1, 2, \dots, n\}$ и $\mathcal{M} = \{1, 2, \dots, M\}$ — множества вершин, U — множество ребер, каждые две вершины $i \in N$ и $L \in \mathcal{M}$ соединены t_{iL} ребрами (если $t_{iL} > 0$).

Под *реберной раскраской мультиграфа* будем понимать, как обычно, такое сопоставление цветов (чисел $1, 2, \dots$) ребрам, при котором никакие два ребра, инцидентные одной и той же вершине, не окрашены в одинаковый цвет (им не приписано одно и то же число).

Известно, что минимальное число цветов, необходимых для раскраски ребер двудольного графа (мультиграфа), равно максимальной степени вершины. Поскольку по построению максимальная степень вершины мультиграфа Γ равна T , то он может быть раскрашен в T цветов (т.е. его ребрам могут быть сопоставлены числа $1, 2, \dots, T$).

Интерпретируя цвета — числа $1, 2, \dots, T$ — как номера временных интервалов $(0, 1], (1, 2], \dots, (T-1, T]$, а окраску ребра $(i, L) \in U$ в цвет τ — как обслуживание требования i прибором L во временном интервале $(\tau - 1, \tau]$, приходим к заключению, что искомое расписание s^*

существует, и для его построения достаточно решить задачу раскраски мультиграфа Γ в T цветов.

Построенное таким образом расписание s^* может оказаться неприемлемым из-за неоправданно большого числа прерываний. Для построения оптимального расписания с меньшим числом прерываний может быть использован следующий подход.

Введем понятие *обобщенной реберной раскраски мультиграфа*. Эта раскраска отличается от обычной только тем, что, во-первых, ребра, инцидентные одной и той же паре вершин, могут окрашиваться в одинаковый цвет и, во-вторых, если в цвет τ окрашено k_τ ребер вида (i_1, L_1) , то ребра вида (i_2, L_2) ($i_1 \neq i_2, L_1 \neq L_2$) либо не окрашиваются в цвет τ , либо окрашиваются ровно k_τ из них. Число k_τ будем называть *кратностью цвета τ* . При $k_\tau \leq 1$ получаем обычную реберную раскраску мультиграфа.

Обобщенная реберная раскраска мультиграфа Γ называется *оптимальной*, если сумма кратностей k_τ всех использованных при раскраске цветов τ равна T . Известно, что такая раскраска может быть получена в результате выполнения $O((n+M)r \log t^*)$ операций, где $t^* = \max\{t_{iL} \mid i \in N, L \in \mathcal{M}\}$, r — число ненулевых t_{iL} , $i \in N, L \in \mathcal{M}$. При этом число используемых цветов не превышает $O(r \log t^*)$.

Если известна оптимальная обобщенная раскраска ребер мультиграфа Γ , то соответствующее оптимальное (по быстрдействию) расписание s^* может быть построено следующим образом. Пронумеруем использованные при раскраске цвета числами $1, 2, \dots, T_1$ (очевидно, $T_1 \leq T$). Пусть $k_\tau, \tau = 1, 2, \dots, T_1$, — соответствующие кратности цветов. Разобьем интервал планирования $(0, T]$ на T_1 подынтервалов $(0, k_1], (k_1, k_1 + k_2], \dots, (\sum_{\tau=1}^{T_1-1} k_\tau, \sum_{\tau=1}^{T_1} k_\tau]$ и пронумеруем их числами $1, 2, \dots, T_1$. Если хотя

бы одно из ребер вида (i, L) окрашено в цвет $\tau, 1 \leq \tau \leq T_1$ (а следовательно, в цвет τ окрашено ровно k_τ ребер вида (i, L)), то во временном интервале с номером τ требование i непрерывно обслуживается прибором L .

Для уменьшения числа прерываний естественно стремиться к нахождению оптимальной обобщенной раскраски с наименьшим числом цветов T_1 .

3.3. П р и м е р. Имеется четыре источника и три приемника информации. Длительности t_{iL} передачи информации от источника $i, i = 1, 2, 3, 4$, приемнику $L, L \in \{I, II, III\}$ приведены в табл. 3.1.

Если $t_{iL} = 0$, то от источника i приемнику L информация не передается.

Предполагается, что передача информации от каждого источника каждому приемнику может производиться по частям, причем возобновление передачи осуществляется с прерванного места.

Т а б л и ц а 3.1

$i \backslash L$	I	II	III
1	4	3	2
2	3	5	0
3	0	1	3
4	3	0	4

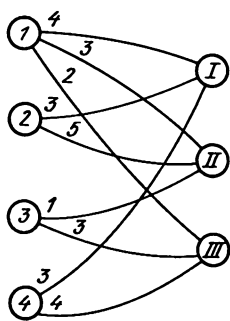


Рис. 3.1

Требуется передать информацию от всех источников всем приемникам как можно скорее.

Для решения этой задачи используем подход, изложенный в п. 3.2 Положим $N = \{1, 2, 3, 4\}$, $\mathcal{M} = \{I, II, III\}$ и построим двудольный мультиграф $\Gamma = (N, \mathcal{M}; U)$, соединяя вершину $i \in N$ с вершиной $L \in \mathcal{M}$ ребрами, если $t_{iL} > 0$, и не соединяя их в противном случае. Полученный при этом мультиграф Γ изображен на рис. 3.1; числа, приписанные ребрам, означают их кратность. Максимальную степень в мультиграфе Γ имеет вершина I , и значит $T = 10$.

Найдем оптимальную обобщенную реберную раскраску мультиграфа Γ . Одна из таких раскрасок приведена в табл. 3.2.

Таблица 3.2

ребро \ цвет	1	2	3	4	5
(1, I)	4				
(1, II)		3			
(1, III)			1		1
(2, I)		3			
(2, II)	4		1		
(3, II)				1	
(3, III)		3			
(4, I)			1	1	1
(4, III)	4				

В этой таблице указаны кратности цветов и ребра, окрашенные в соответствующий цвет.

Соответствующее расписание представлено на рис. 3.2.

3.4. Пусть t_{iL} — произвольные неотрицательные числа. Покажем, что и в этом случае существует расписание s^* такое, что $\bar{t}_{\max}(s^*) = T$, и укажем способ его построения.

Построим матрицу $A = \|a_{pq}\|$ размерности $(n + M) \times (n + M)$ вида

$$A = \left\| \begin{array}{c|c} \|t_{iL}\| & D_n \\ \hline D_M & B \end{array} \right\|,$$

где $\|t_{iL}\|$ — матрица размерности $n \times M$ длительностей обслуживания требований, D_n и D_M — диагональные матрицы размерностей $n \times n$ и $M \times M$ соответственно, B — матрица размерности $M \times n$ такие, что

$$\sum_{q=1}^{n+M} a_{pq} = T, \quad p = 1, \dots, n+M; \quad \sum_{p=1}^{n+M} a_{pq} = T, \quad q = 1, \dots, n+M.$$

Напомним, что в диагональной матрице все элементы, за возможным исключением элементов, стоящих на главной диагонали, равны нулю. В

качестве матрицы B может быть выбрана, например, матрица, транспонированная к $\|t_{iL}\|$.

Построенная таким образом матрица A *кратна дважды стохастической матрице* (т.е. суммы ее элементов, расположенных в каждой строке и каждом столбце, равны) и, следовательно, допускает разложение вида

$$A = \sum_{k=1}^l \delta_k \Pi_k, \quad (3.2)$$

где $l \leq (n + M - 1)^2 + 1$, $\sum_{k=1}^l \delta_k = T$, $\delta_k \geq 0$, $k = 1, \dots, l$, а $\Pi_k = \|\theta_{pq}^k\|$ — *перестановочные матрицы* (матрицы размерности $(n + M) \times (n + M)$ с

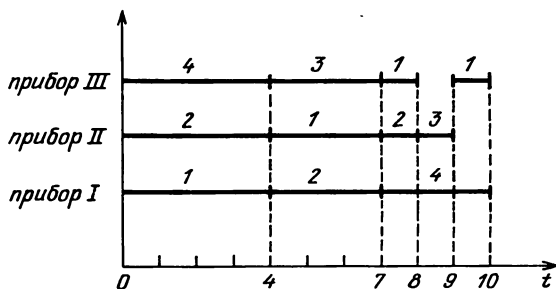


Рис. 3.2

элементами 0 и 1, в каждой строке и каждом столбце которых содержится ровно одна единица).

Если найдено разложение (3.2), то соответствующее оптимальное (по быстродействию) расписание s^* может быть построено следующим образом. Разобьем интервал планирования $(0, T]$ на l подынтервалов $(0, \delta_1]$,

$(\delta_1, \delta_1 + \delta_2], \dots, (\sum_{k=1}^{l-1} \delta_k, \sum_{k=1}^l \delta_k]$, пронумеровав их числами от 1

до l . Если при некотором k , $1 \leq k \leq l$, элемент $\theta_{pq}^k = 1$, $1 \leq p \leq n$, $1 \leq q \leq M$, то будем считать, что требование $i = p$ непрерывно обслуживается прибором $L = q$ во временном интервале с номером k . Нетрудно видеть, что при этом выполняются все требования, предъявляемые к расписанию, и $\bar{t}_{\max}(s^*) = T$.

Отметим, что если все t_{iL} — целые числа, то в разложении (3.2) все δ_k — целые числа и прерывания в расписании s^* имеют место только в целочисленные моменты времени.

Для нахождения разложения (3.2) можно воспользоваться следующим очевидным приемом. Выделим в матрице A множество $R(A)$ ее ненулевых элементов таким образом, чтобы в каждой строке и каждом столбце находился ровно один элемент этого множества. Поскольку матрица A кратна дважды стохастической матрице, то множество $R(A)$ существует.

Пусть δ — наименьший из элементов множества $R(A)$, а $\Pi = \|\theta_{pq}\|$ — перестановочная матрица, у которой $\theta_{pq} = 1$ тогда и только тогда, когда $a_{pq} \in R(A)$. Положим $\delta_1 = \delta$ и $\Pi_1 = \Pi$. Уменьшим в матрице A все ее элементы $a_{pq} \in R(A)$ на величину δ . Обозначим полученную в результате мат-

рицу через A . Эта матрица по-прежнему кратна дважды стохастической матрице. Проводя относительно нее те же рассуждения, получаем второй член $\delta_2 P_2$ разложения (3.2) и т.д. Поскольку на каждом шаге хотя бы один из ненулевых элементов матрицы A обращается в нуль, то через конечное число шагов (порядка $O(nM)$) будет получена нулевая матрица и процесс нахождения разложения (3.2) будет завершен.

Существуют различные способы выделения множества $R(A)$ по заданной матрице A . Так, множество $R(A)$ может быть выделено в результате решения задачи о назначениях (на узкие места) со стоимостной матрицей A либо в результате решения задачи о максимальном паросочетании в соответствующем двудольном графе и т.п. В любом случае для выделения множества $R(A)$ достаточно полиномиально ограниченного (относительно n и M) числа операций.

3.5. В условиях примера п.3.3 в качестве матрицы A можно выбрать, например, матрицу

$$A = \left\| \begin{array}{ccc|cccc} 4 & 3 & 2 & 1 & 0 & 0 & 0 \\ 3 & 5 & 0 & 0 & 2 & 0 & 0 \\ 0 & 1 & 3 & 0 & 0 & 6 & 0 \\ 3 & 0 & 4 & 0 & 0 & 0 & 3 \\ \hline 0 & 0 & 0 & 9 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 7 & 2 & 0 \\ 0 & 0 & 1 & 0 & 0 & 2 & 7 \end{array} \right\|.$$

Одним из возможных разложений вида (3.2) этой матрицы является разложение

$$A = 2 \cdot P_1 + 3 \cdot P_2 + 2 \cdot P_3 + 1 \cdot P_4 + 1 \cdot P_5 + 1 \cdot P_6,$$

где

$$\begin{aligned} \theta_{11}^1 &= \theta_{22}^1 = \theta_{33}^1 = \theta_{47}^1 = \theta_{54}^1 = \theta_{65}^1 = \theta_{76}^1 = 1, \\ \theta_{12}^2 &= \theta_{21}^2 = \theta_{36}^2 = \theta_{43}^2 = \theta_{54}^2 = \theta_{65}^2 = \theta_{77}^2 = 1, \\ \theta_{13}^3 &= \theta_{22}^3 = \theta_{36}^3 = \theta_{41}^3 = \theta_{54}^3 = \theta_{65}^3 = \theta_{77}^3 = 1, \\ \theta_{11}^4 &= \theta_{25}^4 = \theta_{32}^4 = \theta_{43}^4 = \theta_{54}^4 = \theta_{66}^4 = \theta_{77}^4 = 1, \\ \theta_{14}^5 &= \theta_{22}^5 = \theta_{33}^5 = \theta_{41}^5 = \theta_{55}^5 = \theta_{66}^5 = \theta_{77}^5 = 1, \\ \theta_{11}^6 &= \theta_{25}^6 = \theta_{36}^6 = \theta_{47}^6 = \theta_{54}^6 = \theta_{62}^6 = \theta_{73}^6 = 1, \end{aligned}$$

остальные $\theta_{pq}^k = 0$, $1 \leq k \leq 6$, $1 \leq p \leq 7$, $1 \leq q \leq 7$.

Расписание, соответствующее этому разложению, приведено на рис. 3.3.

3.6. Рассмотренная ситуация допускает естественное обобщение. Пусть имеются множество $N = \{1, 2, \dots, n\}$ требований и множество $\mathcal{M} = \{1, 2, \dots, M\}$ обслуживающих приборов. Процесс обслуживания каждого требования включает m стадий. Порядок прохождения стадий заранее не фиксирован и может быть различным для разных требований. На стадии j , $1 \leq j \leq m$, для обслуживания требования i , $i \in N$, могут использоваться приборы множества $\mathcal{M}_{ij} \subseteq \mathcal{M}$. Если $L \in \mathcal{M}_{ij}$ и требование i на стадии j об-

служивается только прибором L , то длительность его обслуживания должна равняться $t_{ijL} > 0$. Величины t_{ijL} предполагаются заданными. В процессе обслуживания допускаются прерывания. Если суммарная длительность обслуживания требования i на стадии j прибором $L \in \mathcal{M}_{ij}$ равна $\tilde{t}_{ijL} < t_{ijL}$, то для обслуживания требования i используется хотя бы один из приборов множества $\mathcal{M}_{ij}$, отличный от L . При этом должно выполняться

соотношение $\sum_{L \in \mathcal{M}_{ij}} \tilde{t}_{ijL} / t_{ijL} = 1$.

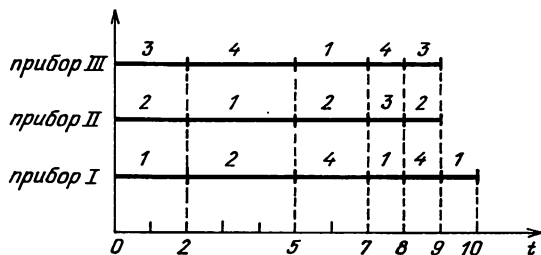


Рис. 3.3

По-прежнему, предполагается, что одновременно каждый прибор $L \in \mathcal{M}$ обслуживает не более одного требования $i \in N$ и каждое требование обслуживается не более, чем одним прибором.

Если положить $m = M, j = L, \mathcal{M}_{ij} = \{L\}, t_{ijL} = t_{iL}$, то получим обслуживающую систему, описанную в п. 3.1. Если положить $m = 1, \mathcal{M}_{i1} = \mathcal{M}, t_{i1L} = t_{iL}$, то получаем одностадийную систему с M параллельными приборами.

Для построения оптимального (по быстродействию) расписания в рассматриваемом случае можно воспользоваться следующим подходом.

Суммарные длительности обслуживания требований $\tilde{t}_{ijL} \geq 0$ в интервале планирования $(0, T_0]$, где T_0 — достаточно большое число, должны удовлетворять следующим ограничениям:

$$\sum_{L \in \mathcal{M}_{ij}} \frac{\tilde{t}_{ijL}}{t_{ijL}} = 1, \quad i = 1, \dots, n, \quad j = 1, \dots, m, \quad (3.3)$$

$$\sum_{i \in N} \sum_{j=1}^m \tilde{t}_{ijL} \leq T_0, \quad L \in \mathcal{M}, \quad (3.4)$$

$$\sum_{j=1}^m \sum_{L \in \mathcal{M}_{ij}} \tilde{t}_{ijL} \leq T_0, \quad i \in N. \quad (3.5)$$

Обозначим через $\tilde{t}_{ijL}^*$ решение задачи линейного программирования, состоящей в минимизации величины T_0 при указанных ограничениях.

Положим $t_{iL}^* = \sum_{j=1}^m \tilde{t}_{ijL}^*, i = 1, \dots, n, L = 1, \dots, M$. Интерпретируя величину t_{iL}^* как длительность обслуживания требования $i \in N$ прибором $L \in \mathcal{M}$ в системе с M приборами и нефиксированным порядком их прохо-

дения, приходим к рассмотренной выше ситуации (см. п. 3.1). Используя один из описанных в пп. 3.2 и 3.4 методов, получаем искомое оптимальное расписание s^* .

Для нахождения величин t_{iL}^* (решение задачи линейного программирования) и последующего построения расписания s^* (с прерываниями) для системы с нефиксированным порядком прохождения приборов требуется не более *полиномиального* (относительно n , m и M) числа операций.

Трудоемкость построения оптимального расписания может быть существенно уменьшена при некоторых дополнительных предположениях относительно обслуживающей системы.

Пусть, например, множество $\mathcal{M}$ приборов разбито на m , $m \leq M$, непустых попарно непересекающихся подмножеств $\mathcal{M}_j$, $j = 1, \dots, m$ и $\mathcal{M}_{ij} = \mathcal{M}_j$, $i \in N$. Все приборы каждого множества $\mathcal{M}_j$ предполагаются *идентичными*, т.е. $t_{ijL} = t_{ij}$, $L \in \mathcal{M}_j$.

Поскольку каждый прибор L используется только на одной из стадий, то можно положить $\tilde{t}_{ijL} = \tilde{t}_{iL}$ и записать соотношения (3.3)–(3.5) в виде

$$\sum_{L \in \mathcal{M}_j} \tilde{t}_{iL} = t_{ij}, \quad i = 1, \dots, n, \quad j = 1, \dots, m, \quad (3.6)$$

$$\sum_{i \in N} \tilde{t}_{iL} \leq T_0, \quad L \in \mathcal{M}_j, \quad j = 1, \dots, m, \quad (3.7)$$

$$\sum_{j=1}^m \sum_{L \in \mathcal{M}_j} \tilde{t}_{iL} \leq T_0, \quad i = 1, \dots, n. \quad (3.8)$$

Суммируя неравенства (3.7) по всем $L \in \mathcal{M}_j$ при фиксированном, j , $1 \leq j \leq m$, и учитывая (3.6), приходим к заключению, что наименьшее значение T_0 при указанных ограничениях равно

$$T_0^* = \max \left\{ \max_{i \in N} \sum_{j=1}^m t_{ij}, \max_{1 \leq j \leq m} \frac{1}{|\mathcal{M}_j|} \sum_{i \in N} t_{ij} \right\}.$$

Чтобы найти значения t_{iL}^* , можно поступить следующим образом. Для каждого j , $1 \leq j \leq m$, построим оптимальное (по быстродействию) расписание s_j^* обслуживания множества N требований параллельными идентичными приборами множества $\mathcal{M}_j$, разрешая прерывание процесса обслуживания любого требования $i \in N$ любым прибором $L \in \mathcal{M}_j$. Как известно, *построение такого расписания требует не более чем $O(n)$ операций*, причем общее

время обслуживания требований при расписании s_j^* равно $\max_{i \in N} \{ \max_{L \in \mathcal{M}_j} t_{iL} \}$.

В качестве $\tilde{t}_{iL}^*$ выберем суммарное время обслуживания требования i прибором $L \in \mathcal{M}_j$ при расписании s_j^* . Поскольку каждый прибор L принадлежит только одному из множеств $\mathcal{M}_j$, $1 \leq j \leq m$, можно положить $t_{iL}^* = \tilde{t}_{iL}^*$.

3.7. В качестве примера рассмотрим задачу построения оптимального по быстродействию расписания двухстадийного обслуживания четырех требований тремя приборами A , B и C . На стадии I используются при-

Таблица 3.3

i	1	2	3	4
I	2	3	4	3
II	3	1	1	4

Таблица 3.4

i	1	2	3	4
A	2	3	1	0
B	0	0	3	3
C	3	1	1	4

боры A и B , на стадии II – прибор C . Приборы A и B идентичны. Порядок прохождения стадий произвольный. Допускаются прерывания. Длительности обслуживания требований на каждой стадии приведены в табл. 3.3.

В рассматриваемом случае значение $T_0^* = 9$.

Оптимальное по быстродействию расписание обслуживания требований параллельными идентичными приборами A и B приведено на рис. 3.4.

Значения t_{iL}^* , $i = 1, 2, 3, 4$, $L \in \{A, B, C\}$ приведены в табл. 3.4.

Используя любой из алгоритмов, описанных в пп. 3.2 и 3.4, построим оптимальное (по быстродействию) расписание обслуживания четырех

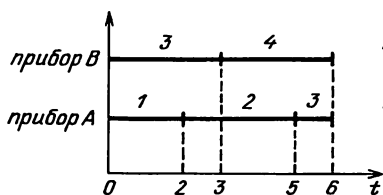


Рис. 3.4

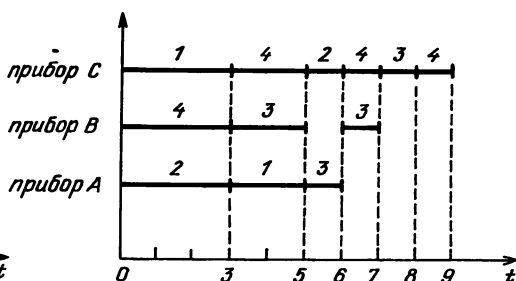


Рис. 3.5

требований в системе с тремя приборами A , B и C , нефиксированным порядком их прохождения и длительностями обслуживания, приведенными в табл. 3.4. Это расписание (рис. 3.5) и является искомым оптимальным расписанием.

3.8. Рассмотрим следующую задачу построения оптимального по быстродействию расписания.

В систему, состоящую из двух приборов A и B , в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Длительность обслуживания требования $i \in N$ прибором A равна $a_i \geq 0$ единицам времени.

и прибором B равна $b_i \geq 0$ единицам времени. Запись $a_i = 0$ или $b_i = 0$ означает, что требование не обслуживается приборами A или B соответственно.

Множество N состоит из трех попарно непересекающихся подмножеств N_{AB} , N_{BA} и N_O требований. Требования множества N_{AB} обслуживаются сначала прибором A , затем прибором B (маршрут обслуживания (A, B)); маршрутом обслуживания требований множества N_{BA} является (B, A) ; требования множества N_O могут обслуживаться в соответствии с любым из маршрутов (A, B) или (B, A) , причем маршруты заранее не заданы и могут быть различными для разных требований. Ко множеству N_O отнесены также все требования, для которых либо $a_i = 0$, либо $b_i = 0$.

В любой момент времени каждый прибор обслуживает не более одного требования и каждое требование обслуживается не более чем одним прибором. Разрешены прерывания процесса обслуживания любого требования любым прибором.

Требуется построить оптимальное (по быстродействию) расписание s^* обслуживания требований множества N .

Отметим, что в случае запрещения прерываний имеем ситуацию, рассмотренную в п. 1.4 этой главы. Далее в этом пункте обозначения и понятия, введенные в п. 1.4, употребляются без специальной оговорки.

В рассматриваемом случае для любого расписания s (с прерываниями) справедливо очевидное неравенство

$$\begin{aligned} \bar{t}_{\max}(s) &\geq \max\{a(N), b(N), \bar{t}_{\max}(s^*_{AB}), \\ &\bar{t}_{\max}(s^*_{BA}), \max\{a_i + b_i \mid i \in N_O\}\}. \end{aligned} \quad (3.9)$$

Это означает, что пары наборов I–V, перечисленные в п. 1.4, определяют оптимальное расписание s^* с прерываниями в тех же случаях, что и без прерываний.

Рассмотрим пару наборов

$$\begin{aligned} &(A; [0, m]; [a'_m, (\pi(N_{AB}), \pi(N_{BA} \cup N_O \setminus \{m\}))]; [t^0, m]); \\ &(B; [0, \pi(N_{BA})]; [\max\{a'_m, b(N_{BA})\}, m]; \\ &[t^0, \pi(N_{AB} \cup N_O \setminus \{m\})]), \end{aligned}$$

где $a'_m = a_m - b(N_{AB} \cup N_O \setminus \{m\})$ и $t^0 = \max\{a'_m + a(N \setminus \{m\}), a'_m + b_m, b(N_{BA} \cup \{m\})\}$.

Покажем, что эта пара наборов определяет оптимальное расписание s^* (с прерываниями) в случае, когда выполняются неравенства (1.10), (1.12), (1.21) и (1.23) (см. рис. 1.4 в п. 1.4 этой главы).

Действительно, если $t^0 = a'_m + b_m$, то $a'_m \geq b(N_{BA})$ и $b_m \geq a(N \setminus \{m\})$; тогда $\max\{\bar{t}_{iB}(s^*) \mid i \in N_{BA}\} = b(N_{BA}) \leq a'_m \leq \min\{t^0_{iA}(s^*) \mid i \in N_{BA}\}$ и $\max\{\bar{t}_{iA}(s^*) \mid i \in N_{AB} \cup N_O \setminus \{m\}\} = a'_m + a(N \setminus \{m\}) \leq a'_m + b_m = \min\{t^0_{iB}(s^*) \mid i \in N_{AB} \cup N_O \setminus \{m\}\}$. Требование m обслуживается прибором B во временном интервале $(a'_m, a'_m + b_m]$ и прибором A в интервалах $(0, a'_m]$ и $(a'_m + b_m, a_m + b_m]$. Кроме того, $\bar{t}_{\max}(s^*) = a_m + b_m$. Расписание s^* изображено на рис. 3.6, а.

Если $t^0 = b(N_{BA} \cup \{m\})$, то $b(N_{BA}) \geq a'_m$. В силу (1.10) и (1.23) имеем $\min\{t^0_{iA}(s^*) \mid i \in N_{BA}\} = a'_m + a(N_{AB}) > a(N_{AB}) \geq b(N_{BA}) =$

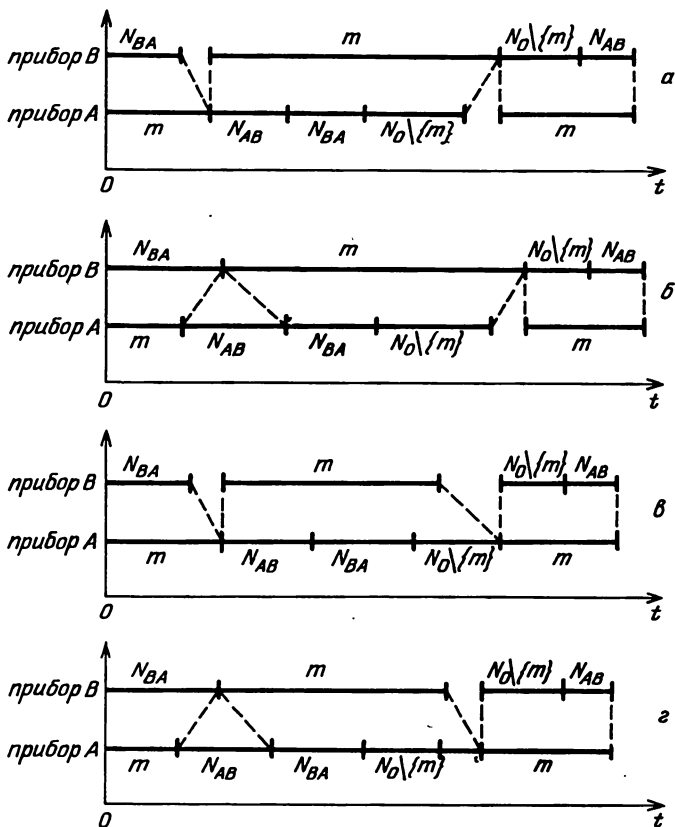


Рис. 3.6

$= \max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\}$. Далее, $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\} = a'_m + a(N \setminus \{m\}) \leq t^0 = \min\{t_{iB}^0(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\}$. Требование m обслуживается прибором B во временном интервале $(b(N_{BA}), b(N_{BA} \cup \{m\}))$ и прибором A в интервалах $(0, a'_m]$ и $(b(N_{BA} \cup \{m\}), b(N))$. Кроме того, $\bar{t}_{\max}(s^*) = b(N)$. Расписание s^* изображено на рис. 3.6 б.

Если $t^0 = a'_m + a(N \setminus \{m\})$, то в силу (1.10) и (1.23) имеем $\min\{t_{iA}^0(s^*) | i \in N_{BA}\} = a'_m + a(N_{BA}) > a(N_{AB}) \geq b(N_{BA}) = \max\{\bar{t}_{iB}(s^*) | i \in N_{BA}\}$. Далее, $\max\{\bar{t}_{iA}(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\} = t^0 = \min\{t_{iB}^0(s^*) | i \in N_{AB} \cup N_O \setminus \{m\}\}$. Требование m обслуживается прибором B во временном интервале $(\max\{a'_m, b(N_{BA})\}, t^0)$ и прибором A в интервалах $(0, a'_m]$ и $(t^0, a(N))$. Кроме того, $\bar{t}_{\max}(s^*) = a(N)$. Расписание s^* изображено на рис. 3.6 в, г.

Трудоемкость построения этого расписания не превосходит $O(n)$ операций.

Если не имеет места неравенство (1.10), то, как и в п. 1.4 § 1, можно поменять ролями приборы.

Учитывая трудоемкость построения расписаний, которые определяются парами наборов I—V, заключаем, что в рассматриваемом случае оптимальное расписание s^* (с прерываниями) может быть построено не более чем за $O(n + \tilde{n} \log \tilde{n})$ операций, где $\tilde{n} = \max \{|N_{AB}|, |N_{BA}|\}$.

Отметим также, что в случаях, когда $N_{AB} = \phi$ или $N_{BA} = \phi$, разрешение прерываний не уменьшает общего времени обслуживания.

3.9. Рассмотрим задачу построения оптимального (по быстродействию) расписания обслуживания множества N требований M приборами при следующем дополнительном ограничении на процесс обслуживания: если начат процесс обслуживания требования прибором, то это требование не может обслуживаться никаким другим прибором до завершения его обслуживания (возможно, с прерываниями) данным прибором.

Покажем, что рассматриваемая задача является *NP-трудной* при $M = 3$. Этой задаче соответствует следующая *задача распознавания*: определить, существует ли такое расписание s^0 обслуживания требований множества N , что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y . Покажем, что к ней сводится задача о разбиении (см. вводную часть этой главы).

Положим $n = n_0 + 3$; $t_{i1} = t_{i2} = e_i$, $t_{i3} = 0$, $i \in N^0$; $t_{n_0+1,1} = 3E$, $t_{n_0+1,2} = 0$, $t_{n_0+1,3} = 3E$; $t_{n_0+2,1} = E$, $t_{n_0+2,2} = 4E$, $t_{n_0+2,3} = E$; $t_{n_0+3,1} = 0$, $t_{n_0+3,2} = 0$, $t_{n_0+3,3} = 2E$. Покажем, что в построенной задаче расписание s^0 такое, что $\bar{t}_{\max}(s^0) \leq y = 6E$, существует тогда и только тогда, когда имеет решение задача о разбиении.

Если существует указанное разбиение множества N^0 на подмножества N_1^0 и N_2^0 , то искомое расписание s^0 можно построить, например, следующим образом. Каждый прибор, начиная с момента времени $d = 0$, обслуживает требования без прерываний процесса обслуживания и непрерывно одно за другим; прибор 1 обслуживает требования в последовательности $(n_0 + 1, \pi_{N_1^0}, n_0 + 2)$, прибор 2 — в последовательности $(\pi_{N_1^0}, n_0 + 2, \pi_{N_2^0})$, прибор 3 — в последовательности $(n_0 + 2, n_0 + 3, n_0 + 1)$.

Пусть существует расписание s^0 . Поскольку $t_{n_0+1}(M) = t_{n_0+2}(M) = y$, то в силу (3.1) значение $\bar{t}_{\max}(s^0) = 6E$, причем требования $n_0 + 1$ и $n_0 + 2$ в любой момент времени $t \in (0, 6E]$ обслуживаются одним из приборов. Отсюда следует, что в процессе обслуживания этих требований любым прибором не может быть прерываний. Поскольку $t_1(N) = t_2(N) = t_3(N) = 6E$, $N = \{1, 2, \dots, n_0 + 3\}$, то во временном интервале $(0, 6E]$ каждый прибор функционирует без простоев.

Предположим, что при расписании s^0 значение $t_{n_0+2,2}^0 = 0$. Тогда требование $n_0 + 1$ не может обслуживаться прибором 1 либо прибором 3 во временном интервале $(3E, 6E]$, что говорит о невозможности сделанного предположения. Аналогично убеждаемся в невозможности предположения, что $t_{n_0+2,2}^0 = 2E$. Следовательно, требование $n_0 + 2$ обслуживается прибором 2 во временном интервале $(E, 5E]$ и требования множества N^0 обслуживаются этим прибором в интервалах $(0, E]$ и $(5E, 6E]$.

Предположим, что при расписании s^0 обслуживание хотя бы одного из требований множества N^0 прибором 2 начато, но не закончено в интервале $(0, E]$. Тогда во временном интервале $(E, 5E]$ имеет место простой прибора 1, в то время как при расписании s^0 все приборы функционируют без простоев. Отсюда следует, что все требования множества N^0 , обслуживание которых прибором 2 начато в интервале $(0, E]$, должны быть полностью обслужены этим прибором в указанном интервале, причем суммарная длительность обслуживания этих требований должна быть равна E . Обозначая через N_1^0 подмножество всех требований множества N^0 , обслуживаемых прибором 2 в интервале $(0, E]$, и полагая $N_2^0 = N^0 \setminus N_1^0$, получаем решение задачи о разбиении.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о разбиении к рассматриваемой задаче распознавания.

3.10. В заключение этого параграфа рассмотрим задачу построения оптимального (по быстродействию) расписания s^* обслуживания множества $N = \{1, 2, \dots, n\}$ в системе, состоящей из M приборов, в предположении, что в процессе обслуживания каждого требования не допускаются задержки. Покажем, что *эта задача остается NP-трудной при $M \geq 2$, даже если разрешены прерывания процесса обслуживания каждого требования каждым прибором.*

Сформулируем соответствующую *задачу распознавания*: определить, существует ли расписание s^0 (с прерываниями), при котором процесс обслуживания каждого требования протекает без задержек и $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что к этой задаче при $M = 2$ сводится задача о разбиении. Положим $n = n_0 + 2$, $N = \{1, 2, \dots, n\}$. Пусть

$$t_{i1} = 0, \quad t_{i2} = e_i, \quad i \in N^0;$$

$$t_{n_0+1,1} = t_{n_0+2,1} = 2E;$$

$$t_{n_0+1,2} = t_{n_0+2,2} = E.$$

Покажем, что в построенной задаче расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = 4E$, существует тогда и только тогда, когда имеет решение задача о разбиении.

Пусть задача о разбиении имеет решение и N_1^0, N_2^0 — найденные подмножества множества N^0 . Тогда расписание s^0 существует и может быть построено, например, следующим образом. Оба прибора функционируют в интервале $(0, 4E]$ без простоев, процесс обслуживания каждого требования множества N протекает без прерываний. Прибор 1 обслуживает требования в последовательности $(n_0 + 1, n_0 + 2)$, а прибор 2 — в последовательности $(\pi_{N_1^0}, n_0 + 2, n_0 + 1, \pi_{N_2^0})$.

Пусть существует расписание s^0 . Так как $t_1(N) = t_2(N) = 4E$, то $\bar{t}_{\max}(s^0) = y$ и при расписании s^0 оба прибора функционируют без простоев в интервале $(0, 4E]$. Поскольку прибор 1 обслуживает только требования $n_0 + 1$ и $n_0 + 2$ и эти требования не различаются по длительностям обслуживания, будем считать, что при расписании s^0 значение $t_{n_0+1,1}^0 = 0$ (доказательство в случае, когда $t_{n_0+2,1}^0 = 0$, аналогичное).

Так как требование $n_0 + 1$ должно обслуживаться без задержек, то процесс его обслуживания протекает в интервале $(0, 3E]$. Поскольку прибор 1 функционирует без простоев и $t_{n_0+1,1} = 2E$, то в интервале $(0, 3E]$ прибор 1 обслуживает требование $n_0 + 2$ в течение ровно E единиц времени.

Далее, поскольку $t_{n_0+1,2} = t_{n_0+2,2} = E$, требование $n_0 + 2$ может обслуживаться прибором 2 в течение $E' \leq E$ единиц времени в интервале $(0, 3E]$.

Покажем, что $E' = E$. Действительно, в противном случае в интервале $(3E, 4E]$ прибор 1 обслуживает только требование $n_0 + 2$ и это же требование в этом же интервале должно обслуживаться в течение $E - E' > 0$ единиц времени прибором 2.

Таким образом, прибор 2 в интервале $(0, 3E]$ обслуживает требования $n_0 + 1$ и $n_0 + 2$ в течение E единиц времени каждое. Следовательно, требования множества N^0 должны обслуживаться прибором 2 в течение E единиц времени в интервале $(0, 3E]$ и в течение E единиц времени в интервале $(3E, 4E]$. Это означает, что задача о разбиении имеет решение.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о разбиении к рассматриваемой задаче распознавания.

§ 4. Общее время обслуживания. Упорядоченность

В этом параграфе рассматривается задача построения оптимального по быстродействию расписания обслуживания требований в системе с нефиксированными маршрутами в предположении, что на множестве требований задано отношение строгого порядка, регламентирующее (в том или ином смысле) возможную последовательность их обслуживания.

4.1. Пусть на множестве $N = \{1, 2, \dots, n\}$ требований задано отношение $\rightarrow$ строгого порядка и G — граф редукции этого отношения. Запись $i \rightarrow j$ (" i предшествует j "), $i, j \in N$, содержательно означает, что обслуживание требования j любым прибором $L \in \mathcal{M} = \{1, 2, \dots, M\}$ не может быть начато, пока не завершится процесс обслуживания требования i всеми приборами.

Если G — *цепь* (т.е. N — линейно упорядоченное множество), то построение оптимального расписания (как с прерываниями, так и без прерываний) не вызывает особых затруднений: первое по порядку требование должно быть обслужено всеми приборами (в произвольной последовательности), после завершения его обслуживания должно быть начато обслуживание второго по порядку требования и т.д.

Покажем, что *если все компоненты связности графа G — одновершинные графы, за исключением одной, являющейся цепью, то задача построения оптимального расписания (без прерываний) является NP-трудной в сильном смысле при $M \geq 2$.*

Этой задаче соответствует следующая *задача распознавания*: определить, существует ли такое расписание s^0 , при котором общее время обслуживания $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y .

Покажем, что при $M = 2$ к этой задаче сводится задача о 3-разбиении (см. вводную часть этой главы).

Положим $n = 5n_0$, $N = \{1, 2, \dots, n\}$. Множество N требований обслуживается двумя приборами A и B , длительности обслуживания требова-

ния $i \in N$ приборами A и B обозначим через a_i и b_i соответственно. Положим

$$a_i = b_i = e_i, \quad i \in N^0;$$

$$a_i = E, b_i = 0, \quad i = 3n_0 + 1, 3n_0 + 3, \dots, 5n_0 - 1;$$

$$a_i = 0, b_i = E, \quad i = 3n_0 + 2, 3n_0 + 4, \dots, 5n_0.$$

Зададим на множестве N отношение $\rightarrow$, полагая, что i непосредственно предшествует j , $i, j \in N$, тогда и только тогда, когда $j = i + 1, i = 3n_0 + 1, \dots, 5n_0 - 1$.

Покажем, что в построенной задаче расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = 2n_0E$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Если существует указанное разбиение множества N^0 на подмножества N_j^0 , $1 \leq j \leq n_0$, то искомое расписание s^0 можно построить, например, следующим образом. Каждый прибор, начиная с момента времени $d = 0$, обслуживает требования непрерывно одно за другим; прибор A обслуживает требования в последовательности $(3n_0 + 1, \pi_{N_1^0}, 3n_0 + 3, \pi_{N_2^0}, \dots, 3n_0 + 2j - 1, \pi_{N_j^0}, \dots, 5n_0 - 1, \pi_{N_{n_0}^0})$, а прибор B — в последовательности $(\pi_{N_1^0}, 3n_0 + 2, \pi_{N_2^0}, 3n_0 + 4, \dots, \pi_{N_j^0}, 3n_0 + 2j, \dots, \pi_{N_{n_0}^0}, 5n_0)$.

Пусть существует расписание s^0 . Поскольку $a(N) = b(N) = y$, то $\bar{t}_{\max}(s^0) = y$, причем оба прибора во временном интервале $(0, y]$ функционируют без простоев.

Так как множество $N \setminus N^0$ требований линейно упорядочено, то

$$\bar{t}_{5n_0, B} \geq \sum_{j=1}^{n_0} a_{3n_0+2j-1} + \sum_{j=1}^{n_0} b_{3n_0+2j} = 2n_0E = y.$$

Отсюда следует, что $\bar{t}_{n_0, B} = 2n_0E$ и $t_{5n_0, B}^0 = (2n_0 - 1)E$. Далее,

$$\bar{t}_{5n_0-1, A} \geq \sum_{j=1}^{n_0} a_{3n_0+2j-1} + \sum_{j=1}^{n_0-1} b_{3n_0+2j} = (2n_0 - 1)E$$

и

$$\bar{t}_{5n_0-1, A} \leq t_{5n_0, B}^0 = (2n_0 - 1)E.$$

Эти неравенства означают, что $\bar{t}_{5n_0-1, A} = (2n_0 - 1)E$ и $t_{5n_0-1, A}^0 = (2n_0 - 2)E$.

Рассуждая аналогичным образом, убеждаемся, что требование $3n_0 + 2j - 1$ обслуживается прибором A во временном интервале $((2j - 2)E, (2j - 1)E]$, $1 \leq j \leq n_0$, а требование $3n_0 + 2j$ обслуживается прибором B во временном интервале $((2j - 1)E, 2jE]$, $1 \leq j \leq n_0$. Требования множества N^0 должны обслуживаться прибором A во временных интервалах $(E, 2E], (3E, 4E], \dots, ((2n_0 - 1)E, 2n_0E]$, а прибором B — во временных интервалах $(0, E], (2E, 3E], \dots, ((2n_0 - 2)E, (2n_0 - 1)E]$. Поскольку $E/4 < e_i < E/2$, $i \in N^0$, и каждый из приборов A и B функционирует без простоев, то в каждом из указанных интервалов должно обслуживаться ровно три требования множества N^0 , суммар-

ная длительность обслуживания которых равна E . Обозначая через N_j^0 множество требований, обслуживаемых прибором A во временном интервале $((2j-2)E, (2j-1)E]$, $1 \leq j \leq n_0$, получаем решение задачи о 3-разбиении.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

4.2. Пусть все компоненты связности графа G — одновершинные графы, за исключением одной, являющейся цепью. Требование $i \in N$ поступает в систему в момент времени $d_i \geq 0$. Покажем, что оптимальное (по быстродействию) расписание с прерываниями может быть построено за полиномиальное число операций.

Пусть $N_0 \subset N$ — непустое множество требований, соответствующих изолированным вершинам графа G . Множество требований, соответствующих вершинам компоненты — цепи графа G , обозначим через N_1 . По условию $N_1 \cup N_0 = N$.

Пусть $|N_1| = k$. Пронумеруем требования таким образом, чтобы $N_1 = \{1, 2, \dots, k\}$, $N_0 = \{k+1, k+2, \dots, n\}$.

Вычислим наиболее ранние моменты начала и завершения обслуживания требований множества N_1 . Имеем

$$t_1^0 = d_1, \quad \bar{t}_1 = t_1^0 + \sum_{L \in \mathcal{M}} t_{1L};$$

$$t_i^0 = \max\{d_i, \bar{t}_{i-1}\}, \quad \bar{t}_i = t_i^0 + \sum_{L \in \mathcal{M}} t_{iL}, \quad i = 2, \dots, k.$$

Построим последовательность $\delta_1, \delta_2, \dots, \delta_{p+1}$ — монотонно возрастающую последовательность попарно различных значений $t_i^0, \bar{t}_i, i \in N_1$; и $d_i, i \in N_0$. Обозначим $\Delta_j = \delta_{j+1} - \delta_j, j = 1, \dots, p$.

Решим задачу линейного программирования

$$t \rightarrow \min,$$

$$\sum_{L \in \mathcal{M}} x_{iL}^j \leq \Delta_j, \quad i \in N, \quad j = 1, \dots, p;$$

$$\sum_{i \in N} x_{iL}^j \leq \Delta_j, \quad L \in \mathcal{M}, \quad j = 1, \dots, p;$$

$$y_{iL} \leq t, \quad i \in N_0, \quad L \in \mathcal{M};$$

$$\sum_{j=1}^p x_{iL}^j + y_{iL} = t_{iL}, \quad i \in N_0, \quad L \in \mathcal{M};$$

$$\sum_{j=1}^p x_{iL}^j = t_{iL}, \quad i \in N_1, \quad L \in \mathcal{M};$$

$$t \geq 0, \quad x_{iL}^j \geq 0, \quad j = 1, \dots, p, \quad L \in \mathcal{M}, \quad i \in N;$$

$$y_{iL} \geq 0, \quad i \in N_0, \quad L \in \mathcal{M};$$

$$x_{iL}^j = 0, \text{ если } t_i^0 > \delta_j \text{ или } \bar{t}_i < \delta_{j+1}, \quad i \in N_1, \quad L \in \mathcal{M};$$

$$x_{iL}^j = 0, \text{ если } d_i > \delta_j, \quad i \in N_0, \quad L \in \mathcal{M}.$$

Пусть $\hat{t}$ — найденное наименьшее значение t и $\hat{x}_{iL}^j, \hat{y}_{iL}$ — значения переменных, при которых $t = \hat{t}$.

Величину x_{iL}^j будем интерпретировать как суммарную длительность обслуживания требования $i \in N$ прибором $L \in \mathcal{M}$ во временном интервале $(\delta_j, \delta_{j+1}]$. Аналогично величину y_{iL} будем понимать как суммарную длительность обслуживания требования $i \in N_0$ прибором $L \in \mathcal{M}$ во временном интервале $(\delta_{p+1}, \delta_{p+1} + t]$.

Для построения расписания s^* достаточно при помощи одного из алгоритмов, приведенных в п.3.2, 3.4 § 3 этой главы, построить оптимальное (по быстродействию) расписание обслуживания требований множества N в каждом из интервалов $(\delta_j, \delta_{j+1}]$, $j = 1, \dots, p$, считая длительность обслуживания требования $i \in N$ прибором $L \in \mathcal{M}$ в интервале $(\delta_j, \delta_{j+1}]$ равной $\hat{x}_{iL}^j$. Если $\hat{t} > 0$, то следует также построить оптимальное расписание обслуживания требований множества N_0 в интервале $(\delta_{p+1}, \delta_{p+1} + \hat{t}]$, считая длительность обслуживания требования $i \in N_0$ прибором $L \in \mathcal{M}$ в этом интервале равной $\hat{y}_{iL}$.

Поскольку сформулированная задача линейного программирования может быть решена за полиномиальное число операций, трудоемкость алгоритмов из п.3.2 или 3.4 также полиномиальна и число решаемых ими подзадач не превосходит $p + 1 \leq 3n$, то задача построения s^* в рассматриваемом случае имеет полиномиальную трудоемкость.

4.3. Рассмотрим ситуацию, отличающуюся от описанной в предыдущем пункте только тем, что $M = 2$ и все $d_i = 0$, $i \in N$. Покажем, что оптимальное по быстродействию расписание (с прерываниями) в этом случае может быть построено за $O(n)$ операций.

Обслуживающие приборы по-прежнему будем обозначать через A и B , а длительности обслуживания требования $i \in N$ этими приборами — через a_i и b_i соответственно.

Пусть $N_0 \subset N$ — непустое множество требований, соответствующих изолированным вершинам графа G . Компоненту-цепь графа G обозначим через G_1 , а множество требований, соответствующих ее вершинам, — через N_1 . По условию $N_0 \cup N_1 = N$.

Положим

$$T = \max \{a(N), b(N), \sum_{j \in N_1} (a_j + b_j), \max_{i \in N_0} \{a_i + b_i\}\}.$$

Очевидно, для любого расписания s (с прерываниями) справедливо соотношение $\bar{t}_{\max}(s) \geq T$. Построим расписание s^* , при котором $\bar{t}_{\max}(s) = T$. Расписание s^* , очевидно, будет искомым оптимальным расписанием.

Пусть $|N_1| = k$. Пронумеруем требования таким образом, что $N_1 = \{1, 2, \dots, k\}$ и $N_0 = \{k+1, k+2, \dots, n\}$. Требование r будем обслуживать (без прерываний) прибором A во временном интервале

$$\left(\sum_{j=1}^{r-1} (a_j + b_j), \sum_{j=1}^{r-1} (a_j + b_j) + a_r \right] \text{ и прибором } B \text{ во временном интервале}$$

$$\left(\sum_{j=1}^{r-1} (a_j + b_j) + a_r, \sum_{j=1}^r (a_j + b_j) \right], r = 1, \dots, k.$$

Предположим, что построено расписание s_{q-1} обслуживания требований $1, 2, \dots, q-1$, $k < q \leq n$, при котором $\bar{t}_p = \max_{1 \leq i \leq q-1} \bar{t}_i \leq T$.

Построим расписание обслуживания требования q .

Если $T \geq \bar{t}_p + a_q + b_q$, то требование q будем обслуживать (без прерываний) прибором A во временном интервале $(\bar{t}_p, \bar{t}_p + a_q]$, а прибором B — в интервале $(\bar{t}_p + a_q, \bar{t}_p + a_q + b_q]$.

Если $T < \bar{t}_p + a_q + b_q$, то требование q не может быть полностью обслужено в интервале $(\bar{t}_p, T]$. Вычислим значение

$$t = \min \{ T - \max \{ \sum_{j=1}^q b_j - \bar{t}_p, 0 \}, \bar{t}_p + a_q \}.$$

Требование q будем обслуживать прибором A в течение $a_q - t + \bar{t}_p$ единиц времени в тех подынтервалах интервала $(0, \bar{t}_p]$, в течение которых прибор A простаивает, а также в течение $t - \bar{t}_p$ единиц времени (если $t > \bar{t}_p$) в интервале $(\bar{t}_p, t]$. Такое обслуживание возможно, поскольку суммарная длительность простоя прибора A в интервале $(0, \bar{t}_p]$ равна $\bar{t}_p - \sum_{j=1}^{q-1} a_j$ и выполняется неравенство $\sum_{j=1}^q a_j \leq t$, равносильное неравенству $a_q - t + \bar{t}_p \leq \bar{t}_p - \sum_{j=1}^{q-1} a_j$.

Требование q будем обслуживать прибором B в течение $\max \{ b_q - T + t, 0 \}$ единиц времени в тех подынтервалах интервала $(0, \bar{t}_p]$, в течение которых прибор B простаивает, а также в течение $T - t$ единиц времени (если $t < T$) в интервале $(t, T]$. Такое обслуживание возможно, поскольку суммарная длительность простоя прибора B в интервале $(0, \bar{t}_p]$ равна $\bar{t}_p - \sum_{j=1}^{q-1} b_j$ и выполняется неравенство $\sum_{j=1}^q b_j \leq \bar{t}_p + T - t$, равносильное неравенству $b_q - T + t \leq \bar{t}_p - \sum_{j=1}^{q-1} b_j$.

Выполняя описанные построения для каждого требования q , $q = k+1, \dots, n$, придем к расписанию $s^* = s_n^*$, при котором $\bar{t}_{\max}(s^*) = T$.

Таблица 4.1

i	1	2	3	4	5
a_i	2	3	2	1	4
b_i	3	2	1	3	6

Можно показать, что при использовании специальной структуры данных (стека) для хранения интервалов простоя каждого прибора расписание s^* будет построено за $O(n)$ операций.

4.4. П р и м е р. Две ремонтно-строительные бригады A и B разного профиля должны произвести ремонт пяти объектов; продолжительность ремонта объекта i бригадами A и B обозначим через a_i и b_i соответственно. Значения a_i и b_i приведены в табл. 4.1.

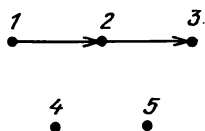


Рис. 4.1

Порядок, в котором бригады ведут работы на данном объекте, не имеет значения. На каждом объекте должна работать вся бригада, т.е. одновременная работа одной и той же бригады на различных объектах не допускается. Две бригады не могут одновременно работать на одном и том же объекте. Работы, проводимые бригадой на любом объекте, могут быть прерваны, а затем возобновлены.

Первые три объекта находятся на различных этажах одного и того же здания. Ремонтные работы на каждом из этих объектов, кроме первого, могут начаться лишь после того, как будет полностью отремонтирован предыдущий объект. Другими словами, на множестве объектов задано отношение строгого порядка, граф редукции которого изображен на рис 4.1.

Требуется построить такое расписание выполнения ремонтных работ, при котором все объекты были бы отремонтированы как можно раньше.

Воспользуемся алгоритмом из п. 4.3. В данном случае $N_1 = \{1, 2, 3\}$, $N_0 = \{4, 5\}$, $T = \max\{12, 15, 13, 10\} = 15$.

Расписание ремонта объектов множества N_1 изображено на рис. 4.2, а.

Построим расписание выполнения ремонта объекта 4. Поскольку $\bar{t}_p = \bar{t}_3 = 13$, то $\bar{t}_3 + a_4 + b_4 > T$. Вычислим $t = \min\{15 - \max\{9 - 13, 0\}, 13 + 1\}$ Расписание ремонта объектов $\{1, 2, 3, 4\}$ представлено на рис. 4.2, б.

Искомое расписание ремонта всех пяти объектов изображено на рис. 4.2, в.

4.5. Пусть каждая компонента связности графа G является цепью. Покажем, что в этом случае задача построения оптимального по быстройдействию расписания (с прерываниями) является NP-трудной в сильном смысле при $M \geq 2$.

Этой задаче соответствует следующая задача распознавания: определить, существует ли такое расписание s^0 обслуживания требований множества N , что $\bar{t}_{\max}(s^0) \leq u$ для заданного числа u . Покажем, что к последней задаче в случае $M = 2$ сводится задача о 3-разбиении.

Пусть $n = 7n_0$ и $N = \{1, 2, \dots, n_0\}$. Каждое требование $i \in N$ обслуживается двумя приборами A и B , длительности обслуживания равны a_i и

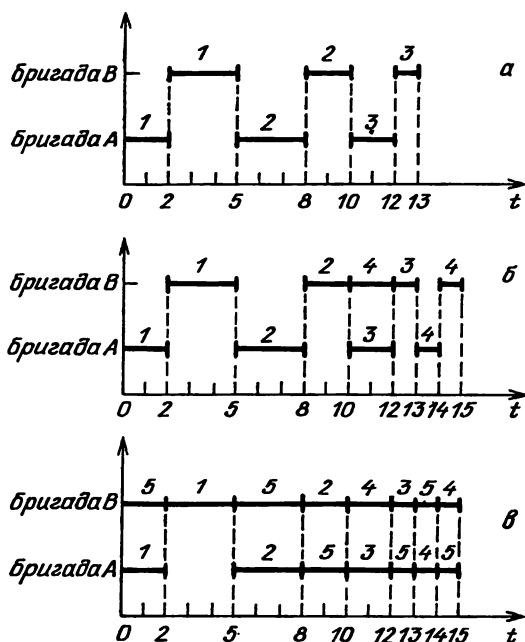


Рис. 4.2

b_j соответственно. Положим

$$a_j = b_j = E, \quad j = 1, \dots, n_0,$$

$$a_{n_0+k} = 0, \quad b_{n_0+k} = e_k, \quad k = 1, \dots, 3n_0,$$

$$a_{4n_0+k} = e_k, \quad b_{4n_0+k} = 0, \quad k = 1, \dots, 3n_0.$$

На множестве N зададим отношение строгого порядка, полагая, что i_1 непосредственно предшествует i_2 ($i_1, i_2 \in N$) тогда и только тогда, когда $i_1 = j$, $i_2 = j + 1$, $j = 1, \dots, n_0 - 1$, либо $i_1 = n_0 + k$, $i_2 = 4n_0 + k$, $k = 1, \dots, 3n_0$.

Покажем, что в построенной задаче расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = 2n_0E$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Пусть задача о 3-разбиении имеет решение и $N_j^0 = \{\alpha_{j1}, \alpha_{j2}, \alpha_{j3}\}$, $j = 1, \dots, n_0$, — найденные трехэлементные подмножества. Тогда искомое расписание s^0 можно построить следующим образом. Каждый прибор начиная с момента времени $d = 0$ обслуживает требования множества N непрерывно одно за другим, причем прибор A обслуживает требования в последовательности $1, 4n_0 + \alpha_{11}, 4n_0 + \alpha_{12}, 4n_0 + \alpha_{13}, 2, 4n_0 + \alpha_{21}, 4n_0 + \alpha_{22}, 4n_0 + \alpha_{23}, 3, \dots, n_0, 4n_0 + \alpha_{n_01}, 4n_0 + \alpha_{n_02}, 4n_0 + \alpha_{n_03}$, а прибор B — в последовательности $n_0 + \alpha_{11}, n_0 + \alpha_{12}, n_0 + \alpha_{13}, 1, n_0 + \alpha_{21}, n_0 + \alpha_{22}, n_0 + \alpha_{23}, 2, \dots, n_0 - 1, n_0 + \alpha_{n_01}, n_0 + \alpha_{n_02}, n_0 + \alpha_{n_03}, n_0$.

Пусть расписание s^0 существует. Покажем, что задача о 3-разбиении имеет решение.

Так как $a(N) = b(N) = y$, то $\bar{t}_{\max}(s^0) = y$ и при расписании s^0 оба прибора в интервале $(0, y]$ функционируют без простоев. Кроме того, поскольку $\sum_{j=1}^{n_0} (a_j + b_j) = y$, то при расписании s^0 для каждого требова-

ния j , $1 \leq j \leq n_0$, справедливы соотношения $t_j^0 = 2(j-1)E$ и $\bar{t}_j = 2jE$.

Рассмотрим временной интервал $(0, 2E]$, в котором оба прибора обслуживают требование 1. Кроме того, в этом интервале прибор B в течение E единиц времени обслуживает требования вида $n_0 + k$, $1 \leq k \leq 3n_0$, а прибор A также в течение E единиц времени обслуживает те требования вида $4n_0 + k$, $1 \leq k \leq 3n_0$, которым предшествуют (относительно заданного на N порядка) требования, уже обслуженные прибором B .

Прибор B в интервале $(0, 2E]$ не может полностью обслужить более трех требований вида $n_0 + k$, так как $e_k > E/4$, $k = 1, \dots, 3n_0$. Если суммарная длительность обслуживания прибором B тех требований, которые полностью обслужены в интервале $(0, 2E]$ этим прибором, меньше E , то в интервале $(0, 2E]$ прибор A будет простаивать, что невозможно. Отсюда следует, что при расписании s^0 не существует требований, обслуживание которых прибором B начато, но не закончено в интервале $(0, 2E]$.

Рассуждая аналогичным образом, нетрудно показать, что в каждом временном интервале $(2(j-1)E, 2jE]$, $1 \leq j \leq n_0$, каждый прибор A и B обслуживает кроме требования j еще три требования, суммарная длительность обслуживания которых равна E . Это означает, что задача о 3-разбиении имеет решение.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

4.6. Пусть на множестве $N = \{1, 2, \dots, n\}$ требований задано отношение $\rightarrow$ строгого порядка и G — граф редукции этого отношения. В отличие от предыдущих пунктов здесь и далее в этом параграфе запись $i \rightarrow j$ (" i предшествует j ") будем интерпретировать следующим образом: обслуживание требования j любым прибором $L \in \mathcal{M} = \{1, 2, \dots, M\}$ не может быть начато, пока не завершится процесс обслуживания требования i этим прибором.

Если G — цепь, то задача построения оптимального расписания (как с прерываниями, так и без прерываний) эквивалентна задаче построения оптимального расписания обслуживания неупорядоченного множества требований в системе поточного типа. Чтобы убедиться в этом, достаточно поменять ролями приборы и требования. Последняя задача рассматривалась в § 1 гл. 2, откуда, в частности, следует, что исходная задача при $M = 2$ может быть решена за $O(n \log n)$ операций (если не допускаются прерывания) и за $O(n^2)$ операций (если прерывания допускаются).

Покажем, что если все компоненты связности графа G — одновершинные графы, за исключением одной, являющейся цепью, то задача построения оптимального расписания (без прерываний) является NP-трудной в сильном смысле при $M \geq 2$.

Этой задаче соответствует следующая задача распознавания: определить, существует ли такое расписание s^0 обслуживания требований множества N , что $\bar{t}_{\max}(s^0) \leq y$ для заданного числа y . Покажем, что к последней задаче в случае $M = 2$ сводится задача о 3-разбиении.

Положим $n = 7n_0 + 1$ и $N = \{1, 2, \dots, n\}$. Обслуживающие приборы обозначим через A и B , а длительность обслуживания требования $i \in N$ этими приборами — через a_i и b_i соответственно. Пусть

$$C = [E/2],$$

$$a_i = b_i = e_i, \quad i \in N^0;$$

$$a_{3n_0+1} = 0, \quad b_{3n_0+1} = E - C;$$

$$a_{3n_0+i} = E - C, \quad b_{3n_0+i} = En_0 + C, \quad i = 4k - 2, \quad k = 1, \dots, n_0;$$

$$a_{3n_0+i+1} = En_0 + C, \quad b_{3n_0+i+1} = E - C, \quad i = 4k - 2, \quad k = 1, \dots, n_0;$$

$$a_{3n_0+i+2} = En_0 + E - C, \quad b_{3n_0+i+2} = C, \quad i = 4k - 2, \quad k = 1, \dots, n_0;$$

$$a_{3n_0+i+3} = C, \quad b_{3n_0+i+3} = En_0 + E - C, \quad i = 4k - 2, \quad k = 1, \dots, n-1;$$

$$a_{7n_0+1} = E, \quad b_{7n_0+1} = En_0 + E - C.$$

Зададим на множестве N отношение $\rightarrow$, полагая, что i непосредственно предшествует j тогда и только тогда, когда $j = i + 1, i = 3n_0 + 1, \dots, 7n_0$.

Покажем, что в построенной задаче расписание s^0 , при котором $\bar{t}_{\max}(s^0) \leq y = (2n_0^2 + 3n_0 + 1)E - C$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Пусть задача о 3-разбиении имеет решение и $N_j^0, 1 \leq j \leq n_0$, — найденные трехэлементные подмножества. Тогда расписание s^0 может быть построено следующим образом (рис. 4.3). Каждый из приборов A и B обслуживает требование непрерывно одно за другим начиная с момента времени $d = 0$. Прибор A обслуживает требование множества $N \setminus N^0$ в порядке возрастания их номеров; между требованиями с номерами $3n_0 + 4(j-1) + 3$ и $3n_0 + 4j$ обслуживаются требования множества $N_j^0, 1 \leq j \leq n_0$. Прибор B обслуживает требования множества $N \setminus N^0$ в порядке возрастания их номеров; между требованиями с номерами $3n_0 + 4j + 1$ и $3n_0 + 4j + 2$

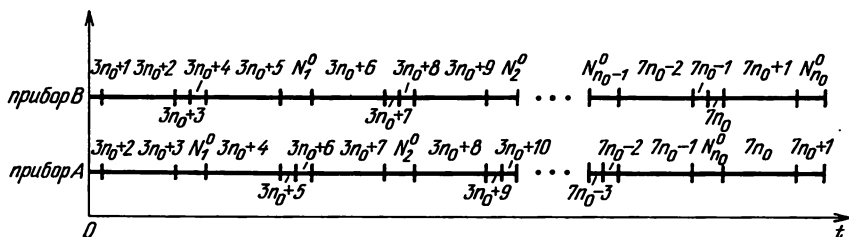


Рис. 4.3

обслуживаются требования множества $N_j^0, 1 \leq j \leq n_0 - 1$; требования множества $N_{n_0}^0$ обслуживаются вслед за требованием $7n_0 + 1$. Требования каждого из множеств N_j^0 могут обслуживаться каждым прибором в произвольной последовательности.

Пусть существует расписание s^0 . Поскольку $a(N) = b(N) = y$, то $\bar{t}_{\max}(s^0) = y$ и при расписании s^0 оба прибора функционируют без простоев.

Обозначим через N_{AB} множество требований из $N \setminus N^0$, которые при расписании s^0 обслуживаются сначала прибором A , затем прибором B . В аналогичном смысле будем упорядочивать обозначение N_{BA} .

Если предположить, что требование $7n_0 + 1$ принадлежит N_{AB} , то получим расписание, для которого общее время обслуживания оказывается не меньше чем

$$\sum_{i=3n_0+2}^{7n_0+1} a_i + b_{7n_0+1} = y + E - C.$$

Следовательно, требование $7n_0 + 1$ принадлежит N_{BA} . Требование $7n_0$ также принадлежит N_{BA} , поскольку в противном случае общее время обслуживания ограничено снизу числом

$$\sum_{i=3n_0+2}^{7n_0+1} a_i + b_{7n_0} + b_{7n_0+1} > y.$$

С другой стороны, требования $7n_0 - 1$ и $7n_0 - 2$ должны принадлежать множеству N_{AB} . Действительно, если $7n_0 - 1 \in N_{BA}$, то общее время обслуживания оказывается не меньше чем

$$\sum_{i=3n_0+1}^{7n_0-1} b_i + \sum_{i=7n_0-1}^{7n_0+1} a_i = y + E.$$

Если же $7n_0 - 2 \in N_{BA}$, то общее время обслуживания оказывается не меньше чем

$$\sum_{i=3n_0+1}^{7n_0-2} b_i + \sum_{i=7n_0-2}^{7n_0+1} a_i = y + E.$$

Рассуждая аналогичным образом, нетрудно убедиться, что $N_{AB} = \{3n_0 + 1, 3n_0 + 4(j-1) + 2, 3n_0 + 4(j-1) + 3 \mid j = 1, \dots, n_0\}$, а $N_{BA} = \{3n_0 + 4j, 3n_0 + 4j + 1 \mid j = 1, \dots, n_0\}$. При этом требования множества N^0 должны обслуживаться прибором A во временных интервалах $(E(n_0 + 1) + E(j-1)(2n_0 + 3), E(n_0 + 2) + E(j-1)(2n_0 + 3))$, $1 \leq j \leq n_0$, а прибором B — во временных интервалах $(Ej(2n_0 + 3) - C, Ej(2n_0 + 3) + E - C)$, $1 \leq j \leq n_0$. Поскольку оба прибора функционируют без простоев, длина каждого интервала равна E и выполняются неравенства $E/4 < e_i < E/2$, $i \in N^0$, то в каждом из указанных интервалов должно обслуживаться ровно три требования множества N^0 , суммарная длительность обслуживания которых равна E . Обозначая через N_j^0 множество требований, обслуживаемых прибором A во временном интервале $(E(n_0 + 1) + E(j-1)(2n_0 + 3), E(n_0 + 2) + E(j-1)(2n_0 + 3))$, $1 \leq j \leq n_0$, получаем решение задачи о 3-разбиении.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

4.7. Пусть каждая компонента связности графа G является либо цепью, либо одновершинным графом. Покажем, что при $M = 2$ оптимальное расписание s^* (с прерываниями) может быть построено не более чем за $O(n^2)$ операций.

Обслуживающие приборы по-прежнему будем обозначать через A и B , а длительности обслуживания требования $i \in N$ этими приборами через a_i и b_i соответственно.

Компоненту-цепь графа G обозначим через G_l , а множество требований, соответствующих ее вершинам, — через N_l , $l = 1, \dots, p$, $p \geq 1$. Пусть N_0 — множество (возможно, пустое) требований, соответствующих изолированным вершинам графа G .

Каждому множеству N_l , $1 \leq l \leq p$, сопоставим новое требование $\bar{l}$ (отличное от требований множества N), полагая $a_{\bar{l}} = a(N_l)$ и $b_{\bar{l}} = b(N_l)$.

При помощи алгоритма из п. 1.2 § 1 этой главы построим расписание s_1 обслуживания (без прерываний) множества $\{\bar{1}, \bar{2}, \dots, \bar{p}\} \cup N_0$ независимых требований. Если при этом

$$\bar{t}_{\max}(s_1) = \max\{a(N), b(N), \max\{a_i + b_i \mid i \in N_0\}\},$$

то расписание s_1 может быть преобразовано в расписание s^* путем замены каждого из требований $\bar{l}$ на линейно упорядоченное подмножество требований N_l , обслуживаемых соответствующим прибором непрерывно одно за другим. При этом, очевидно, $\bar{t}_{\max}(s^*) = \bar{t}_{\max}(s_1)$.

Если $\bar{t}_{\max}(s_1) = a_{\bar{q}} + b_{\bar{q}} = \max\{a_{\bar{l}} + b_{\bar{l}} \mid l = 1, \dots, p\}$, то при помощи алгоритма из § 1 гл. 2 построим расписание (с прерываниями) обслуживания линейно упорядоченного подмножества требований N_q . Обозначим это расписание через s_2 .

Пусть w_A и w_B означают суммарную длительность простоя прибора A и прибора B во временном интервале $(0, \bar{t}_{\max}(s_2)]$ при расписании s_2 . Заметим, что в указанном интервале оба прибора одновременно не простаивают. Очевидно, $\bar{t}_{\max}(s_2) = a_{\bar{q}} + w_A = b_{\bar{q}} + w_B$.

Введем в рассмотрение новое (отличное от ранее рассмотренных) требование z , полагая $a_z = a(N \setminus N_q)$, $b_z = b(N \setminus N_q)$.

Поскольку $\bar{t}_{\max}(s_1) > \max\{a(N), b(N)\}$, то $a_z \leq b_{\bar{q}}$ и $b_z \leq a_{\bar{q}}$.

Если $a_z \leq w_A$ и $b_z \leq w_B$, то требование z будем обслуживать (с прерываниями) в интервалах простоя прибора A и прибора B в течение a_z и b_z единиц времени соответственно. Полученное при этом расписание обозначим s_3 .

Пусть не выполняется хотя бы одно из неравенств $a_z \leq w_A$ и $b_z \leq w_B$.

Предположим, что $a(N) \geq b(N)$ (в противном случае можно провести аналогичные рассуждения). Отсюда следует, что $a_z > w_A$.

Требование z будем обслуживать (с прерываниями) прибором A в течение a_z единиц времени в интервале $(0, a(N)]$ и прибором B в течение $\min\{b_z, w_B\}$ единиц времени в интервале $(0, \bar{t}_{\max}(s_2)]$. Обслуживание будем проводить только в тех подынтервалах указанных интервалов, в которых соответствующий прибор не обслуживает требований из множества N_q .

Если при этом $w_B \geq b_z$, то получаем расписание s_3 обслуживания требований множества $N_q \cup \{z\}$.

Если $w_B < b_z$, то в интервале $(\bar{t}_{\max}(s_2), a(N)]$ длины $a_z - w_A$ прибор B простаивает. Так как $w_A = b_{\bar{q}} + w_B - a_{\bar{q}}$ и $a(N) \geq b(N)$, то $a_z - w_A \geq b_z - w_B$.

Пусть $\delta_1, \delta_2, \dots, \delta_{2r}$ — возрастающая последовательность моментов времени такая, что при расписании s_2 в каждом интервале $(\delta_{2j-1}, \delta_{2j}]$,

$1 \leq j \leq r$, оба прибора функционируют одновременно, причем

$$\sum_{j=1}^r (\delta_{2j} - \delta_{2j-1}) = a_{\overline{q}} - w_B.$$

Поскольку $b_z \leq a_{\overline{q}}$, то найдутся такой интервал $(\delta_{2k-1}, \delta_{2k}]$, $1 \leq k \leq r$, и такой момент времени $t^0 \in (\delta_{2k-1}, \delta_{2k}]$, что

$$\delta_{2k} - t^0 + \sum_{j=k+1}^r (\delta_{2j} - \delta_{2j-1}) = b_z - w_B.$$

Все требования, обслуживаемые прибором B при расписании s_2 в интервалах $(t^0, \delta_{2k}]$, $(\delta_{2k+1}, \delta_{2k+2}]$, $\dots$, $(\delta_{2r-1}, \delta_{2r}]$, будем обслуживать этим прибором в той же последовательности в интервале $(\bar{t}_{\max}(s_2), \bar{t}_{\max}(s_2) + b_z - w_B]$. При этом не исключено, что обслуживание одного из требований множества N_q прервется в момент времени t^0 и возобновится в момент времени $\bar{t}_{\max}(s_2)$.

В интервалах $(t_0, \delta_{2k}]$, $(\delta_{2k+1}, \delta_{2k+2}]$, $\dots$, $(\delta_{2r-1}, \delta_{2r}]$ будем обслуживать прибором B требование z . Полученное расписание обслуживания требований множества $N_q \cup \{z\}$ обозначим через s_3 .

Таким образом, в любом случае будет построено расписание s_3 обслуживания требований множества $N_q \cup \{z\}$. Это расписание может быть преобразовано в расписание s^* путем замены требования z на требования множеств N_l , $l = 0, \dots, p$, $l \neq q$, с сохранением линейного порядка на указанных множествах (если $l > 0$).

Очевидно,

$$\bar{t}_{\max}(s^*) = \bar{t}_{\max}(s_3) = \max\{\bar{t}_{\max}(s_2), a(N), b(N)\}$$

и расписание s^* является оптимальным.

Наиболее трудоемкой процедурой описанного алгоритма является построение расписания s_2 , требующее $O(|N_q|^2)$ операций. Тем самым, расписание s^* может быть построено в результате выполнения не более чем $O(n^2)$ операций.

4.8. П р и м е р. Несколько изменим условия примера из п.4.4. Число ремонтируемых объектов увеличим до восьми. Величины a_i и b_i приведены в табл. 4.2.

Каждая бригада может ремонтировать объект 6 только после того, как она отремонтировала объект 5. Аналогично каждая бригада может ремонтировать объекты 2, 3 и 4 только после того, как она отремонтировала объект с меньшим номером. Другими словами, на множестве

Т а б л и ц а 4.2

i	1	2	3	4	5	6	7	8
a_i	2	3	5	4	2	2	2	1
b_i	1	3	2	2	5	4	1	3

объектов задано отношение строгого порядка (в смысле п. 4.6), граф редукции которого изображен на рис. 4.4.

Требуется таким образом организовать ремонт объектов, чтобы все они были отремонтированы как можно раньше.

Вспользуемся алгоритмом из п.7. В данном случае $N_0 = \{7, 8\}$, $N_1 = \{1, 2, 3, 4\}$, $N_2 = \{5, 6\}$.

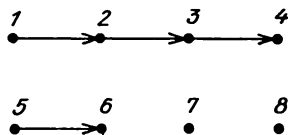


Рис. 4.4

Введем в рассмотрение два новых "объекта" $\bar{1}$ и $\bar{2}$, для которых положим $a_{\bar{1}} = a(N_1) = 14$, $b_{\bar{1}} = b(N_1) = 8$, $a_{\bar{2}} = a(N_2) = 4$, $b_{\bar{2}} = b(N_2) = 9$.

Так как выполняется соотношение $a_{\bar{1}} + b_{\bar{1}} = 22 = \max\{a(N), b(N)\}$, $\max\{a_i + b_i \mid i \in N_0\}$, $\max\{a_l + b_l \mid l = \bar{1}, \bar{2}\}$, то построим по алгоритму п. 1.3 – 1.6 § 1 гл. 2 расписание s_2 ремонта объектов 1, 2, 3 и 4 (рис. 4.5, а).

Получим $\bar{t}_{\max}(s_2) = 16 < \max\{a(N), b(N)\} = 21$.

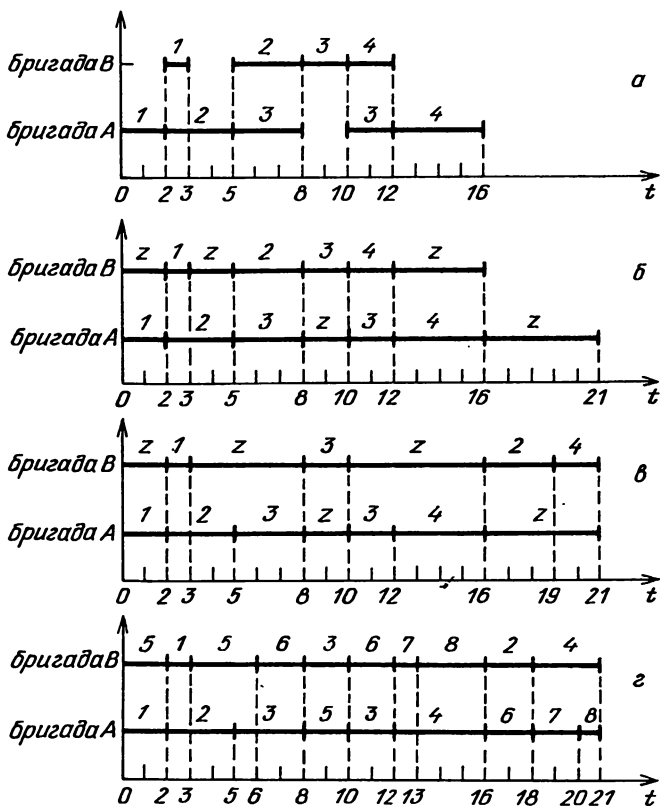


Рис. 4.5

Введем новый "объект" z , для которого положим $a_z = a(N \setminus N_1) = 7$, $b_z = b(N \setminus N_1) = 13$.

При расписании s_2 в интервале $(0, 16]$ имеем $w_A = 2$ и $w_B = 8$, т.е. $a_z > w_A$. "Объект" z должен ремонтироваться бригадой A в интервале $(0, 21]$ так, как изображено на рис. 4.5, б. На этом же рисунке изображено, как "объект" z мог бы ремонтироваться бригадой B в течение $\min\{b_z, w_B\} = 8$ единиц времени.

Бригада B должна ремонтировать "объект" z еще в течение пяти единиц времени.

Выберем интервалы $(5, 8]$ и $(10, 12]$; их суммарная длина равна 5, и при расписании s_2 (см. рис. 4.5, а) в этих интервалах обе бригады ремонтируют объекты множества N_1 . Объекты, ранее ремонтируемые бригадой B в выбранных интервалах, будут теперь ремонтироваться этой бригадой в интервале $(16, 21]$, а в освободившихся интервалах бригада B будет ремонтировать "объект" z . Расписание ремонта объектов 1, 2, 3 и 3, а также "объекта" z изображено на рис. 4.5, в.

Чтобы получить искомое расписание, достаточно в построенном расписании заменить "объект" z на объекты 5, 6, 7 и 8 (в указанной последовательности). Полученное в результате расписание s^* изображено на рис. 4.5, г.

§ 5. Директивные сроки

Рассмотрим систему с M последовательными приборами и нефиксированным порядком их прохождения, на вход которой в момент времени $d_i \geq 0$ поступает требование $i \in N = \{1, 2, \dots, n\}$. Для каждого требования i задан директивный срок D_i , к которому необходимо или, во всяком случае, желательно завершить обслуживание этого требования. В этом параграфе описываются эффективные методы построения расписаний, при которых все требования обслуживаются в заданные директивные сроки (если такие расписания существуют), а также расписаний с минимаксным временным смещением. Предполагается, что в процессе обслуживания требований допускаются прерывания. Если прерывания запрещены, то задача минимизации максимального временного смещения оказывается NP -трудной в сильном смысле при $M \geq 2$, $d_i = 0$, $i \in N$. Показано также, что задача построения расписания, минимизирующего число запаздывающих требований, NP -трудна при $M \geq 2$, $d_i = 0$, $D_i = D$, $i \in N$, независимо от того, допускаются прерывания или нет.

5.1. Пусть требование $i \in N$ поступает в очередь на обслуживание в момент времени $d_i \geq 0$, длительность его обслуживания прибором $L \in \mathcal{M} = \{1, 2, \dots, M\}$ равна $t_{iL} \geq 0$, директивный срок — $D_i \geq 0$. Порядок прохождения приборов каждым требованием не фиксирован, допускаются прерывания в процессе обслуживания.

Опишем один из возможных подходов, который позволяет построить расписание s^0 , при котором все требования обслуживаются в заданные директивные сроки (т.е. $\bar{t}_i(s^0) \leq D_i$), либо убедиться в невозможности построения такого расписания.

Пусть δ_k , $1 \leq k \leq p+1 \leq 2n$, — упорядоченная по возрастанию последовательность попарно различных значений d_i и D_i , $i \in N$. Временные интер-

валы $(\delta_1, \delta_2]$, $(\delta_2, \delta_3]$, ..., $(\delta_p, \delta_{p+1}]$ пронумеруем числами от 1 до p и положим $\Delta_k = \delta_{k+1} - \delta_k$, $1 \leq k \leq p$.

Рассмотрим систему линейных соотношений

$$\sum_{L \in M} x_{iL}^k \leq \Delta_k, \quad k = 1, \dots, p, i \in N; \quad (5.1)$$

$$\sum_{i \in N} x_{iL}^k \leq \Delta_k, \quad k = 1, \dots, p, L \in \mathcal{M}; \quad (5.2)$$

$$\sum_{k=1}^p x_{iL}^k = t_{iL}, \quad i \in N, L \in \mathcal{M}. \quad (5.3)$$

Здесь x_{iL}^k означает суммарную длительность обслуживания требования $i \in N$ прибором L во временном интервале $(\delta_k, \delta_{k+1}]$. Величины x_{iL}^k неотрицательные, если $d_i \leq \delta_k$ и $D_i \geq \delta_{k+1}$, и равны нулю в противном случае.

При указанной интерпретации величин x_{iL}^k соотношения (5.1) и (5.2) соответственно означают, что суммарная длительность обслуживания требования $i \in N$ всеми приборами в интервале $(\delta_k, \delta_{k+1}]$ и суммарная длительность функционирования прибора $L \in \mathcal{M}$ в этом интервале не превосходят длины интервала Δ_k . Соотношение (5.3) означает, что все требования будут полностью обслужены.

Очевидно, если система (5.1) – (5.3) несовместна, то расписания s^0 не существует. Если же система совместна и $\tilde{x}_{iL}^k$, $i \in N$, $L \in \mathcal{M}$, $k = 1, \dots, p$, – ее решение, то расписание s^0 существует и может быть построено, например, следующим образом.

Для каждого k , $1 \leq k \leq p$, используя один из методов, описанных в п. 3.2 или п. 3.4 § 3 этой главы, построим оптимальное по быстродействию расписание обслуживания (с прерываниями) множества N требований множеством $\mathcal{M}$ приборов во временном интервале $(\delta_k, \delta_{k+1}]$, считая величины $\tilde{x}_{iL}^k$ длительностями обслуживания требования $i \in N$ прибором $L \in \mathcal{M}$ в указанном интервале.

Поскольку $p < 2n$ и алгоритмы из § 3 этой главы имеют полиномиальную (относительно n и M) трудоемкость, а решение системы (5.1) – (5.3) также может быть найдено за полиномиальное число операций, то в результате выполнения *полиномиального числа операций* можно построить расписание s^0 , при котором все требования обслуживаются в заданные дедлайновые сроки, либо убедиться в невозможности его построения.

5.2. Рассмотрим некоторые частные случаи.

1) Пусть $d_i = 0$, $D_i = D$, $i \in N$. В этом случае можно воспользоваться одним из описанных в § 3 этой главы методов построения оптимального по быстродействию расписания s^* обслуживания (с прерываниями) требований множества N . *Расписание s^* является искомым, если $\bar{t}_{\max}(s^*) \leq D$.*

В противном случае не существует расписания, при котором все требования обслуживаются в заданные директивные сроки.

2) Пусть $d_i = 0, i \in N' \subset N$; $d_i = d, i \in N'' = N \setminus N'$; $D_i = D, i \in N$; $0 < d < D$. Построим сеть $\tilde{G}$, множество вершин которой состоит из множества $V = \{v_i \mid i \in N'\}$ вершин, соответствующих требованиям множества N' , множества $W = \{w_L \mid L \in \mathcal{M}\}$ вершин, соответствующих приборам, а также содержит источник и сток. Источник соединен дугой с каждой вершиной $v_i \in V$, величина допустимого потока по этой дуге ограничена снизу величиной $\max\{t_i(\mathcal{M}) - d, 0\}$ и сверху — величиной $\min\{t_i(\mathcal{M}), D - d\}$. Каждая вершина $w_L \in W$ соединена дугой со стоком, величина допустимого потока по этой дуге ограничена снизу величиной $\max\{t_L(N') - d, 0\}$ и сверху — величиной $\min\{t_L(N'), D - d - t_L(N'')\}$. Каждая вершина $v_i, i \in N'$, соединена с каждой вершиной $w_L, L \in M$, дугой пропускной способности t_{iL} .

Покажем, что расписание s^0 существует тогда и только тогда, когда в сети $\tilde{G}$ существует допустимый поток.

Пусть допустимый поток существует и f_{iL} — значение потока по дуге $(v_i, w_L), i \in N', L \in M$.

Для требования $i \in N''$ суммарные длительности его обслуживания прибором L во временных интервалах $(d, D]$ и $(0, d]$ соответственно равны t_{iL} и 0. Величины f_{iL} и $t_{iL} - f_{iL}$ будем интерпретировать как суммарные длительности обслуживания требования $i \in N'$ прибором L в интервалах $(d, D]$ и $(0, d]$ соответственно. Тогда величина потока по дуге из источника в любую вершину $v_i, i \in N'$, означает суммарную длительность обслуживания требования i всеми приборами во временном интервале $(d, D]$. Из допустимости потока следует, что каждое требование $i \in N'$ обслуживается всеми приборами в интервале $(d, D]$ не более $D - d$ единиц времени и в интервале $(0, d]$ — не более d единиц времени. С другой стороны, величину потока по дуге из вершины $w_L, L \in \mathcal{M}$, в сток можно интерпретировать как суммарное время обслуживания требований множества N' прибором L во временном интервале $(d, D]$. Из допустимости потока следует, что каждый прибор L во временном интервале $(0, d]$ обслуживает требования множества N' не более d единиц времени, а во временном интервале $(d, D]$ — требования множества N не более $D - d$ единиц времени.

Таким образом, если в сети $\tilde{G}$ существует допустимый поток, то могут быть найдены суммарные длительности обслуживания каждого требования каждым прибором в каждом из временных интервалов $(0, d]$ и $(d, D]$ и для построения расписания s^0 с $\bar{t}_{\max}(s^0) \leq D$ можно использовать подход, описанный в предыдущем пункте.

Нетрудно убедиться, что если в сети $\tilde{G}$ допустимого потока нет, то расписание s^0 не может быть построено.

Отметим, что для построения допустимого потока в сети $\tilde{G}$ требуется выполнить не более $O(n^3 + M^3)$ операций.

3) Пусть $d_i = 0, i \in N$; $D_i = D', i \in N' \subset N$; $D_i = D'', i \in N'' = N \setminus N', 0 < D' < D''$. Полагая $D = D'$ и $d = D' - D''$, приходим к ситуации, рассмотренной в п. 2).

5.3. Перейдем к рассмотрению задачи построения расписания s^* , которому соответствует наименьшее значение максимального временного смещения $L_{\max}(s) = \max\{\bar{t}_i(s) - D_i \mid i \in N\}$.

Нетрудно убедиться, что величина $L_{\max}(s^*)$ равна наименьшему значению λ^* параметра λ такого, что каждое требование $i \in N$ может быть обслужено не позднее момента времени $D_i + \lambda$. Обозначим множество таких расписаний через $S(\lambda)$. Для нахождения λ^* воспользуемся методом дихотомии.

Положим $\underline{\lambda} = -\max\{D_i - t_i(\mathcal{M}) - d_i \mid i \in N\}$ и $\bar{\lambda} = \max\{D_i \mid i \in N\} + \max\{\max_{i \in N} t_i(\mathcal{M}), \max_{L \in \mathcal{M}} t_L(N)\}$; $\Delta = \bar{\lambda} - \underline{\lambda}$. Очевидно, $S(\underline{\lambda}) = \emptyset$ и $S(\bar{\lambda}) \neq \emptyset$.

Вычислим $\tilde{\lambda} = (\bar{\lambda} + \underline{\lambda})/2$. Используя подход, описанный в п. 5.1, проверяем выполнимость условия $S(\tilde{\lambda}) \neq \emptyset$. Если $S(\tilde{\lambda}) = \emptyset$, то полагаем $\underline{\lambda} = \tilde{\lambda}$, в противном случае полагаем $\bar{\lambda} = \tilde{\lambda}$, и переходим к следующей итерации. Процесс продолжается до тех пор, пока не выполнится равенство $\underline{\lambda} = \bar{\lambda}$. Это значение λ и является искомым λ^* .

На каждой из $O(\log \Delta)$ итераций алгоритма нахождения λ^* требуется проверять выполнимость условия $S(\tilde{\lambda}) \neq \emptyset$. Так как алгоритм из п. 5.1 имеет полиномиальную трудоемкость, то нахождение значения λ^* и построение соответствующего расписания потребует выполнения не более чем полиномиального числа операций.

5.4. В некоторых частных случаях объем вычислений, необходимых для нахождения значения λ^* и построения соответствующего расписания s^* , может быть существенно уменьшен. Рассмотрим, например, задачу минимизации максимального временного смещения $L_{\max}(s)$ при $M = 2$ и $d_i = 0, i \in N$.

Пусть в момент времени $d = 0$ в систему, состоящую из двух приборов A и B , поступает множество $N = \{1, 2, \dots, n\}$ требований, a_i и b_i — длительности обслуживания требования $i \in N$ приборами A и B соответственно. Заданы директивные сроки $D_i \geq 0$, требования пронумерованы таким образом, что $D_1 \leq D_2 \leq \dots \leq D_n$.

1) Установим необходимые и достаточные условия существования расписания, при котором все требования обслуживаются в заданные директивные сроки.

Пусть $N_j = \{1, 2, \dots, j\}, 1 \leq j \leq n; N_0 = \emptyset$,

$$a(N_j) = \sum_{i \in N_j} a_i, \quad b(N_j) = \sum_{i \in N_j} b_i, \quad a(N_0) = b(N_0) = 0.$$

Выберем некоторое требование $j \in N$ и предположим, что существует расписание обслуживания требований множества N_{j-1} , при котором для всех требований этого множества не нарушаются директивные сроки.

Рассмотрим временной интервал $(0, D_j]$. Суммарную длину всех его подынтервалов, в течение которых простаивает только прибор A , только прибор B или оба прибора A и B , обозначим соответственно через x_j , y_j и z_j .

Поскольку обслуживание каждого требования множества N_{j-1} завершается не позднее соответствующего директивного срока, то имеют место соотношения

$$\begin{aligned} x_j + z_j &= D_j - a(N_{j-1}), \\ y_j + z_j &= D_j - b(N_{j-1}). \end{aligned} \quad (5.4)$$

Для того чтобы обслуживание требования j завершилось не позднее директивного срока D_j , необходимо и достаточно, чтобы выполнялась система неравенств

$$\begin{aligned} x_j + z_j &\geq a_j, \\ y_j + z_j &\geq b_j, \\ x_j + y_j + z_j &\geq a_j + b_j. \end{aligned} \quad (5.5)$$

Подставляя (5.4) в (5.5), получаем равносильную систему неравенств

$$\begin{aligned} a(N_j) &\leq D_j, \\ b(N_j) &\leq D_j, \\ a(N_j) + b(N_j) &\leq 2D_j - z_j. \end{aligned} \quad (5.6)$$

Вычислим наименьшие возможные (в смысле существования расписания без запаздывающих требований) значения z_j , $j \in N$. При $j = 1$, очевидно, $z_1 = D_1$. При $j = q$, $q > 1$, значение z_q будет не меньше чем $D_q - D_{q-1}$. Кроме того, длительность одновременного простоя обоих приборов в интервале $(0, D_{q-1}]$ в худшем случае будет равна $\max\{z_{q-1} - a_{q-1} - b_{q-1}, 0\}$. Следовательно, получаем рекуррентное соотношение

$$z_1 = D_1, \quad z_j = D_j - D_{j-1} + \max\{0, z_{j-1} - a_{j-1} - b_{j-1}\}. \quad (5.7)$$

Таким образом, для существования расписания, при котором все требования обслуживаются в заданные директивные сроки, необходимо и достаточно, чтобы система неравенств (5.6) была совместна для каждого j , $1 \leq j \leq n$, и значений z_j , вычисленных по формулам (5.7). Проверку выполнимости этого условия можно осуществить за $O(n)$ операций.

2) Как уже отмечалось в п. 5.3, минимальное значение $L_{\max}(s)$ равно минимальному значению λ^* параметра λ такого, что каждое требование $i \in N$ может быть обслужено не позже директивного срока $D_i + \lambda$.

В рассматриваемом случае может быть предложен *специальный метод нахождения величины λ^** (отличный от метода дихотомии).

Нетрудно получить соответствующее рекуррентное соотношение для z_j , подставив в (5.7) $D_j + \lambda$ вместо D_j . Для дальнейших рассуждений удобно, однако, чтобы параметр λ не фигурировал в рекуррентном соотношении.

Покажем, что

$$z_j = \max \{ D_j + \lambda - a(N_{j-1}) - b(N_{j-1}), z'_j \}, \quad 1 \leq j \leq n, \quad (5.8)$$

где

$$z'_1 = -\infty, \quad z'_j = D_j - D_{j-1} + \max \{ 0, z'_{j-1} - a_{j-1} - b_{j-1} \}, \quad j = 2, \dots, n. \quad (5.9)$$

Воспользуемся методом математической индукции. При $j = 1$ соотношение (5.8), очевидно, выполняется. Пусть (5.8) справедливо при всех j , меньших некоторого $q \in N$. Из (5.7) следует, что

$$z_q = D_q - D_{q-1} + \max \{ 0, z_{q-1} - a_{q-1} - b_{q-1} \}.$$

Заменяя в этом соотношении z_{q-1} согласно (5.8), получаем

$$\begin{aligned} z_q &= D_q - D_{q-1} + \max \{ 0, \max \{ (D_{q-1} + \lambda) - \\ &- a(N_{q-2}) - b(N_{q-2}), z'_{q-1} \} - a_{q-1} - b_{q-1} \} = \\ &= D_q - D_{q-1} + \max \{ 0, D_{q-1} + \lambda - a(N_{q-1}) - b(N_{q-1}), \\ &z'_{q-1} - a_{q-1} - b_{q-1} \}. \end{aligned}$$

Из (5.9) при $j = q$ следует, что $z_q = D_q - D_{q-1} + \max \{ D_{q-1} + \lambda - a(N_{q-1}) - b(N_{q-1}), z'_q - (D_q - D_{q-1}) \}$, откуда непосредственно следует соотношение (5.8) при $j = q$.

Подставляя в (5.6) $D_j + \lambda$ вместо D_j и учитывая (5.8), получаем систему неравенств

$$\begin{aligned} a(N_j) &\leq D_j + \lambda, \\ b(N_j) &\leq D_j + \lambda, \\ a_j + b_j &\leq D_j + \lambda, \\ a(N_j) + b(N_j) &\leq 2(D_j + \lambda) - z'_j. \end{aligned} \quad (5.10)$$

Наименьшее значение параметра λ , при котором система (5.10) совместна для каждого $j \in N$, равно

$$\lambda^* = \max_{j \in N} \left\{ \max \left\{ a(N_j), b(N_j), a_j + b_j, \frac{1}{2} (a(N_j) + b(N_j) + z'_j) \right\} - D_j \right\}. \quad (5.11)$$

Значение λ^* может быть найдено за $O(n)$ операций путем вычисления значений $z'_j, j \in N$, и последующего использования формулы (5.11).

3) Опишем способ построения расписания, минимизирующего величину $L_{\max}(s)$, или, что то же, расписания, при котором директивные сроки $D_i^* = D_i + \lambda^*$ не нарушаются ни для одного требования.

Рассмотрим интервал $(0, D_1^*]$. Поскольку $a_1 + b_1 \leq D_1^*$, положим $t_{1B}^0 = 0$, $\bar{t}_{1B} = b_1 = t_{1A}^0$. Тогда в интервале $(0, D_2^*]$ перед обслуживанием требования $j = 2$ оба прибора одновременно простаивают в подынтервале $(D_2^* - z_2, D_2^*]$ и только в этом подынтервале. Кроме того, имеется подынтервал $(D_2^* - z_2 - y_2, D_2^* - z_2]$, в котором простаивает только прибор B .

Пусть построено расписание обслуживания требований множества N_{j-1} , $j > 1$, при котором ни одно из требований $i \in N_{j-1}$ не запаздывает относительно директивного срока D_i^* . Это расписание построено таким образом, что в интервале $(0, D_j^*]$ оба прибора простаивают одновременно только в подынтервале $(D_j^* - z_j, D_j^*]$, $D_j - z_j \leq D_{j-1}$, и один из подынтервалов, в которых простаивает только прибор B , имеет вид $(t, D_j^* - z_j]$.

Требование j будем обслуживать таким образом, чтобы $\bar{t}_j \leq D_j^*$; в интервале $(0, D_{j+1}^*]$ после обслуживания требования j перед обслуживанием требования $j + 1$ оба прибора одновременно простаивали только в подынтервале $(D_{j+1}^* - z_{j+1}, D_{j+1}^*]$; нашелся бы интервал вида $(t, D_{j+1}^* - z_{j+1}]$, в течение которого прибор B простаивал.

Возможны следующие ситуации.

а) Пусть $a_j + b_j \leq z_j$. Тогда требование j может быть полностью обслужено в интервале $(D_j^* - z_j, D_j^*]$. Положим $t_{jB}^0 = D_j^* - z_j$, $\bar{t}_{jB} = t_{jB}^0 + b_j = t_{jA}^0$, $\bar{t}_{jA} = t_{jA}^0 + a_j$.

б) Пусть $a_j + b_j > z_j$. Максимально возможная длительность обслуживания требования j прибором A в интервале $(D_j^* - z_j, D_j^*]$ равна $a_j' = \min \{a_j, z_j, z_j - (b_j - y_j)\}$, где y_j — суммарная длительность простоя прибора B в интервале $(0, D_j^*]$ после завершения обслуживания требований множества N_{j-1} .

I) Пусть $a_j' = a_j$. Тогда требование j может быть обслужено без прерываний прибором A , причем $\bar{t}_{jA} = D_j^*$. Требование j может обслуживаться прибором B в интервале $(D_j^* - z_j, D_j^* - a_j]$, а также в течение $a_j + b_j - z_j$ единиц времени в интервалах, где простаивает только прибор B . Действительно, так как $a_j' = a_j$, то $a_j \leq z_j - (b_j - y_j)$, т.е. $a_j + b_j - z_j \leq y_j$.

II) Пусть $a_j' = z_j$. Тогда требование j может обслуживаться прибором A в интервале $(D_j^* - z_j, D_j^*]$, а также в течение $a_j - z_j$ единиц времени в интервалах, где простаивает только прибор A . Прибор B может обслуживать требование j в интервалах, где он простаивает. Так как $a_j' = z_j$, то $b_j < y_j$ и такое обслуживание может быть осуществлено.

III) Пусть $a_j' = z_j - (b_j - y_j)$. Это означает, что $b_j \geq y_j$. Поэтому требование j может обслуживаться во всех интервалах, где простаивает прибор B , а также в течение $b_j - y_j$ единиц времени в интервале $(D_j^* - z_j, D_j^* - a_j']$. Требование j может обслуживаться прибором A в течение a_j' единиц времени в интервале $(D_j^* - a_j', D_j^*]$, а также (поскольку $a_j \geq a_j'$), в течение $a_j - a_j'$ единиц времени в интервалах, где простаивает прибор A .

Нетрудно проверить, что построенное таким образом расписание обслуживания требований множества N_j удовлетворяет всем указанным выше условиям.

Таблица 5.1

j	1	2	3	4	5
a_j	2	4	1	3	4
b_j	3	3	2	3	6
D_j	6	9	13	15	16

Таблица 5.2

j	1	2	3	4	5
$a(N_j)$	2	6	7	10	14
$b(N_j)$	3	6	8	11	17
z'_j	$-\infty$	3	4	3	1

Можно показать, что *трудоемкость описанного процесса последовательного построения расписания s^* не превосходит $O(n)$ операций*, если для хранения подынтервалов простоя каждого из приборов использовать специальную структуру данных (так называемый стек).

5.5. П р и м е р. Построим расписание s^* обслуживания $n = 5$ требований двумя приборами A и B , которому соответствует наименьшее значение $L_{\max}(s)$. Длительности обслуживания и директивные сроки приведены в табл. 5.1. Каждое требование может обслуживаться, начиная с момента $d = 0$. Допускаются прерывания.

Вычислим частичные суммы $a(N_j)$, $b(N_j)$ и найдем значения z'_j по формуле (5.9). Результаты вычислений приведены в табл. 5.2.

Значение λ^* , вычисленное по формуле (5.11), равно единице. Положим $D_j^* = D_j + 1$.

Построим расписание, при котором не нарушаются директивные сроки D_j^* . Для $j = 1$ расписание изображено на рис. 5.1, а.

При $j = 2$ имеем $z_2 = 10 - 5 = 5$, $y_2 = 2$. Выполняется неравенство $a_2 + b_2 = 4 + 3 > z_2 = 5$. Вычислим $a'_2 = \min \{4, 5, 5 - (3 - 2)\} = 4$. Требование $j = 2$ обслуживается так, как представлено на рис. 5.1 б.

При $j = 3$ имеем $z_3 = 4$ и $a_3 + b_3 = 1 + 2 < z_3 = 4$. Расписание обслуживания требований $\{1, 2, 3\}$ изображено на рис. 5.1 в.

При $j = 4$ имеем $z_4 = 3$, $y_4 = 5$. Так как $a_4 + b_4 = 3 + 3 > z_4 = 3$, вычислим $a'_4 = \min \{3, 3, 3 - (3 - 5)\} = 3$. Расписание обслуживания требований $\{1, 2, 3, 4\}$ представлено на рис. 5.1, г.

При $j = 5$ имеем $z_5 = 1$, $y_5 = 5$; так как $a_5 + b_5 = 4 + 6 > z_5 = 5$, вычислим $a'_5 = \min \{4, 1, 1 - (6 - 5)\} = 0$. Искомое расписание изображено на рис. 5.1, д.

5.6. Задача построения расписания s^* , минимизирующего максимальное временное смещение $L_{\max}(s^*)$, существенно усложняется, если прерывания процесса обслуживания каждого требования каждым прибором запрещены. Покажем, что *эта задача является NP-трудной в сильном смысле при $M \geq 2$ и $d_i = 0, i \in N$* .

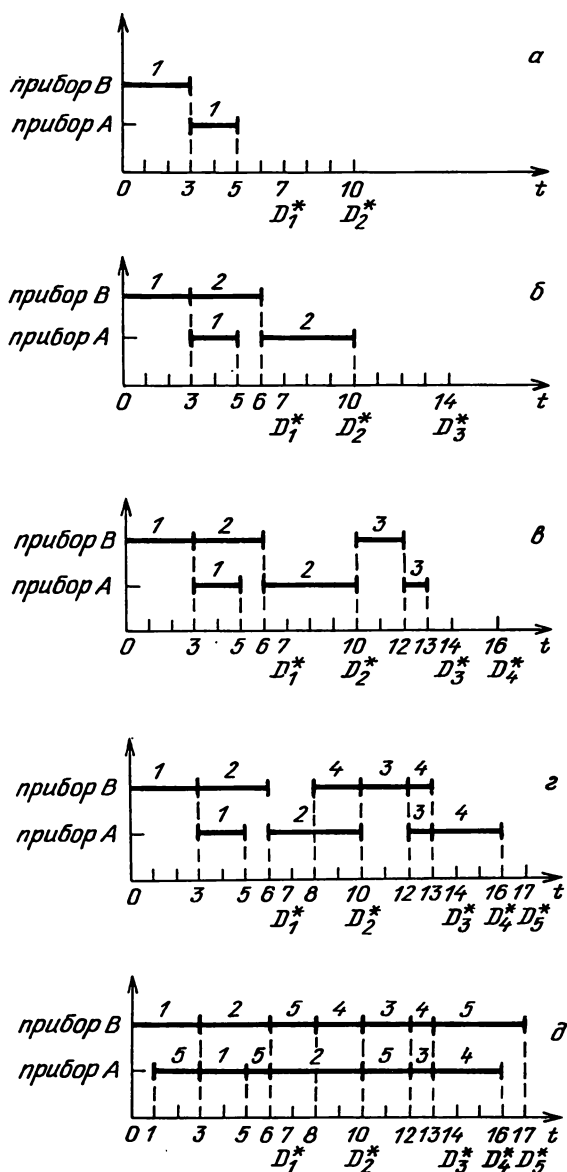


Рис. 5.1

Рассмотрим соответствующую задачу распознавания при $M = 2$: определить, существует ли расписание s^0 обслуживания (без прерываний) множества N требований двумя приборами А и В такое, что $L_{\max}(s^0) \leq u$ для любого заданного числа u .

Покажем, что к этой задаче сводится задача о 3-разбиении (см. вводную часть этой главы).

Пусть a_i и b_i — длительности обслуживания требования приборами A и B соответственно, D_i — директивные сроки. Положим

$$n = 4n_0, \quad N = \{1, 2, \dots, n\};$$

$$a_i = 0, \quad b_i = e_i, \quad D_i = 3n_0 E, \quad i \in N^0;$$

$$a_{3n_0+1} = 0, \quad b_{3n_0+1} = 2E, \quad D_{3n_0+1} = 2E;$$

$$a_{3n_0+j} = 3E, \quad b_{3n_0+j} = 2E, \quad D_{3n_0+j} = (3j-1)E, \quad j = 2, \dots, n_0.$$

Покажем, что в построенной задаче расписание s^0 такое, что $L_{\max}(s^0) \leq y = 0$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

Пусть задача о 3-разбиении имеет решение и $N_j^0, j = 1, \dots, n_0$, — найденные подмножества. Тогда искомое расписание s^0 существует и может быть построено следующим образом. Каждый из приборов A и B обслуживает требования непрерывно одно за другим начиная с момента времени $d = 0$. Прибор A обслуживает требования $3n_0 + 2, 3n_0 + 3, \dots, 4n_0$ в указанной последовательности, а прибор B обслуживает все требования множества N в последовательности $3n_0 + 1, \pi_{N_1^0}, 3n_0 + 2, \pi_{N_2^0}, \dots, 4n_0 - 1, \pi_{N_{n_0}^0} - 1, 4n_0, \pi_{N_{n_0}^0}$.

Пусть существует расписание s^0 , при котором все требования обслуживаются в заданные директивные сроки $D_i, i \in N$. Поскольку $b_{3n_0+1} = D_{3n_0+1}, a_{3n_0+2} + b_{3n_0+2} = D_{3n_0+2}$, то при расписании s^0 значения $t_{3n_0+1,B}^0 = 0, t_{3n_0+2,A}^0 = 0, t_{3n_0+2,B}^0 = \bar{t}_{3n_0+2,A}$. Требование $3n_0 + 3$ не может обслуживаться сначала прибором B , затем прибором A , так как в этом случае $t_{3n_0+3,B}^0 \geq D_{3n_0+2}$ и $\bar{t}_{3n_0+3} > D_{3n_0+3}$. Поэтому для требования $3n_0 + 3$ справедливы соотношения $t_{3n_0+3,A}^0 = \bar{t}_{3n_0+2,A}, t_{3n_0+3,B}^0 = \bar{t}_{3n_0+3,A}$. Рассуждая аналогичным образом, приходим к заключению, что прибор A обслуживает требования $3n_0 + 2, 3n_0 + 3, \dots, 4n_0$ непрерывно одно за другим, а требование $3n_0 + j$ обслуживается прибором B во временном интервале $(3(j-1)E, (3j-1)E], 1 \leq j \leq n_0$. Тогда требования множества N^0 должны обслуживаться прибором B во временных интервалах $((3j-1)E, 3jE], 1 \leq j \leq n_0$. Так как $E/4 < e_i < E/2$ и длина каждого из указанных интервалов равна E , то в каждом из них обслуживается ровно три требования множества N^0 . Обозначая через N_j^0 множество требований, обслуживаемых прибором B во временном интервале $((3j-1)E, 3jE], j = 1, \dots, n_0$, получаем решение задачи о 3-разбиении.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

5.7. Рассмотрим задачу построения расписания, минимизирующего число требований, запаздывающих относительно директивных сроков. Покажем, что *задача построения такого расписания (как с прерываниями, так и без прерываний) NP-трудна, даже если имеется только два обслуживающих прибора, все требования поступают в систему одновременно и имеют общий директивный срок.*

Доказательство проведем в несколько этапов.

1) Введем в рассмотрение *задачу распознавания*, которую в дальнейшем будем называть задачей о разбиении с ограничениями. Эта

задача заключается в следующем. Имеется множество $N^0 = \{1, 2, \dots, 2n_0\}$, каждому элементу i которого сопоставлено (рациональное) число e_i , причем $e(N^0) = 2E$, $0 < e_i < 2E/n_0$, $i = 1, \dots, 2n_0$. Требуется определить, можно ли разбить множество N^0 на два подмножества N_1^0 и N_2^0 , каждое из которых содержит ровно n_0 элементов и $e(N_1^0) = e(N_2^0) = E$.

Докажем, что эта задача *NP-трудна*. Для этого покажем, что к ней сводится следующая задача распознавания. Имеется множество $\bar{N}^0 = \{1, 2, \dots, \bar{n}_0\}$, каждому элементу которого сопоставлено натуральное число $\bar{e}_i$, причем $\bar{e}(\bar{N}^0) = 2\bar{E}$. Требуется определить, можно ли разбить множество $\bar{N}^0$ на два таких подмножества $\bar{N}_1^0$ и $\bar{N}_2^0$, каждое из которых содержит ровно $\bar{n}_0/2$ элементов, что $\bar{e}(\bar{N}_1^0) = \bar{e}(\bar{N}_2^0)$. Сформулированная задача *NP-полна* и представляет собой известную задачу о разбиении (см. вводную часть этой главы) в несколько суженной постановке.

Для доказательства сводимости положим:

$$\begin{aligned} n_0 &= \bar{n}_0, & e_i &= \bar{e}_i & i &= 1, \dots, n_0; \\ e_i &= \frac{2\bar{n}_0 - 1}{\bar{n}_0} \bar{E}, & i &= n_0 + 1, \dots, 2n_0; \\ E &= \frac{2\bar{n}_0 + 1}{2} \bar{E}. \end{aligned}$$

Нетрудно проверить, что $0 < e_i < 2E/n_0$, $i = 1, \dots, 2n_0$.

Покажем, что построенная задача имеет решение тогда и только тогда, когда множество $\bar{N}^0$ можно разбить на два подмножества $\bar{N}_1^0$ и $\bar{N}_2^0$ с указанными свойствами.

Пусть существует требуемое разбиение множества $\bar{N}^0$ на подмножества $\bar{N}_1^0$ и $\bar{N}_2^0$. Тогда имеет решение задача о разбиении с ограничениями, причем $N_1^0 = \bar{N}_1^0 \cup \{\bar{n}_0 + 1, \bar{n}_0 + 2, \dots, 3\bar{n}_0/2\}$, $N_2^0 = \bar{N}_2^0 \cup \{3\bar{n}_0/2 + 1, 3\bar{n}_0/2 + 2, \dots, 2\bar{n}_0\}$.

Пусть теперь подмножества N_1^0 и N_2^0 — решение задачи о разбиении с ограничениями. Тогда каждое из них содержит ровно $\bar{n}_0/2$ элементов, больших $\bar{n}_0$.

Действительно, пусть, например, подмножество N_1^0 содержит $\bar{n}_0/2 + 1$ элементов, больших $\bar{n}_0$. Тогда

$$\begin{aligned} e(N_1^0) &> \frac{2\bar{n}_0 - 1}{\bar{n}_0} \bar{E} \left(\frac{\bar{n}_0}{2} + 1 \right) = \frac{2\bar{n}_0^2 + 3\bar{n}_0 - 2}{2\bar{n}_0} \bar{E} \\ e(N_2^0) &< \frac{2\bar{n}_0 - 1}{\bar{n}_0} \bar{E} \left(\frac{\bar{n}_0}{2} - 1 \right) + 2\bar{E} = \frac{2\bar{n}_0^2 - 5\bar{n}_0 + 2}{2\bar{n}_0} \bar{E} + 2\bar{E}. \end{aligned}$$

Следовательно, $e(N_1^0) - e(N_2^0) > (2 - 2/\bar{n}_0) \bar{E} > 0$.

Таким образом, в каждом из подмножеств N_1^0 и N_2^0 содержится ровно $\bar{n}_0/2$ элементов, больших $\bar{n}_0$, и ровно $\bar{n}_0/2$ элементов, не меньших $\bar{n}_0$. Полагая

$$\bar{N}_1^0 = \{i \in N_1^0 \mid 1 \leq i \leq \bar{n}_0\}, \quad \bar{N}_2^0 = \{i \in N_2^0 \mid 1 \leq i \leq \bar{n}_0\},$$

получаем требуемое разбиение множества N^0 .

Тем самым за $O(\bar{n}_0)$ операций реализовано сведение задачи о разбиении (в суженной постановке) к задаче о разбиении с ограничениями.

2) Рассмотрим следующую задачу распознавания, соответствующую задаче построения расписания, минимизирующего число запаздывающих требований.

Имеется два прибора A и B и множество $N = \{1, 2, \dots, n\}$ требований. Длительности обслуживания требования $i \in N$ приборами A и B соответственно равны a_i и b_i . Задан директивный срок D , общий для всех требований. Необходимо определить, существует ли такое расписание s^0 , при котором число запаздывающих требований не больше заданного числа y .

Покажем, что эта задача является NP-трудной независимо от того, разрешены прерывания или нет. Для этого покажем, что к ней сводится задача о разбиении с ограничениями.

Положим $n = 2n_0$, $a_i = e_i$, $b_i = 2E/n_0 - e_i$, $i = 1, \dots, n$; $D = E$. Покажем, что в построенной задаче расписание s^0 , при котором число запаздывающих требований не больше $y = n_0$, существует тогда и только тогда, когда имеет решение задача о разбиении с ограничениями.

Пусть задача о разбиении с ограничениями имеет решение, и N_1^0, N_2^0 — найденные подмножества. Поскольку $e(N_1^0) = D$, то $a(N_1^0) = D$, и $b(N_1^0) = n_0 \cdot 2D/n_0 - D = D$. Следовательно, множество требований N_1^0 может быть полностью обслужено во временном интервале $(0, D]$, причем в этом интервале оба прибора не обслуживают требований множества N_2^0 .

Расписание обслуживания множества требований N_1^0 может быть построено в соответствии с алгоритмом из п. 1.2 § 1 гл. 3 (если прерывания запрещены) или в соответствии с одним из алгоритмов в § 3 той же главы (если прерывания разрешены). Множество требований N_2^0 может обслуживаться во временном интервале $(D, +\infty)$ произвольным образом.

В результате получаем расписание s^0 , при котором число запаздывающих требований равно $|N_2^0| = n_0 = y$.

Пусть существует расписание s^0 , при котором число запаздывающих требований не превосходит y . Пусть R — множество всех незапаздывающих требований. Тогда $a(R) \leq D$, $b(R) \leq D$. Заметим, что $|R| \geq n_0$. Если при этом $|R| > n_0$, то $b(R) = 2D|R|/n_0 - a(R) > (2|R| - n_0)D/n_0 > D$. Поэтому $|R| = n_0$. Далее, если $a(R) < D$, то $b(R) = 2D - a(R) > D$. Следовательно, $a(R) = b(R) = D$. Полагая $N_1^0 = R$ и $N_2^0 = N \setminus R$, получаем решение задачи о разбиении с ограничениями.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о разбиении с ограничениями к рассматриваемой задаче распознавания.

§ 6. Суммарное время обслуживания. Одинаковые длительности

В систему с M приборами в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Порядок прохождения приборов нефиксирован и может быть различным для разных требований. Предполагается, что длительности t_{iL} обслуживания требований приборами одинаковы. Не нарушая общности рассуждений, их можно считать равными единице. В этом параграфе описываются эффективные алгоритмы построения рас-

писаний без прерываний, минимизирующих (взвешенную) сумму моментов завершения обслуживания требований, при условии, что все $t_{iL} = 1$, $i = 1, \dots, n$, $L = 1, \dots, M$. Если некоторые $t_{iL} = 0$, то задача минимизации суммарного времени обслуживания оказывается *NP*-трудной.

6.1. Рассмотрим задачу построения расписания s^* без прерываний, минимизирующего суммарное время $\Sigma \bar{t}_i(s)$ обслуживания требований множества N системой, состоящей из M приборов. Предполагается, что каждое требование должно быть обслужено каждым прибором, причем все длительности обслуживания $t_{iL} = 1$.

Пусть дано произвольное расписание s и требования пронумерованы по неубыванию значений $\bar{t}_i(s)$, $i \in N$. Положим

$$N_p = \{1, 2, \dots, p\} \subseteq N, \quad p = 1, \dots, n, \quad \bar{t}(N_p, s) = \sum_{i \in N_p} \bar{t}_i(s).$$

Произвольному расписанию s для исходной задачи соответствует расписание $\bar{s}$ обслуживания требований множества $N = \{1, 2, \dots, n\}$ в системе, состоящей из M параллельных идентичных приборов, при условии, что длительность обслуживания каждого требования равна M и в процессе обслуживания каждого требования допускаются прерывания. Отметим, что $\bar{t}_i(s) = \bar{t}_i(\bar{s})$, $i \in N$.

Расписание, которому соответствует минимальное значение суммарного времени обслуживания в системе с параллельными приборами, можно, как известно, искать в классе расписаний, при которых процесс обслуживания каждого требования протекает без прерываний.

Пусть $n = kM + r$, $k \geq 0$, $0 \leq r < M$, k и r — целые числа. В качестве оптимального расписания для построенной задачи с параллельными приборами может быть выбрано, например, расписание, при котором r приборов функционируют без простоев во временном интервале $(0, (k+1)M]$ и обслуживают (без прерываний) по $k+1$ требований каждый, а остальные $M - r$ приборов функционируют без простоев в интервале $(0, kM]$ и обслуживают (без прерываний) по k требований каждый. Суммарное время обслуживания при таком расписании равно $(k+1)Mr + M^2 k(k+1)/2$.

Таким образом, при любом расписании s для исходной задачи имеет место соотношение

$$\bar{t}(N_{kM+r}, s) \geq (k+1)Mr + M^2 k(k+1)/2. \quad (6.1)$$

Укажем несколько способов построения расписания s^* , при котором в (6.1) имеет место знак равенства.

Поскольку в процессе обслуживания не допускается прерываний, то искоемое расписание s^* однозначно определяется матрицей $\|t_{iL}(s^*)\|$ моментов завершения обслуживания каждого требования каждым прибором. Введем в рассмотрение матрицу $\Lambda = \|\lambda_{Lq}\|$, где $\lambda_{Lq} = i$, если прибор L обслуживает требование $i \in N$ q -ым по порядку.

1) Пусть $n = kM$. Матрицу Λ построим следующим образом. Пусть Λ_j — матрица размерности $M \times M$, представляющая собой латинский квадрат, построенный на множестве $\{(j-1)M+1, (j-1)M+2, \dots, jM\}$, $1 \leq j \leq k$ (см. § 2 этой главы).

Будем интерпретировать матрицу Λ_j как подматрицу матрицы Λ , содержащую ее столбцы с номерами $(j-1)M+1, (j-1)M+2, \dots, jM$. Положим $\bar{t}_{\lambda_{Lq}, L}(s_1^*) = q, q = 1, \dots, n, L = 1, \dots, M$.

При расписании s_1^* , определяемом матрицей $\|\bar{t}_{iL}(s_1^*)\|$, ровно M требований будут обслужены к каждому моменту времени $jM, 1 \leq j \leq k$, поэтому $\bar{t}(N_n, s_1^*) = M^2 k(k+1)/2$, т.е. расписание s_1^* оптимально.

Отметим, что в рассматриваемом случае расписание s_1^* является также оптимальным по быстродействию. Кроме того, $\bar{t}(N_p, s_1^*) \leq \bar{t}(N_p, s)$ для всех $p, 1 \leq p \leq n$, и любого расписания s .

2) Пусть $n = kM + r, k \geq 0, 0 < r < M$. Дополним множество требований $M - r$ фиктивными требованиями $n+1, n+2, \dots, n+M-r$, длительность обслуживания каждого из которых любым прибором положим равной единице. Матрицу $\Lambda = \|\lambda_{Lq}\|$ размерности $M \times (k+1)M$ разобьем на $k+1$ подматриц Λ_j , содержащих столбцы матрицы с номерами $(j-1)M+1, (j-1)M+2, \dots, jM, 1 \leq j \leq k+1$.

а) Пусть каждая из подматриц Λ_j представляет собой латинский квадрат, построенный на множестве $\{(j-1)M+1, (j-1)M+2, \dots, jM\}, 1 \leq j \leq k+1$.

Положим $\bar{t}_{\lambda_{Lq}, L}(s_2^*) = q, q = 1, \dots, n+M-r, L = 1, \dots, M, \lambda_{Lq} \leq n$.

При расписании s_2^* , определяемом матрицей $\|\bar{t}_{iL}(s_2^*)\|$, ровно M требований обслуживаются к каждому моменту времени $jM, 1 \leq j \leq k$, и r требований — к моменту времени $(k+1)M$. Поэтому

$$\bar{t}(N_n, s_2^*) = \sum_{j=1}^k jM^2 + (k+1)rM = M^2(k+1)k/2 + (k+1)rM,$$

т.е. расписание s_2^* оптимально.

Нетрудно проверить, что $\bar{t}(N_p, s_2^*) \leq \bar{t}(N_p, s)$ для всех $p, 1 \leq p \leq n$, и любого расписания s .

Отметим, что расписания s_1^* и s_2^* остаются оптимальными и в случае, когда каждое требование должно обслуживаться без задержек.

б) Пусть $k \geq 1$ и каждая из подматриц Λ_j представляет собой латинский квадрат, построенный на множестве $\{(j-1)M+1, (j-1)M+2, \dots, jM\}, 1 \leq j \leq k-1$. Пусть подматрица Λ_k представляет собой латинский квадрат, построенный на множестве $\{(k-1)M+1, (k-1)M+2, \dots, (k-1)M+r, kM+r+1, kM+r+2, \dots, (k+1)M\}$, а подматрица Λ_{k+1} — латинский квадрат, построенный на множестве $\{(k-1)M+r+1, (k-1)M+r+2, \dots, kM+r\}$.

Если в некоторой строке $L, 1 \leq L \leq M$, матрицы Λ расположен элемент, равный $kM+r+j, 1 \leq j \leq M-r$, заменим его на элемент, расположенный в той же строке и в столбце с номером $kM+r+j$. После того как подматрица Λ_k не будет содержать элементы, большие чем $kM+r$, исключим из рассмотрения столбцы матрицы Λ , номера которых превышают $kM+r$. Полученная при этом матрица Λ имеет размерность $M \times (kM+r)$ и представляет собой латинский прямоугольник, построенный на множестве N .

Положим $\bar{t}_{\lambda_{Lq}, L}(s_3^*) = q, q = 1, \dots, n, L = 1, \dots, M$.

$L \backslash i$	1	2	3	4	5	6	7	8
1	1	2	3	4	5	6	7	8
2	2	3	1	5	6	4	8	9
3	3	1	2	6	4	5	9	7

При расписании s_3^* , определяемом матрицей $\|\bar{t}_{iL}(s_3^*)\|$, ровно M требований обслуживаются к каждому моменту времени jM , $1 \leq j \leq k-1$, r требований — к моменту времени kM и M требований — к моменту времени $kM+r$. Поэтому

$$\begin{aligned} \bar{t}(N_n, s_3^*) &= \sum_{j=1}^{k-1} jM^2 + kMr + M(kM+r) = \\ &= M^2 k(k+1)/2 + (k+1)Mr, \end{aligned}$$

т.е. расписание s_3^* оптимально.

Отметим, что это расписание будет также *оптимальным по быстрдействию*, так как $\max \{\bar{t}_i(s_3^*) | i \in N\} = kM + r = n$. Построение матриц Λ и $\|\bar{t}_{iL}(s^*)\|$ в каждом из рассмотренных выше случаев требует выполнения не более чем $O(nM)$ операций.

6.2. П р и м е р. Восемь одновременно пришедших в поликлинику пациентов должны пройти медицинский осмотр, побывав на приеме у трех врачей различной специализации. Время пребывания пациента у каждого из врачей равно 10 минутам. Требуется таким образом организовать прохождение медосмотра, чтобы суммарное время, затраченное всеми пациентами, было минимальным.

В качестве единицы измерения времени выберем 10 минут и придем к ситуации, рассмотренной в п. 6.1. Здесь $n = 8$, $M = 3$, $k = 2$, $r = 2$.

Построим оптимальное расписание вида s_2^* из п. 6.1.

Добавим фиктивного "пациента", присвоив ему номер 9. Остальных пациентов пронумеруем числами от 1 до 8.

Положим

$$\Lambda = \left\| \begin{array}{ccc|ccc|ccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 3 & 1 & 2 & 6 & 4 & 5 & 9 & 7 & 8 \\ 2 & 3 & 1 & 5 & 6 & 4 & 8 & 9 & 7 \end{array} \right\|.$$

Моменты завершения обслуживания пациентов приведены в табл. 6.1. Соответствующее расписание изображено на рис. 6.1 а.

Построим также расписание вида s_3^* из п. 6.1.

В качестве исходной выберем матрицу

$$\Lambda = \left\| \begin{array}{ccc|ccc|ccc} 1 & 2 & 3 & 4 & 5 & 9 & 6 & 7 & 8 \\ 3 & 2 & 1 & 9 & 4 & 5 & 8 & 6 & 7 \\ 2 & 3 & 1 & 5 & 9 & 4 & 7 & 8 & 6 \end{array} \right\|.$$

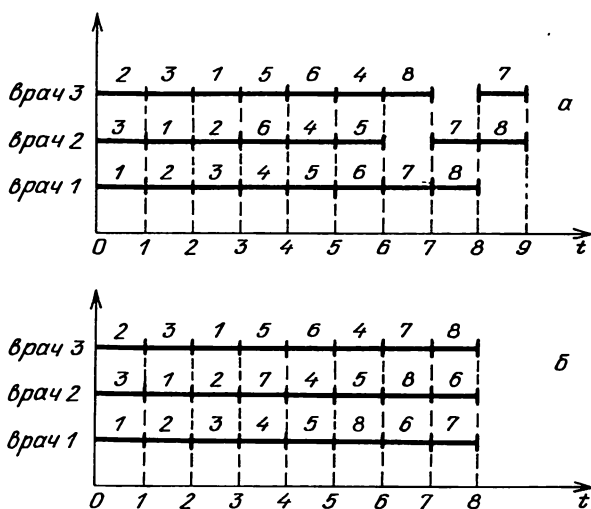


Рис. 6.1

Преобразованная матрица выглядит следующим образом:

$$\Lambda = \left\| \begin{array}{ccc|ccc|cc} 1 & 2 & 3 & 4 & 5 & 8 & 6 & 7 \\ 3 & 1 & 2 & 7 & 4 & 5 & 8 & 6 \\ 2 & 3 & 1 & 5 & 6 & 7 & 7 & 8 \end{array} \right\|.$$

Моменты завершения обслуживания пациентов приведены в табл. 6.2.

Соответствующее расписание представлено на рис. 6.1 б.

Суммарная длительность обслуживания при обоих построенных расписаниях равна 45 единицам времени (450 минут). Последнее расписание оптимально по быстрдействию.

6.3. Рассмотренная в предыдущих пунктах ситуация допускает естественное обобщение. Каждому требованию $i \in N$ сопоставим вес $w_i > 0$ и укажем способ построения расписания s^* (без прерываний), при котором величина $\sum_{i \in N} w_i \bar{t}_i(s)$ была бы минимальна. По-прежнему предполагается,

что все $t_{iL} = 1, i = 1, \dots, n, L = 1, \dots, M$.

Пронумеруем требования таким образом, что $w_1 \geq w_2 \geq \dots \geq w_n$.

Очевидно, для любой последовательности $x_1, x_2, \dots, x_n$ неотрицатель-

ных чисел линейная форма $\sum_{i=1}^n w_i x_{ji}$ достигает наименьшего значения

Таблица 6.2

$L \backslash i$	1	2	3	4	5	6	7	8
1	1	2	3	4	5	7	8	6
2	2	3	1	5	6	8	4	7
3	3	1	2	6	4	5	7	8

на множестве всех перестановок вида $(j_1, j_2, \dots, j_n)$ при $x_{j_1} \leq x_{j_2} \leq \dots \leq x_{j_n}$. Следовательно, расписание s^* следует искать в классе S расписаний s таких, что последовательность $\bar{t}_1(s), \bar{t}_2(s), \dots, \bar{t}_n(s)$ моментов завершения обслуживания требований является неубывающей.

Покажем, что в качестве расписания s^* может быть выбрано либо расписание s_1^* , если $n = kM$, либо s_2^* , если $n = kM + r$, $0 < r < M$, где расписания s_1^* и s_2^* построены так, как описано в п. 6.1. При этом, очевидно, $s^* \in S$. Из п. 6.1 следует, что для произвольного расписания $s \in S$ и указанного расписания справедливы соотношения

$$\sum_{i=1}^p \bar{t}_i(s^*) \leq \sum_{i=1}^p \bar{t}_i(s), \quad p = 1, \dots, n. \quad (6.2)$$

Покажем, что $\sum_{i=1}^n w_i \bar{t}_i(s^*) \leq \sum_{i=1}^n w_i \bar{t}_i(s)$. Действительно, положим $y_n = w_n$, $y_p = w_p - w_{p+1}$, $p = n-1, n-2, \dots, 2, 1$. Заметим, что все $y_p \geq 0$ и справедливо соотношение $w_i = \sum_{p=1}^n y_p$, $i = 1, \dots, n$.

Учитывая (6.2), имеем

$$\begin{aligned} \sum_{i=1}^n w_i \bar{t}_i(s^*) &= \sum_{i=1}^n \bar{t}_i(s^*) \sum_{p=i}^n y_p = \\ &= \sum_{p=1}^n y_p \sum_{i=1}^p \bar{t}_i(s^*) \leq \sum_{p=1}^n y_p \sum_{i=1}^p \bar{t}_i(s) \leq \sum_{i=1}^n w_i \bar{t}_i(s). \end{aligned}$$

Таким образом, для построения оптимального расписания s^* достаточно пронумеровать требования по невозрастанию весов w_i , положить $s^* = s_1^*$, если $n = kM$, или $s^* = s_2^*$, если $n = kM + r$, $k \geq 0$, $0 < r < M$. Трудоемкость построения s^* не превосходит $O(n \log n + nM)$ операций.

6.4. Покажем, что в случае, когда некоторые $t_{iL} = 0$, а остальные $t_{iL} = 1$, задача построения расписания, минимизирующего суммарное время обслуживания, является NP-трудной. Напомним, что запись $t_{iL} = 0$ означает, что требование $i \in N = \{1, 2, \dots, n\}$ прибором $L \in \mathcal{M} = \{1, 2, \dots, M\}$ не обслуживается.

Рассмотрим соответствующую задачу распознавания: определить, существует ли расписание s^0 (как с прерываниями, так и без прерываний) такое, что $\sum_{i \in N} \bar{t}_i(s^0) \leq y$ для заданного y . Покажем, что к ней сводится

задача о 3-раскраске 4-регулярного графа (см. вводную часть данной главы).

Положим $n = 95|X| + 5|U|$, $N = \{1, 2, \dots, n\}$; $M = 25|X| + 5|U|$, $\mathcal{M} = \{1, 2, \dots, M\}$. Как известно, число ребер произвольного графа равно полусумме степеней его вершин. Так как граф G 4-регулярный, заключаем, что $|U| = 2|X|$, т.е. $n = 3M$.

Множество $\mathcal{M}$ приборов разобьем на два подмножества $\mathcal{M}(U)$ и $\mathcal{M}(X)$. Первое подмножество содержит $5|U|$ приборов, соответствующих ребрам $u_k \in U$ графа G и обозначаемых $U_l(k)$, $1 \leq l \leq 5$, $1 \leq k \leq |U|$. Второе подмножество состоит из $25|X|$ приборов, соответ-

вующих вершинам $x_j \in X$ графа G и обозначаемых $X_{lq}(j)$, $1 \leq l \leq 5$, $1 \leq q \leq 5$, $1 \leq j \leq |X|$.

Множество N требований разобьем на три подмножества $N(W)$, $N(U)$, $N(X)$.

Множество $N(W)$ требований состоит из $25|X|$ требований, обозначаемых $w_{lq}(j)$, $1 \leq l \leq 5$, $1 \leq q \leq 5$, $1 \leq j \leq |X|$. Если в графе G вершине x_j инцидентны ребра $u_{k_1}, u_{k_2}, u_{k_3}, u_{k_4}$, то требование $w_{lq}(j)$, $1 \leq l \leq 5$, $1 \leq q \leq 5$, обслуживается четырьмя приборами $U_l(k_1), U_l(k_2), U_l(k_3), U_l(k_4)$ из множества $\mathcal{M}(U)$, а также прибором $X_{lq}(j)$ и не обслуживается всеми другими приборами.

Множество $N(U)$ требований состоит из $5|U|$ требований, соответствующих ребрам u_k графа G и обозначаемых $u_p(k)$, $1 \leq p \leq 5$, $1 \leq k \leq |U|$. Требование $u_p(k)$ обслуживается только пятью приборами вида $U_l(k)$, $1 \leq l \leq 5$.

Множество $N(X)$ требований состоит из $70|X|$ требований, соответствующих вершинам x_j графа G и обозначаемых $x_{lp}(j)$, $1 \leq l \leq 5$, $1 \leq p \leq 14$, $1 \leq j \leq |X|$. Требование $x_{lp}(j)$ обслуживается пятью приборами $X_{lq}(j)$, $1 \leq q \leq 5$.

Таким образом, каждое требование множества N должно быть обслужено ровно пятью приборами. Длительности обслуживания положим равными единице.

Покажем, что задача о 3-раскраске 4-регулярного графа имеет решение тогда и только тогда, когда в построенной задаче существует расписание s^0 такое, что $\sum_{i \in N} \bar{t}(s^0) \leq 10n$.

1) Пусть граф G 3-раскрашиваем и X_1, X_2, X_3 — найденные подмножества вершин. Разобьем множество требований N на три непересекающихся подмножества N_1, N_2 и N_3 следующим образом. Для каждого r , $1 \leq r \leq 3$, положим $W_r = \{w_{lq}(j) | l = 1, 2, 3, 4, 5, q = 1, 2, 3, 4, 5, x_j \in X_1\}$. Пусть U_r — множество, состоящее из всех требований вида $u_p(k)$, $1 \leq p \leq 5$, таких, что ребро u_k не инцидентно вершине из X_r . Определим также

$$V_1 = \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 1, 2, 3, 4, x_j \in X_1\} \cup$$

$$\cup \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 5, 6, 7, 8, 9, x_j \in X_1 \cup X_3\},$$

$$V_2 = \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 5, 6, 7, 8, 9, x_j \in X_1\} \cup$$

$$\cup \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 1, 2, 3, 4, x_j \in X_2\} \cup$$

$$\cup \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 10, 11, 12, 13, 14, x_j \in X_3\},$$

$$V_3 = \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 10, 11, 12, 13, 14, x_j \in X_1 \cup X_2\} \cup$$

$$\cup \{x_{lp}(j) | l = 1, 2, 3, 4, 5, p = 1, 2, 3, 4, x_j \in X_3\}.$$

Положим $N_r = \overline{W_r} \cup U_r \cup V_r$, $r = 1, 2, 3$. Покажем, что обслуживание всех требований каждого из подмножеств N_1, N_2 и N_3 может быть осуществлено за 5 единиц времени. Доказательство проведем для подмножества N_1 (для других подмножеств доказательство аналогичное).

Зафиксируем некоторый прибор $U_l(k) \in \mathcal{M}(U)$. По построению, он может обслуживать только требования из множеств $N(U)$ и $N(W)$.

Если ребро u_k инцидентно некоторой вершине $x_i \in X_1$, то требования $w_{lq}(i)$, $1 \leq q \leq 5$, принадлежат множеству $W_1 \subset N_1$ и должны обслуживаться прибором $U_l(k)$. С другой стороны, если $u_k = (x_i, x_j)$, то требования

$w_{lq}(j)$, $1 \leq l \leq 5$, $1 \leq q \leq 5$, не принадлежат множеству W_1 , так как вершина x_j не принадлежит X_1 . Так как ребро u_k инцидентно вершине $x_i \in X_1$, то требования $u_p(k)$, $1 \leq p \leq 5$, не принадлежат множеству U_1 , а значит, и множеству N_1 .

Пусть ребро u_k не инцидентно ни одной вершине из X_1 . Тогда прибор $U_i(k)$ не обслуживает ни одного требования множества W_1 . С другой стороны, этот прибор будет обслуживать требования $u_p(k)$, $p = 1, 2, 3, 4, 5$.

Зафиксируем некоторый прибор $X_{lq}(j) \in \mathcal{M}(X)$. По построению, этот прибор может обслуживать только требования из множеств $N(X)$ и $N(W)$.

Предположим, что $x_j \in X_1$. Тогда требование $w_{lq}(j)$ принадлежит W_1 и обслуживается прибором $X_{lq}(j)$. Кроме того, прибор $X_{lq}(j)$ обслуживает требования $x_{lp}(j)$, $1 \leq p \leq 14$. Однако по построению требования $x_{lp}(j)$, $p = 1, 2, 3, 4$, принадлежат множеству X_1 , а оставшиеся требования этого множества не принадлежат.

Если $x_j \notin X_1$, то прибор $X_{lq}(j)$ не обслуживает ни одного требования из множества W_1 . Кроме того, требования $x_{lp}(j)$, $p \in \{1, 2, 3, 4, 10, 11, 12, 13, 14\}$ не принадлежат множеству N_1 . Таким образом, из требований множества N_1 прибор $X_{lq}(j)$ обслуживает только требования $x_{lp}(j)$, $5 \leq p \leq 9$.

Итак, каждый прибор обслуживает ровно пять требований каждого из подмножеств N_1 , N_2 и N_3 . Поскольку суммарная длительность обслуживания каждого требования также равна 5, можно сделать вывод, что $|N_r| = M$, $r = 1, 2, 3$.

Используя методы, описанные в § 3 этой главы, можно построить расписание s^0 обслуживания требований множества N_r в интервале $(5(r-1), 5r]$. При этом обслуживание каждого требования множества N_r завершится в момент времени $5r$, откуда следует, что расписание s^0 является искомым.

2) Покажем, что из существования расписания s^0 такого, что $\sum_{i \in N} \bar{t}_i(s^0) \leq 10n$, следует, что граф G 3-раскрасшиваем.

Рассмотрим произвольное расписание s и предположим, что требования пронумерованы таким образом, что $\bar{t}_1(s) \leq \bar{t}_2(s) \leq \dots \leq \bar{t}_n(s)$.

Пусть $\bar{t}_M(s) = 5$. Тогда $\bar{t}_i(s) = 5$, $1 \leq i \leq M$, и в интервале $(0, 5]$ обслуживаются только требования с номерами, не превосходящими M . Это означает, что $\bar{t}_{M+1}(s) \geq 10$. Аналогично можно доказать, что $\bar{t}_{2M+1}(s) \geq 15$, если $\bar{t}_{M+1}(s) = 10$.

а) Покажем, что для каждого r , $1 \leq r \leq 3$, справедливо следующее соотношение:

$$\sum_{i=(r-1)M+1}^{rM} \bar{t}_i(s) \geq 5rM. \quad (6.3)$$

Зафиксируем некоторое r , $1 \leq r \leq 3$. Если $\bar{t}_{(r-1)M+1}(s) \geq 5r$, то (6.3), очевидно, имеет место. Пусть $\bar{t}_{(r-1)M+1}(s) < 5r$. Исключим из рассмотрения требования с номерами, большими rM (если $r < 3$). В интервале $(\bar{t}_{(r-1)M+1}(s), 5r]$ может быть обслужено не более $M-1$ требований, поэтому суммарная длительность простоя всех приборов в этом

интервале отлична от нуля. Так как суммарная длительность обслуживания каждого требования равна 5 и в интервале $(0, 5r]$ имеются простые, то $\bar{t}_{rM}(s) > 5r$. Следовательно, найдется такое требование $m < rM$, что $\bar{t}_{(r-1)M+1}(s) \leq \bar{t}_{(r-1)M+2}(s) \leq \dots \leq \bar{t}_m(s) < 5r$, $\bar{t}_{m+1}(s) > 5r$. Пусть Δ_r — суммарная длительность простоя всех приборов в интервале $(0, 5r]$. Очевидно, $\Delta_r \geq \sum_{i=(r-1)M+1}^m (5r - \bar{t}_i(s))$. С другой стороны,

требования $m+1, \dots, rM$ могут быть обслужены в интервале $(5r, 5r + \Delta_r]$, откуда следует, что $\sum_{i=m+1}^{rM} (\bar{t}_i(s) - 5r) \geq \Delta_r$. Из полученных соотношений непосредственно следует соотношение (6.3).

б) Покажем, что для расписания s соотношение $\sum_{i \in N} \bar{t}_i(s) \leq 10n$ (или, что то же самое, $\sum_{i \in N} \bar{t}_i(s) \leq 30M$) справедливо тогда и только тогда, когда $\bar{t}_{rM} = 5r$, $1 \leq r \leq 3$.

Достаточность этого утверждения очевидна. Докажем необходимость. Так как (6.3) выполняется для любого расписания и любого r , $1 \leq r \leq 3$, то $\sum_{i \in N} \bar{t}_i(s) \geq 30M$. Покажем, что $\bar{t}_{rM}(s) = 5r$, $1 \leq r \leq 3$, если $\sum_{i \in N} \bar{t}_i(s) = 30M$.

Пусть $\bar{t}_M(s) > 5$. Тогда $\sum_{i=1}^n \bar{t}_i(s) > 5M$ и в силу (6.3) $\sum_{i=1}^n \bar{t}_i(s) > 30M$, что невозможно. Итак, $\bar{t}_M(s) = 5$. В этом случае, как было показано выше, $\bar{t}_{M+1}(s) \geq 10$. Следовательно, $\bar{t}_{2M}(s) \geq 10$. Предположим, что $\bar{t}_{2M}(s) > 10$. Тогда в силу (6.3) $\sum_{i=1}^n \bar{t}_i(s) > 30M$. Поэтому $\bar{t}_{2M}(s) = 10$. Рассуждая аналогичным образом, убеждаемся, что $\bar{t}_{3M}(s) = 15$.

Таким образом, при расписании s^0 , для которого $\sum_{i \in N} \bar{t}_i(s^0) \leq 10n$,

к каждому моменту времени 5, 10, 15 завершается обслуживание ровно M требований. Пусть C_r — множество требований, обслуживание которых завершается в момент времени $5r$, $1 \leq r \leq 3$. Так как $|C_r| = M$, то требования этого множества и только они обслуживаются в интервале $(5(r-1), 5r]$, причем в этом интервале ни один прибор не простаивает.

в) Пусть некоторое требование $w_{l_1 q_1}(j_1)$ из множества $N(W)$ принадлежит множеству C_r . Покажем, что $w_{l_1 q}(j_1) \in C_r$, $q = 1, 2, 3, 4, 5$.

Предположим противное: существует требование $w_{l_1 q_2}(j_1) \notin C_r$, $1 \leq q_2 \leq 5$. В этом случае прибор $X_{l_1 q_1}(j_1)$ обслуживает требование $w_{l_1 q_1}(j_1)$ в интервале $(5(r-1), 5r]$, а прибор $X_{l_1 q_2}(j_1)$ не обслуживает требование $w_{l_1 q_2}(j_1)$ в этом интервале. Кроме того, приборы $X_{l_1 q}(j_1)$, $q = 1, 2, 3, 4, 5$, должны обслуживать требования из множества $N(X)$, принадлежащие C_r . Таких требований может быть не более четырех, поскольку прибор $X_{l_1 q_1}(j_1)$ уже обслуживает одно требование. Но тогда прибор $X_{l_1 q_2}(j_1)$ в интервале $(5(r-1), 5r]$ обслуживает не более четырех требований, что невозможно.

г) Если существуют такие $w_{iq_1}(i) \in C_r$ и $w_{iq_2}(j) \in C_r$, то ребро $u_k = (x_i, x_j) \notin U$. Действительно, если бы граф содержал указанное ребро, то прибор $U_1(k)$ должен был бы обслуживать не менее 10 требований во временном интервале, длина которого равна 5.

Положим $X_r = \{x_i \in X \mid w_{iq}(i) \in C_r\}$, $1 \leq r \leq 3$. Очевидно, X_1 , X_2 и X_3 задают 3-раскраску множества вершин графа G .

Поскольку задача о 3-раскраске 4-регулярного графа NP -полна и реализация ее сведения к рассматриваемой задаче распознавания требует выполнения полиномиального (относительно $|X|$) числа операций, то задача построения расписания, минимизирующего суммарное время обслуживания, в случае, когда все $t_{iL} \in \{0, 1\}$, является NP -трудной.

§ 7. Суммарное время обслуживания. Различные длительности

В систему с нефиксированными маршрутами, содержащую M приборов, в момент времени $d = 0$ поступает множество $N = \{1, 2, \dots, n\}$ требований. Известны длительности обслуживания $t_{iL} \geq 0$ каждого требования $i \in N$ каждым прибором L , $1 \leq L \leq M$. Запись $t_{iL} = 0$ означает, что требование i не обслуживается прибором L . Необходимо построить расписание, минимизирующее суммарное время обслуживания требований $\sum_{i \in N} \bar{t}_i$.

В этом параграфе показано, что рассматриваемая задача является NP -трудной в сильном смысле при $M \geq 2$ (если прерывания в процессе обслуживания запрещены) и при $M \geq 3$ (если прерывания допускаются).

7.1. Пусть в процессе обслуживания каждого требования каждым прибором запрещены прерывания. Покажем, что *задача построения расписания, которому соответствует наименьшее значение суммарного времени обслуживания $\sum_{i \in N} \bar{t}_i$, является NP -трудной в сильном смысле.*

Сформулируем соответствующую задачу распознавания при $M = 2$: имеется два прибора A и B и необходимо определить, существует ли такое расписание s^0 (без прерываний), при котором $\sum_{i \in N} \bar{t}_i(s^0) \leq y$ для любого

заданного числа y . Покажем, что к ней сводится задача о 3-разбиении (см. вводную часть этой главы).

Положим

$$u = 3n_0E + 3n_0 + 1, \quad g = 3n_0E + 3n_0 + 6,$$

$$v = n_0(E + 1)g, \quad f = uv + g + E,$$

$$r = 3n_0E + n_0ug + n_0(n_0 - 1)u(g + E)/2,$$

$$h = r + 1 + (n_0 + 1)g, \quad x = 2(n_0 + h)f,$$

$$n = n_0 + 1 + 2h + n_0u, \quad N = \{1, 2, \dots, n\}.$$

Множество N требований разобьем на 4 группы: F -требования, обозначаемые F_j , $j = 0, \dots, n_0$; X -требования, обозначаемые X_l , $l = 1, \dots, h$; Y -требования, обозначаемые Y_l , $l = 1, \dots, h$; U -требования, среди которых будем различать V -требования, обозначаемые V_{jk} , $j = 1, \dots, n_0$, $k = 1, \dots, u - 3$, и W -требования, обозначаемые W_i , $i = 1, \dots, 3n_0$. Длительность обслуживания требования $i \in N$ приборами A и B обозна-

чим соответственно через a_i и b_i . Положим

$$a_{F_0} = 0, \quad b_{F_0} = g;$$

$$a_{F_j} = f, \quad b_{F_j} = g, \quad j = 1, \dots, n_0;$$

$$a_{X_l} = b_{Y_l} = x, \quad b_{X_l} = a_{Y_l} = 0, \quad l = 1, \dots, h;$$

$$a_{V_{jk}} = 0, \quad b_{V_{jk}} = v, \quad j = 1, \dots, n_0, \quad k = 1, \dots, u-3;$$

$$a_{W_i} = 0, \quad b_{W_i} = v + e_i, \quad i = 1, \dots, 3n_0.$$

Обозначим

$$\bar{F} = \sum_{j=0}^{n_0} (g + jf), \quad \bar{X} = \sum_{l=1}^h (n_0 f + lx),$$

$$\bar{Y} = \sum_{l=1}^h (n_0 f + g + lx), \quad \bar{U} = 3n_0 E + \sum_{j=0}^{n_0-1} \sum_{k=1}^u (g + kf + jv),$$

$$y = \bar{F} + \bar{X} + \bar{Y} + \bar{U}.$$

Покажем, что в построенной задаче расписание s^0 такое, что $\sum_{i \in N} \bar{t}_i(s^0) \leq y$, существует тогда и только тогда, когда имеет решение задача о 3-разбиении.

7.2. Пусть задача о 3-разбиении имеет решение и $N_1^0, N_2^0, \dots, N_{n_0}^0$ — найденные подмножества множества N^0 . Каждое подмножество N_j^0 , $1 \leq j \leq n_0$, состоит из трех элементов $\alpha_{j1}, \alpha_{j2}, \alpha_{j3}$ множества N^0 .

Расписание s^0 можно построить следующим образом. Положим

$$\bar{t}_{F_0} = \bar{t}_{F_0 B} = g, \quad \bar{t}_{F_j A} = jf, \quad \bar{t}_{F_j} = \bar{t}_{F_j B} = jf + g, \quad j = 1, \dots, n_0;$$

$$\bar{t}_{X_l} = \bar{t}_{X_l A} = n_0 f + lx, \quad \bar{t}_{Y_l} = \bar{t}_{Y_l B} = n_0 f + lx + g, \quad l = 1, \dots, h.$$

Пусть, кроме того, в интервале $(g + (j-1)f, jf]$ прибор B обслуживает требования $V_{j1}, V_{j2}, \dots, V_{j, u-3}, W_{\alpha_{j1}}, W_{\alpha_{j2}}, W_{\alpha_{j3}}$ в указанной последовательности, $j = 1, \dots, n_0$.

Нетрудно проверить, что суммарное время обслуживания всех F -, X - и Y -требований равно $\bar{F} + \bar{X} + \bar{Y}$. Вычислим суммарное время обслуживания всех U -требований. По построению имеем

$$\begin{aligned} \sum_{j=1}^{n_0} \sum_{k=1}^{u-3} \bar{t}_{V_{jk}} + \sum_{i=1}^{3n_0} \bar{t}_{W_i} &= \sum_{j=1}^{n_0} \sum_{k=1}^{u-3} (g + (j-1)f + kv) + \\ &+ \sum_{j=1}^{n_0} (g + (j-1)f + (u-2)v + e_{\alpha_{j1}}) + \sum_{j=1}^{n_0} (g + (j-1)f + \\ &+ (u-1)v + e_{\alpha_{j1}} + e_{\alpha_{j2}}) + \sum_{j=1}^{n_0} (g + (j-1)f + uv + e_{\alpha_{j1}} + e_{\alpha_{j2}} + e_{\alpha_{j3}}) = \\ &= \sum_{j=1}^{n_0} \sum_{k=1}^u (g + (j-1)f + kv) + \sum_{j=1}^{n_0} (3e_{\alpha_{j1}} + 2e_{\alpha_{j2}} + e_{\alpha_{j3}}) < \\ &< \sum_{j=0}^{n_0-1} \sum_{k=1}^u (g + jf + kv) + 3n_0 E = \bar{U}. \end{aligned}$$

Таким образом, при построенном расписании s^0 суммарное время обслуживания всех требований множества N не превосходит y .

7.3. Пусть существует расписание s^0 , при котором $\sum_{i \in N} \bar{t}_i(s^0) \leq y$.

Покажем, что в этом случае имеет решение задача о 3-разбиении.

Не нарушая общности, можно считать, что F -требования и X -требования обслуживаются прибором A , а Y -требования — прибором B в такой последовательности, что

$$t_{F_0 A}^0 = \bar{t}_{F_0 A} = 0, \quad \bar{t}_{F_{j-1} A} \leq t_{F_j A}^0, \quad j = 1, \dots, n_0,$$

$$\bar{t}_{X_l A} \leq t_{X_{l+1} A}^0, \quad \bar{t}_{Y_l B} \leq t_{Y_{l+1} B}^0, \quad l = 1, \dots, h.$$

1) Покажем, что при расписании s^0 выполняются соотношения $t_{X_1 A}^0 \geq n_0 f$ и $t_{Y_1 B}^0 \geq n_0 f + g$.

Предположим, что $t_{X_1 A}^0 \geq n_0 f$ и существует требование Y_l , $1 \leq l \leq h$, для которого $t_{Y_l B}^0 < n_0 f + g$. В этом случае $l = 1$, поскольку $b_{Y_l} = x > n_0 f + g$ и, следовательно, $\bar{t}_{Y_l B} > n_0 f + g$, даже если $t_{Y_l B}^0 = 0$.

Суммарное время обслуживания Y -требований можно ограничить снизу величиной

$$\tilde{Y} = x + \sum_{l=2}^h (n_0 f + g + lx) = \bar{Y} - n_0 f - x.$$

Если предположить, что все U -требования обслуживаются в интервале $(0, n_0 f + g]$, то суммарное время обслуживания этих требований можно ограничить снизу величиной $\tilde{U}_1 = \sum_{j=1}^{n_0 u} jv = \bar{U} - r$.

Оценка $\tilde{U}_1$ вычислена в предположении, что длительность обслуживания каждого U -требования, включая W -требования, равна v , и эти требования обслуживаются прибором B непрерывно одно за другим начиная с момента времени 0.

Сделаем предположение, что найдется по крайней мере одно F -требование, например F_{n_0} , для которого $t_{F_{n_0} B} > x + g$. Суммарное время обслуживания F -требований можно ограничить снизу величиной $\tilde{F}_1 = \sum_{j=1}^{n_0-1} jf + x + g = \bar{F} - n_0 f - n_0 g + x$.

Тогда суммарное время обслуживания всех требований множества N не меньше величины $\tilde{F}_1 + \bar{X} + \tilde{Y} + \tilde{U}_1 = y + x - 2n_0 f - (n_0 + 1)g - r$, которая больше y , что невозможно.

С другой стороны, если предположить, что все F -требования обслуживаются прибором B в интервале $(0, n_0 f + g]$, то суммарное время обслуживания F -требований может быть ограничено снизу величиной

$$\tilde{F}_2 = \sum_{j=1}^{n_0} jf = \bar{F} - (n_0 + 1)g.$$

Тогда суммарное время обслуживания U -требований будет не меньше чем $\tilde{U}_2 = \sum_{j=1}^{n_0 u - 1} jv + v + x = \bar{U} - r - n_0 u v + v + x$, поскольку по крайней

мере одно из U -требований будет обслужено не ранее момента времени $x + v$.

Суммарное время обслуживания всех требований множества N будет в этом случае не меньше чем $\tilde{F}_2 + \bar{X} + \tilde{Y} + \tilde{U}_2 = y + x + v - r - n_0 uv - - n_0 f - (n_0 + 2)g$, что невозможно.

Таким образом, при расписании s^0 соотношения $t_{X,A}^0 \geq n_0 f$ и $t_{Y,B}^0 < < n_0 f + g$ одновременно не могут иметь места.

Проводя аналогичные рассуждения, можно показать, что при расписании s^0 одновременно не могут иметь места следующие пары соотношений: $t_{X,A}^0 < n_0 f$ и $t_{Y,B}^0 \geq n_0 f + g$, $t_{X,A}^0 < n_0 f$ и $t_{Y,B}^0 < n_0 f + g$.

2) Покажем, что при расписании s^0 выполняются соотношения $t_{X,A}^0 = = n_0 f$, $t_{Y,B}^0 = n_0 f + g$.

Предположим, что $t_{X,A}^0 > n_0 f$, например $t_{X,A}^0 = n_0 f + 1$. Тогда суммарное время обслуживания всех X -требований окажется не меньше чем $\bar{X} + h$. Если в качестве нижней оценки суммарного времени обслуживания F -требований взять величину $\tilde{F}_2$, U -требований — величину $\tilde{U}_1$, Y -требований — величину $\bar{Y}$, то суммарное время обслуживания множества N требований будет не меньше чем $y + (h - r - (n_0 + 1)g) = y + 1$, что невозможно.

Аналогичным образом доказывается недопустимость соотношения $t_{Y,B}^0 > n_0 f + g$.

Итак, $t_{X,A}^0 = n_0 f$ и $t_{Y,B}^0 = n_0 f + g$. Отсюда, очевидно, следует, что $\bar{t}_{F_j A} = jf$, $j = 1, \dots, n_0$. Кроме того, в интервале $(0, n_0 f + g]$ прибор B не простаивает и обслуживает все F -требования и U -требования. Если бы прибор B простаивал, то обслуживание этим прибором по крайней мере одного из U -требований или F -требований началось бы не ранее момента времени $\bar{t}_{Y,B} = n_0 f + g + x$, что привело бы к тому, что суммарное время обслуживания всех требований превысило бы допустимое значение.

3) Покажем, что расписание s^0 без увеличения суммарного времени обслуживания может быть преобразовано в расписание $\hat{s}$ такое, что $\bar{t}_{F_j A} \leq \bar{t}_{F_j B}$, $j = 0, \dots, n_0$. Число выполняемых при этом операций ограничено полиномом от n .

Пусть p , $0 \leq p \leq n_0$, — наименьший индекс, для которого $\bar{t}_{F_p A} > \bar{t}_{F_p B}$. Докажем существование такого V -требований (обозначаемого через V^*), что $pf + g \leq \bar{t}_{V^* B} \leq pf + g + v$.

Если в интервале $(pf - v - E/2, (p + 1)f]$ прибор B обслуживает некоторое F -требование, то им окажется требование F_i , где $i < p$. Тогда, не увеличивая суммарного времени обслуживания, можно поменять местами требование F_i и непосредственно предшествующее ему требование в последовательности, в которой эти требования обслуживаются прибором B . Описанное преобразование будем повторять до тех пор, пока не придем к расписанию, при котором $\bar{t}_{F_i B} < pf - v - E/2$. При этом прибор в интервале $(pf - v - E/2, (p + 1)f]$ обслуживает только U -требования.

Длина интервала такова, что даже если все W -требования обслуживаются прибором B в этом интервале, то в нем может быть обслужено еще не менее четырех V -требований.

Поскольку длительность обслуживания каждого V -требования меньше длительности обслуживания любого W -требования, можно считать, что в последовательности, в которой прибор B обслуживает требования в интервале $(pf - v - E/2, (p+1)f]$, V -требования предшествуют W -требованиям.

Пусть V' — первое из указанных V -требований, для которого имеет место соотношение $\bar{t}_{V'B} \geq pf$. Из предыдущих рассуждений вытекает, что такое требование существует и следующее за ним требование будет также V -требованием. Очевидно, $\bar{t}_{V'B} < pf + v + g$. Если при этом $\bar{t}_{V'B} > pf + g$, то положим $V^* = V'$. В противном случае в качестве V^* выберем требование, следующее за требованием V' . Нетрудно проверить, что обслуживание требования V^* завершается в интервале $(pf + g, pf + g + v]$.

Используя требование V^* , осуществим дальнейшее преобразование расписания. Уменьшим на величину g время начала обслуживания прибором B всех требований, следующих за требованием F_p до требования V^* включительно. Положим $t_{F_p B}^0 = \bar{t}_{V^* B}$. Поскольку $t_{F_p B}^0 \geq pf + g$, получаем $\bar{t}_{F_p A} \leq t_{F_p B}^0$.

При указанном преобразовании может возникнуть конфликтная ситуация, когда некоторое требование F_j , $1 \leq j < p$, должно обслуживаться прибором B в интервале $((j-1)f, jf]$. В этом случае в последовательности, в которой требования обслуживаются прибором B , выделим требование, непосредственно следующее за F_j . Это требование является V -требованием, в чем нетрудно убедиться, проводя такие же рассуждения, как и при доказательстве существования V^* . Поменяем местами требование F_j с выделенным требованием.

В любом случае получаем расписание s^1 , при котором $\bar{t}_{F_j A} \leq t_{F_j B}^0$, $j = 0, \dots, p$. К расписанию s^1 могут быть применены те же преобразования и т.д., пока не будет получено расписание $\hat{s}$, при котором $\bar{t}_{F_j A} \leq t_{F_j B}^0$, $j = 0, \dots, n_0$. Нетрудно убедиться, что число выполняемых при этом элементарных операций полиномиально зависит от числа F - и U -требований.

Покажем, что при переходе от одного расписания к другому суммарное время обслуживания требований не увеличивается. Для этого, очевидно, достаточно рассмотреть расписания s^0 и s^1 .

Поскольку при исходном расписании s^0 имеем $t_{F_j B}^0 \geq jf$, можно считать, что в каждом временном интервале $(jf, (j+1)f]$, $0 \leq j \leq p$, после выполнения описанных преобразований прибор B обслуживает только одно F -требование, а именно требование F_j . Кроме того, в каждом из этих интервалов прибор B обслуживает не менее чем $(f - g)/(v + E/2)$ U -требований (поскольку длительность обслуживания этим прибором любого U -требования не превосходит $v + E/2$).

Из соотношений $v = n_0(E + 1)g > n_0Eg > n_0Eu > uE/2 - 3E/2$ следует, что $E > uE/2 - v - E/2$. Отсюда $uv + E > uv + uE/2 - v - E/2 = (u - 1) \times (v + E/2)$.

Так как $uv + E = f - g$, приходим к заключению, что число U -требований, обслуживаемых в каждом интервале $(jf, (j+1)f]$, $0 \leq j \leq p$, при-

бором B , больше чем $u - 1$. Другими словами, в интервале $(0, (p + 1)f]$ на каждое обслуженное прибором B F -требование приходится не менее u U -требуваний.

Пусть $\tilde{n}$ — число требований, для которых время начала обслуживания прибором B было уменьшено в процессе преобразования исходного расписания. Тогда среди них окажется не менее $\lfloor \tilde{n}/u \rfloor$ F -требуваний. Для требования F_p при исходном расписании выполнялось соотношение $\bar{t}_{F_p B} < t_{F_p A}^0$, поэтому $\bar{t}_{F_p} = \bar{t}_{F_p A} = pf$. После преобразования расписания имеем $\bar{t}_{F_p A} \leq t_{F_p B}^0$ и $\bar{t}_{F_p} = \bar{t}_{F_p B} \leq pf + v + g$ по построению. Таким образом, момент завершения обслуживания требования F_p увеличился не более чем на $v + g$ единиц времени.

При ликвидации конфликтных ситуаций момент завершения обслуживания каждого из требований F_i , $0 \leq i \leq p$, мог увеличиться на v единиц времени.

Таким образом, чтобы показать, что проведенные преобразования не увеличили суммарное время обслуживания требований, достаточно убедиться в справедливости неравенства

$$\tilde{n}g > v + g + \lfloor \tilde{n}/u \rfloor v.$$

Очевидно, $\tilde{n} \geq u$. Положим $\tilde{n} = ku + z$, где $k \geq 1$ и z , $0 \leq z < u$, — целые числа. Интересующее нас неравенство равносильно неравенству $(ku + z)g \geq v + g + kv$. В результате деления обеих частей этого неравенства на g и выполнения очевидных преобразований получаем

$$ku + z \geq (k + 1)n_0(E + 1) + 1.$$

Учитывая, что $u = 3n_0E + 3n_0 + 1$, получаем

$$3n_0kE + 3n_0k + k + z \geq n_0kE + n_0k + n_0E + n_0 + 1.$$

Последнее неравенство, очевидно, имеет место и равносильно исходному.

4) Покажем, что расписание $\hat{s}$ без увеличения суммарного времени обслуживания требований может быть преобразовано в расписание $\tilde{s}$ такое, что $t_{F_j B}^0 - \bar{t}_{F_j A} < v + E/2$, $j = 0, \dots, n_0$. Число выполняемых при этом операций ограничено полиномом от n .

Если при расписании $\hat{s}$ выполнено $t_{F_p B}^0 - \bar{t}_{F_p A} \geq v + E/2$ для некоторого p , $0 \leq p \leq n_0$, то в последовательности, в которой требования обслуживаются прибором B , выделим требование, непосредственно предшествующее требованию F_p . Это требование является U -требуванием, так как во временном интервале $(pf, (p + 1)f]$ прибор B не обслуживает других F -требуваний, кроме требования F_p . Поменяем местами требование F_p и выделенное U -требувание. Суммарное время обслуживания не увеличится, и поскольку длительность обслуживания любого U -требувания прибором B меньше чем $v + E/2$, то требование F_p по-прежнему будет единственным F -требуванием, обслуживаемым прибором B в интервале $(pf, (p + 1)f]$.

Приводя указанные преобразования необходимое число раз, получаем искомое расписание $\tilde{s}$. Нетрудно убедиться, что число выполняемых при этом операций зависит полиномиально от числа F - и U -требуваний.

5) Покажем, что при расписании $\tilde{s}$ в каждом временном интервале $(\bar{t}_{F_j B}, t_{F_{j+1} B}^0]$, $j = 0, \dots, n_0 - 1$, прибор B обслуживает не более $u + 1$ и не менее $u - 1$ U -требуваний.

Предположим, что существует интервал $(\bar{t}_{F_p B}, t_{F_{p+1} B}^0]$, $0 \leq p \leq n_0 - 1$, в котором число U -требуваний, обслуживаемых прибором B , равно $u - 2$. Длина указанного интервала будет не меньше чем $f - v - E/2$, что достигается при $\bar{t}_{F_p B} = pf + v + E/2$ и $\bar{t}_{F_{p+1} B} = (p + 1)f$. Даже если в этом интервале прибор B будет обслуживать все W -требувания, то суммарная длительность их обслуживания окажется равной $(u - 2)v + n_0 E$. Так как $f - v - E/2 = uv + E/2 - v + g$ и $g + v + E/2 > n_0 E$, то в рассматриваемом интервале прибор B будет простаивать, что невозможно.

Предположим, что существует интервал $(\bar{t}_{F_p B}, t_{F_{p+1} B}^0]$, $0 \leq p \leq n_0$, в котором число U -требуваний, обслуживаемых прибором B , равно $u + 2$. Длина указанного интервала будет не больше чем $f + v - g + E/2$, что достигается при $\bar{t}_{F_p B} = pf + g$ и $t_{F_{p+1} B}^0 = (p + 1)f + v + E/2$. Даже если в указанном интервале прибор B обслуживает только V -требувания, то суммарная длительность их обслуживания окажется равной $(u + 2)v$. Поскольку $v > E/2$, то $(u + 2)v > (u + 1)v + E/2 > f - g + v + E/2$. Это означает, что $u + 2$ U -требуваний не могут быть полностью обслужены в указанном интервале.

6) Покажем, что при расписании $\tilde{s}$ в каждом временном интервале $(\bar{t}_{F_j B}, t_{F_{j+1} B}^0]$, $0 \leq j \leq n_0 - 1$, прибор B обслуживает ровно u U -требуваний.

Предположим, что найдется требование F_p , $0 \leq p \leq n_0$, такое, что в интервале $(0, t_{F_p B}^0]$ число U -требуваний, обслуживаемых прибором B , меньше чем pu . Если учесть, что в указанном интервале прибор B обслуживает также требования F_j , $0 \leq j \leq p - 1$, то суммарная длительность обслуживания прибором B всех требований, предшествующих F_p , окажется не больше чем $pg + (pu - 1)v + n_0 E$. Эта величина, однако, меньше pf , т.е. в интервале $(0, t_{F_p B}^0]$ прибор B простаивает, что невозможно.

Число U -требуваний, обслуживаемых прибором B в интервале $(\bar{t}_{F_j B}, t_{F_{j+1} B}^0]$, $0 \leq j \leq n_0 - 1$, обозначим через c_j . Предположим, что существуют такие p и q , $0 \leq p < q \leq n_0 - 1$, что $c_p = c_q = u + 1$, в то время как $c_j = u$, $j = p + 1, \dots, q - 1$. Тогда число U -требуваний, обслуживаемых прибором B в интервале $(\bar{t}_{F_p B}, t_{F_q B}^0]$, равно $(q + 1 - p)u + 2$. Суммарная длительность обслуживания прибором B всех требований в интервале $(\bar{t}_{F_p B}, t_{F_q B}^0]$ будет не меньше чем $((q + 1 - p)u + 2)v + (q + 1 - p)g$, что больше максимально возможной длины указанного интервала (равной $(q + 1 - p)f + v + E/2$).

Таким образом, в последовательности $c_0, c_1, \dots, c_{n_0-1}$ либо все $c_i = u$, либо эта последовательность содержит по крайней мере одну подпоследовательность вида $c_p = u + 1$, $c_j = u$, $0 \leq p < j < q \leq n_0 - 1$, $c_q = u - 1$.

Покажем, что второй случай не может иметь места. Пусть последовательность $c_0, c_1, \dots, c_{n_0-1}$ содержит ровно одну подпоследовательность

указанного вида. Тогда каждое требование F_j , $p < j \leq q$, не может быть обслужено прибором B ранее момента времени $pf + (j - p)g + ((j - p)u + 1)v = jf + v - (j - p)E$.

Положим $R_j = v - (j - p)E$, $j \in \{p + 1, p + 2, \dots, q\}$ и $R_j = 0$, $j \in \{0, 1, \dots, p, q + 1, q + 2, \dots, n_0\}$.

Суммарное время обслуживания F -требований и U -требований можно ограничить снизу соответственно величинами

$$\tilde{F}_3 = \sum_{j=0}^{n_0} (g + jf + R_j) \quad \text{и} \quad \tilde{U}_3 = \sum_{j=0}^{n_0-1} \sum_{k=1}^{c_j} (g + jf + kv + R_j).$$

Имеем

$$\begin{aligned} \tilde{F}_3 - \bar{F} &= \sum_{j=p+1}^q (g + jf + v - (j - p)E) - \sum_{j=p+1}^q (g + jf) = \\ &= v - (q - p)E + \sum_{j=p+1}^{q-1} (v - (j - p)E), \end{aligned}$$

$$\text{где} \quad \sum_{j=p+1}^{q-1} (v - (j - p)E) = 0 \quad \text{при} \quad q = p + 1.$$

Аналогично

$$\begin{aligned} \tilde{U}_3 - \bar{U} &= -3n_0E + \sum_{j=0}^{n_0-1} \left(\sum_{k=1}^{c_j} (g + jf + kv + R_j) - \right. \\ &- \sum_{k=1}^u (g + jf + kv)) = -3n_0E + \left(\sum_{k=1}^{u+1} (g + kv + pf) - \right. \\ &- \sum_{k=1}^u (g + kv + pf)) + \sum_{j=p+1}^{q-1} \sum_{k=1}^u R_j + \\ &+ \left(\sum_{k=1}^{u-1} (g + kv + qf + R_q) - \sum_{k=1}^u (g + kv + qf) \right) = \\ &= -3n_0E - (q - p)(g + uE) - \sum_{j=p+1}^{q-1} u(j - p)E, \end{aligned}$$

$$\text{где} \quad \sum_{j=p+1}^{q-1} u(j - p)E = 0 \quad \text{при} \quad q = p + 1.$$

Поскольку $v - (q - p)E > (q - p)(g + uE) + 3n_0E$ и $v - (j - p)E > u(j - p)E$, $p < j < q$, то получаем соотношение $\tilde{U}_3 + \tilde{F}_3 > \bar{U} + \bar{F}$. Если суммарное время обслуживания X - и Y -требований ограничить снизу величинами $\bar{X}$ и $\bar{Y}$ соответственно, то суммарное время обслуживания всех требований множества N будет больше чем u , что невозможно.

7) Покажем, что при расписании $\tilde{\pi}$ значение $t_{F_j B}^0 = jf$, $j = 0, \dots, n_0$.

Предположим, что найдется требование F_p , $0 \leq p \leq n_0$, для которого $t_{F_p B}^0 > pf$. Положим для определенности $t_{F_p B}^0 = pf + 1$.

Тогда суммарное время обслуживания F -требований можно ограничить снизу величиной $\bar{F} + 1$.

Нижнюю оценку суммарного времени обслуживания U -требуемых вычислим в предположении, что длительность обслуживания каждого W -требования прибором B равна v . Если бы при расписании $\tilde{s}$ значение $t_{FjB}^0 = jf, j = 0, \dots, n_0$, то суммарное время обслуживания U -требуемых было бы не меньше чем $\bar{U} - 3n_0E$. Поскольку $t_{FpB}^0 = pf + 1$, то указанная нижняя оценка равна $\bar{U} + u - 3n_0E$.

Выбирая в качестве нижних оценок суммарного времени обслуживания X - и Y -требуемых величины $\bar{X}$ и $\bar{Y}$ соответственно и учитывая, что $u + 1 > 3n_0E$, приходим к заключению, что суммарное время обслуживания всех требований больше u , что невозможно.

8) Таким образом, при расписании $\tilde{s}$ значение $t_{FjB}^0 = jf, j = 0, \dots, n_0$, в каждом временном интервале $((j-1)f + g, jf]$, $1 \leq j \leq n_0$, прибор B не простаивает и число обслуживаемых им U -требуемых равно u .

Следовательно, в каждом из указанных интервалов прибор B обслуживает ровно три W -требования, сумма длительностей обслуживания которых равна $3v + E$. Тем самым получаем решение задачи о 3-разбиении.

Таким образом, за $O(n_0^2 E)$ операций реализовано сведение задачи о 3-разбиении к рассматриваемой задаче распознавания.

7.4. Покажем, что задача построения расписания s^* , которому соответствует наименьшее значение суммарного времени обслуживания $\sum_{i \in N} \bar{t}_i(s)$, является NP-трудной в сильном смысле при $M \geq 3$, если в процессе обслуживания каждого требования каждым прибором допускаются прерывания.

Сформулируем соответствующую задачу распознавания: определить, существует ли расписание s^0 (с прерываниями), при котором $\sum_{i \in N} \bar{t}_i(s^0) \leq u$ для заданного числа u . Покажем, что при $M = 3$ к этой задаче сводится задача о 3-разбиении.

Положим $N = N^0 = \{1, 2, \dots, 3n_0\}$, $t_{i1} = e_i$, $t_{i2} = t_{i3} = (E - e_i)/2$, $i = 1, \dots, 3n_0$.

Очевидно, $t_{i1} + t_{i2} + t_{i3} = E$.

Покажем, что в построенной задаче расписание s^0 , при котором $\sum_{i \in N} \bar{t}_i(s^0) \leq u = 3En_0(n_0 + 1)/2$, существует тогда и только тогда, когда имеет решение задачи о 3-разбиении.

1) Пусть задача о 3-разбиении имеет решение и $N_1^0, N_2^0, \dots, N_{n_0}^0$ — найденные трехэлементные подмножества множества N^0 . Тогда для любого j , $1 \leq j \leq n_0$, справедливо $\sum_{i \in N_j^0} t_{iL} = E$, $L = 1, 2, 3$.

Расписание s^0 существует и может быть построено следующим образом. Пусть во временном интервале $((j-1)E, jE]$ обслуживаются только требования множества N_j^0 и требования этого множества не обслуживаются вне указанного интервала.

При помощи одного из алгоритмов, описанных в § 3 гл.3, можно построить расписание обслуживания множества требований N_j^0 во временном интервале $((j-1)E, jE]$, $j = 1, \dots, n_0$. Так как при этом $\bar{t}_i = jE$ для

всех $i \in N_j^0, j = 1, \dots, n_0$, то $\sum_{i \in N} \bar{t}_i = y$. Следовательно, получаем иско-
мое расписание s^0 .

2) Пусть существует расписание s^0 .

а) Покажем, что $\sum_{i \in N} \bar{t}_i(s^0) = y$.

Рассмотрим задачу построения расписания s' обслуживания $3n_0$ требо-
ваний тремя параллельными идентичными приборами, которому соот-
ветствует минимальное значение суммарного времени обслуживания.
Пусть длительность обслуживания каждого требования равна E . Не-
трудно видеть, что $\sum_{i \in N} \bar{t}_i(s') = y$ и $\sum_{i \in N} \bar{t}_i(s') \leq \sum_{i \in N} \bar{t}_i(s^0)$ (см.
§ 6 этой главы). Так как по условию $\sum_{i \in N} \bar{t}_i(s^0) \leq y$, то $\sum_{i \in N} \bar{t}_i(s^0) = y$.

б) Предположим, что при расписании s^0 требования множества N прону-
мерованы таким образом, что $\bar{t}_1 \leq \bar{t}_2 \leq \dots \leq \bar{t}_{3n_0}$. Выберем произволь-
ное $j, 1 \leq j \leq n_0$, и покажем, что $\bar{t}_{3j-2} + \bar{t}_{3j-1} + \bar{t}_{3j} \geq 3jE$.

Обозначим $\bar{t}_{3j-2} = u$ и $\bar{t}_{3j-1} - \bar{t}_{3j-2} = v, \bar{t}_{3j} - \bar{t}_{3j-1} = w$; пусть
 v' и w' соответственно означают суммарную длительность обслуживания
требований $3j-1$ и $3j$ во временном интервале $(0, u]$.

Так как $\bar{t}_{3j-1,1} + \bar{t}_{3j-1,2} + \bar{t}_{3j-1,3} = E$ и $\bar{t}_{3j,1} + \bar{t}_{3j,2} + \bar{t}_{3j,3} = E$, то
 $v + v' \geq E$ и $v + w + w' \geq E$. Кроме того, так как $M = 3$ и $\bar{t}_{3j-2} > \bar{t}_{3j-3}$,
то $3u \geq 3(j-1)E + E + v' + w'$.

Поскольку $\bar{t}_{3j-2} + \bar{t}_{3j-1} + \bar{t}_{3j} = 3u + 2v + w$, имеем

$$\begin{aligned} \bar{t}_{3j-2} + \bar{t}_{3j-1} + \bar{t}_{3j} &\geq 3u + 2E - v' - w' \geq \\ &\geq 3(j-1)E + 3E + v' + w' - (v' + w') = 3jE. \end{aligned}$$

в) Покажем, что задача о 3-разбиении имеет решение. Так как $\bar{t}_{3j-2} +$
 $+ \bar{t}_{3j-1} + \bar{t}_{3j} \geq 3jE, j = 1, \dots, n_0$, то из существования такого j' , что
 $\bar{t}_{3j'-2} + \bar{t}_{3j'-1} + \bar{t}_{3j'} > 3j'E$, следует, что $\sum_{i \in N^0} \bar{t}_i(s^0) > y$.

Следовательно, $\bar{t}_{3j-2} + \bar{t}_{3j-1} + \bar{t}_{3j} = 3jE, j = 1, \dots, n_0$.

Отсюда получаем, что при расписании s^0 в каждом интервале $(3(j-1)E,$
 $3jE], j = 1, \dots, n_0$, обслуживается ровно три требования, суммарная дли-
тельность обслуживания которых прибором 1 равна E . Обозначая через
 N_j^0 множество требований, обслуживаемых в интервале $(3(j-1)E, 3jE],$
 $j = 1, \dots, n_0$, получаем решение задачи о 3-разбиении.

Тем самым за $O(n_0)$ операций реализовано сведение задачи о 3-раз-
биении к рассматриваемой задаче распознавания.

§ 8. Библиографическая справка

Системы с нефиксированными маршрутами изучаются сравнительно
недавно. Принято считать, что первой работой, посвященной этому классу
систем, является статья Т. Гонсалеса и С. Сахни [551]. Однако уже в моно-
графии Р. Беллмана [25, с. 115] приводится формулировка задачи миними-
зации общего времени обслуживания для двухстадийной системы с нефик-

сированными маршрутами. Кроме того, в работе Д. де Верра [829] фактически описан алгоритм построения оптимального по быстродействию расписания с прерываниями, основанный на раскраске ребер двудольного мультиграфа. Насколько известно авторам, задачи построения расписаний для систем с нефиксированными маршрутами до сих пор не нашли отражения в монографической литературе. Не было также обзорных статей, специально посвященных этим задачам, хотя некоторые вопросы затрагивались в обзорах [569, 635, 646, 655]. Ниже предполагается (если не оговорено противное), что множество $N = \{1, 2, \dots, n\}$ требований поступает в систему, состоящую из $M > 1$ приборов, в момент времени $d = 0$. Значения $t_{iL} \geq 0$. Оптимальным называется оптимальное по быстродействию расписание; оно обозначается s^* . Используются обозначения $t^* = \max\{t_{iL} \mid i \in N, L = 1, \dots, M\}$, $t^*(N) = \max\{\sum_{i \in N} t_{iL} \mid L = 1, \dots, M\}$ и r — число отличных от нуля t_{iL} .

8.1. Алгоритм построения расписания s^* при $M = 2$ (см. п. 1.2) предложен Т. Гонсалесом и С. Сахни [551] (см. также [569, 646]). Результаты, приведенные в п. 1.4, получены В.А. Струсевичем. Случай, когда $N_{AB} = \emptyset$ или $N_{BA} = \emptyset$, рассмотрели Т. Масуда, Х. Ишии и Т. Нишида [681]. NP -трудность (в сильном смысле) задачи построения расписания s^* без задержек в обслуживании требований при $M \geq 2$ установлена С. Сахни и Ю. Чо [764].

NP -трудность задачи построения оптимального расписания (без прерываний) при $M \geq 3$ доказана в [551] (см. также [323]). В [551] показано, что эта задача остается NP -трудной, если $M \geq 4$ и каждое требование обслуживается не более чем двумя приборами. Отметим, что в [569, 646] указано, что при переменном числе приборов задача является NP -трудной в сильном смысле.

В [461] показано, что если прерывания запрещены, то при любом расписании s значение $\bar{t}_{\max}(s) \leq t^*(N) + (M-1)t^*$. Из результатов, полученных в [759], следует, что, применяя приемы, описанные в п. 8.4 § 8 гл. 1, можно построить алгоритмы трудоемкости $O(nM)$ операций, позволяющие находить такое расписание s^0 , для которого $\bar{t}_{\max}(s^0)/\bar{t}_{\max}(s^*) \leq [M/2]$.

Попытка оценить устойчивость оптимальных расписаний без прерываний предпринята С.И. Василенко [51].

8.2. Описанные в пп. 2.2 и 2.3 эффективные алгоритмы минимизации общего времени обслуживания при наличии в системе одного или двух доминирующих приборов и запрещении прерываний предложены В.А. Струсевичем [322, 323]. Алгоритм трудоемкости $O(n^2 M^3)$ решения этой задачи в предположении, что $t^*(N) \geq (M^2 + 2M - 1)t^*$ (см. п. 2.5), разработали И. Барани и Т. Фиала [461]. В этой же работе указано, что для решения задачи требуется не более $O(nM^3 + nM^2 \log n)$ операций, если $t^*(N) \geq (8m \log m + 5m)t^*$, где $m = 2^k$, $2^{k-1} < M \leq 2^k$. Отметим, что в [520] предложен алгоритм трудоемкости $O(n^2 M^3)$, если $t^*(N) \geq (16m \log m + 5m)t^*$.

8.3. Подход к решению задачи построения расписания s^* (с прерываниями) в случае целочисленных длительностей обслуживания, основанный на раскраске ребер двудольного мультиграфа (см. п. 3.2), предложили Д. де Верра [829, 830] и А.С. Асратян [12, 13]. В.Г. Визинг [53] показал,

что минимальное число цветов, необходимых для раскраски ребер двудольного мультиграфа, равно максимальной степени вершины. К числу наиболее эффективных алгоритмов нахождения оптимальной реберной раскраски двудольного (мульти)графа $G = (V, E)$, где V и E — множества вершин и (мульти)ребер (мульти)графа G , принадлежат алгоритмы, разработанные Г. Габоу и О. Каривом [529] и имеющие трудоемкости $O(|E| \log^2 |V|)$ и $O(|V^2| \log |V|)$. Они позволяют строить оптимальное расписание для рассматриваемой задачи за $O(r \log^2(n+M))$ и $O((n+M)^2 \times \log(n+M))$ операций соответственно. Заметим, однако, что число прерываний при этом может оказаться неоправданно большим.

Алгоритм нахождения оптимальной обобщенной раскраски ребер двудольного мультиграфа $G = (V, E)$, требующий выполнения не более $O(|V| R \log K)$ операций, где R — число смежных пар вершин в G , а K — наибольшее число ребер, инцидентных одной и той же паре вершин, также предложен в [529]. Используя этот алгоритм, можно построить оптимальное расписание за $O((n+M)r \log t^*)$ операций.

Обзоры других результатов, касающихся реберной раскраски (мульти)графов, приведены в [521, 627].

Подход к решению задачи построения расписания s^* (с прерываниями), основанный на нахождении разложения (3.2) (см. п. 3.4), предложен в [551, 644] (см. также [569]). Утверждение о существовании разложения (3.2) известно в комбинаторном анализе как теорема Биркгофа (см., например, [198, 270]). Обзоры результатов по этой проблематике приведены в [225, 226]. Обобщения теоремы Биркгофа и их приложения к решению задач теории расписаний рассмотрены Д. де Верра [832 — 834].

Указанный подход использован Т. Гонсалесом и С. Сахни [551] для разработки двух алгоритмов построения расписания s^* , трудоемкость которых не превосходит $O(r^2)$ и $O(r(\min\{r, M^2\} + M \log n))$. Основанный на этом же подходе алгоритм из [549] имеет трудоемкость $O(r + \min\{M^4, n^4, r^2\})$.

Отметим, что задача построения оптимального расписания (с прерываниями) имеет не только самостоятельное значение, но и находит применение при решении других задач теории расписаний, в частности при построении оптимального по быстродействию расписания (с прерываниями) для одностадийной системы, состоящей из параллельных приборов различной производительности [643, 644, 833] (см. также [350]).

Задача построения расписания s^* в случае, когда на каждой стадии имеется несколько параллельных приборов (см. п. 3.6), рассмотрена в [648]. Для решения системы линейных соотношений (3.3)–(3.5) можно использовать предложенный Л.Г. Хачияном [380] (см. также [238]) алгоритм, имеющий полиномиальную трудоемкость. Алгоритм построения оптимального по быстродействию расписания (с прерываниями) для одностадийной системы с параллельными идентичными приборами, имеющий трудоемкость $O(n)$ и использующийся в п. 3.6, описан Р. Мак-Нотоном [687] (см. также [350]). Результаты п. 3.8 получены В.А. Струсевичем.

NP -трудность задачи из п. 3.9 доказана в [493].

NP -трудность задачи построения оптимального расписания (с прерываниями) без задержек в обслуживании требований (см. п. 3.10) установлена В.А. Струсевичем и Е.Р. Ерошиной.

Следует отметить, что в терминах построения оптимальных расписаний для системы с нефиксированными маршрутами могут быть сформулированы многие задачи, возникающие при организации спутниковой связи. Так, например, в [116, 120, 606] фактически рассматривается задача построения оптимального расписания с прерываниями; в [661] описан полиномиальный алгоритм построения оптимального расписания, при котором процесс обслуживания каждого требования прерывается и возобновляется ровно один раз. Другие результаты, полученные в этом направлении, можно найти в [455, 472, 554 – 556, 752].

8.4. Задачи построения оптимальных расписаний обслуживания упорядоченных множеств требований рассматривались С.А. Бородичем, В.А. Струсевицем и А.В. Тузиковым. Результат, приведенный в п. 4.1, получен в [38] (см. также [319, 320]). Алгоритм из п. 4.3 предложен в [38].

NP -трудность (в сильном смысле) задач из пп. 4.5 и 4.6 доказана в [37] и [320] соответственно. Алгоритм из п. 4.7 получен в [37]. Обзор результатов по этому направлению содержится в [36].

В обзоре Я. Блажевича, Я. Ленстры и А. Ринной Кана [470], посвященном задачам теории расписаний с ограниченными ресурсами, отмечается, что для построения оптимального расписания (без прерываний) в случае, когда имеется произвольное число видов ресурсов, необходимых для обслуживания каждого требования, достаточно $O(n^3)$ операций, если $M = 2$ и $t_{i1} = t_{i2} = 1, i \in N$. (См. также [471].)

С.А. Бородич [34] рассмотрел ситуацию, когда транспортировка каждого требования i от прибора к прибору осуществляется операторами переноса и требует $\tau_i > 0$ единиц времени. Показано, что независимо от того, допускаются или нет прерывания в процессе обслуживания требований, а также от того, ограничено или нет число операторов переноса, задача построения оптимального расписания является NP -трудной при $M \geq 2$, даже если все $\tau_i = \tau > 0, i = 1, \dots, n$. В некоторых случаях удается за полиномиальное число операций построить либо оптимальное расписание, либо расписание, имеющее ограниченную абсолютную погрешность (см. также [38]).

И. Адир и Н. Амит [418] рассмотрели задачу построения расписания s^* (без прерываний) в предположении, что длительности обслуживания каждого требования зависят от выбираемого маршрута обслуживания. Установлено, что эта задача является NP -трудной при $M \geq 2$; описаны некоторые эффективно разрешимые случаи.

С.А. Бородич [35] рассмотрел ситуацию, когда $M = 2$ и длительности обслуживания являются функциями моментов начала обслуживания требований, т.е. когда $t_{i1} = a_i + \varphi(t_i^0), t_{i2} = b_i + \varphi(t_i^0), i = 1, \dots, n$, где φ — неотрицательная неубывающая функция. Показано, что в случае, когда прерывания допускаются только в целочисленные моменты времени, оптимальное расписание может быть построено в результате выполнения $O(n \log n)$ операций. Если же прерывания допускаются в произвольные моменты времени, то за $O(n \log n)$ операций может быть построено такое расписание s^0 , что $\bar{t}_{\max}(s^0) - \bar{t}_{\max}(s^*) \leq \epsilon$ для любого $\epsilon > 0$. (Здесь s^* — оптимальное расписание в случае, когда $\varphi(t) \equiv 0$.) Если прерыва-

ния не допускаются, то задача является NP -трудной, даже если функция φ является кусочно-постоянной.

8.5. Задача из п. 5.1 рассмотрена Ю. Чо и С. Сахни [493]. Для решения системы (5.1)–(5.3) можно использовать метод, предложенный Л.Г. Хачичяном [380].

Сведение задачи из п. 5.2, 2) к задаче о существовании допустимого потока также проведено в [493]. Основные понятия и терминологию, относящиеся к теории потоков в сетях, читатель может найти в [3, 156, 238, 374, 547]. В частности, в этих работах описан метод, позволяющий свести задачу о существовании допустимого потока к задаче нахождения максимального потока в специальным образом построенной сети. Для нахождения максимального потока А.В. Карзановым [132] (см. также [3, 238, 815]) предложен алгоритм, трудоемкость которого пропорциональна кубу числа вершин сети. Этот метод и позволяет найти допустимый поток в сети $\tilde{G}$ за $O(n^3 + M^3)$ операций.

Задача минимизации максимального временного смещения в случае $M = 2$ рассматривалась Е. Лоулером, Я. Ленстрой и А. Ринной Каном [645]. Если допускаются прерывания, то для решения этой задачи достаточно выполнить $O(n \log n)$ операций (или $O(n)$ операций, если требования упорядочены по неубыванию директивных сроков (см. п. 5.4)). Если же прерывания не допускаются, то задача является NP -трудной в сильном смысле (см. п. 5.6). Отметим, что доказательство этого факта, приведенное в [645], содержит неточности. Соответствующие исправления сделаны в [647]. С.А. Бородич [33] рассмотрел ситуацию, когда на одном приборе прерывания допускаются, а на другом — нет. Предложен алгоритм трудоемкости $O(n^3)$. NP -трудность задачи минимизации числа запаздывающих требований при $M \geq 2$ (см. п. 5.7) доказана С.А. Бородичем [32].

8.6. Задачи минимизации суммарного и взвешенного суммарного времени обслуживания требований единичной длительности рассмотрены В.А. Струсевичем [324]. Тот факт, что решение задачи минимизации суммарного времени обслуживания для одностадийной системы с параллельными идентичными приборами можно искать в классе расписаний без прерываний, установлен Р. Мак-Нотоном [687], а в случае минимизации произвольной ϵ -квазивогнутой неубывающей функции — В.С. Танаевым [341] (см. также [350]). Задача минимизации линейной

формы $\sum_{i=1}^n \omega_i x_{ji}$ на различных подмножествах множества всех перестановок элементов конечного множества рассматривалась многими авторами (см., например, [198, 210, 266 – 268, 326, 327, 329, 350, 351, 378, 780]).

Результат, приведенный в п. 6.4, получен Т. Гонсалесом [550]. Как отмечено в этой работе, сложность задачи не изменится, если расписание искать в классе расписаний без задержек в обслуживании требований.

8.7. Дж. Ачугбю и Ф. Чин [412] показали, что задача минимизации суммарного времени обслуживания при $M \geq 2$ является NP -трудной в сильном смысле, если прерывания запрещены (см. пп. 7.1, 7.2).

Отметим, что используемый при этом подход является развитием метода доказательства аналогичного результата для системы поточного метода

типа [533]. В [412] показано также, что при любом расписании s , при котором в любой момент времени $t \in (0, \bar{t}_{\max}(s)]$ функционирует хотя бы один прибор, имеет место соотношение $\sum_{i=1}^n \bar{t}_i(s) / \sum_{i=1}^n \bar{t}_i(s^*) \leq n$, причем эта оценка достижима. Там же предложен алгоритм трудоемкости $O(nM + n \log n)$ построения такого расписания s^0 , что

$\sum_{i=1}^n \bar{t}_i(s^0) / \sum_{i=1}^n \bar{t}_i(s^*) \leq M$. Эта оценка также достижима.

К. Лиу и Р. Булфин [665] доказали, что указанная задача при $M \geq 3$ является NP -трудной в сильном смысле, если допускаются прерывания (см. п. 7.3). Если потребовать, чтобы процесс обслуживания требования i завершался не позже заранее заданного момента времени D_i , $i = 1, \dots, n$, задача оказывается NP -трудной в сильном смысле и при $M = 2$.

Задачи со случайными длительностями обслуживания требований рассматривались в [504, 741].

Все рассмотренные ранее ситуации допускают естественную интерпретацию в терминах сетевого планирования с ограниченными ресурсами. Для этого достаточно ввести понятие операции как некоторого протекающего во времени процесса, имеющего определенную длительность и требующего определенных ресурсов. Каждые две операции могут быть зависимыми или независимыми в том смысле, что календарное время выполнения одной из них оказывает или не оказывает влияния на календарное время выполнения другой. Каждый прибор можно рассматривать как нескладируемый неделимый ресурс, ограничивающий возможности одновременного выполнения двух или более операций по обслуживанию требований.

Характерные особенности рассматриваемых в этой книге обслуживающих систем позволяют при поиске оптимальных расписаний их функционирования использовать специфические сетевые модели в виде *смешанных (дизъюнктивных) графов*.

Для доказательства NP -полноты рассматриваемой в § 2 задачи распознавания используется следующая задача о выполнимости.

Пусть $X = \{x_1, x_2, \dots, x_m\}$ — множество булевых переменных, принимающих значение 1 (истина) или 0 (ложь). *Дизъюнкцией* над X называется множество *литералов*, т.е. переменных x_i из множества X или их отрицаний $\bar{x}_i$ ($\bar{x}_i = 1$ тогда и только тогда, когда $x_i = 0$). Предполагается, что все переменные в дизъюнкции различны. Дизъюнкция *выполнена* при некотором наборе значений переменных тогда и только тогда, когда при этом наборе хотя бы один из ее литералов принимает значение 1. Множество дизъюнкций $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_\omega\}$ называется *выполнимым* в том и только том случае, когда найдется некоторый набор значений переменных, при котором выполнена каждая дизъюнкция $\Omega_k, k = 1, \dots, \omega$.

Задача о выполнимости состоит в следующем. Даны множество переменных X и множество Ω дизъюнкций над X . Выполнимо ли множество дизъюнкций Ω ?

Если каждая дизъюнкция из Ω содержит k литералов, получаем задачу о k -выполнимости. Эта задача NP -полна при $k \geq 3$ [158], а при $k < 3$ решается за $O(\omega)$ элементарных действий [518].

§ 1. Сетевое представление обслуживающих систем

В этом параграфе приводятся алгоритмы построения расписаний, допустимых относительно взвешенных смешанных графов. Под *смешанным графом* понимается конечный (мульти) граф, содержащий дуги и (неориентированные) ребра.

1.1. Пусть дано конечное множество операций $Q = \{1, 2, \dots, q\}$. Под *операцией* в теории расписаний обычно понимается процесс обслуживания отдельного требования отдельным прибором при некотором конкретном обращении к этому прибору или в более общем случае — процесс одновременного обслуживания нескольких требований несколькими приборами.

Каждой операции $i \in Q$ сопоставляется положительное действительное число t_i — длительность ее выполнения. Момент начала (окончания) операции $i \in Q$ обозначим через t_i^0 (соответственно $\bar{t}_i$). Предполагается, что каждая операция $i \in Q$ выполняется без прерываний и $t_i^0 \geq 0$. Значение $\bar{t}_i$ равно $t_i^0 + t_i$.

Расписание выполнения операций однозначно определяется заданием моментов начала (или окончания) всех операций. Качество расписания характеризуется значением заданной неубывающей по каждому из аргументов действительной функции $F(x_1, x_2, \dots, x_q)$ при $x_i = \bar{t}_i, i = 1, \dots, q$. Расписание считается *оптимальным*, если ему соответствует наименьшее значение указанной функции.

Типичные для задач теории расписаний ограничения на порядок выполнения операций, возможности их одновременного выполнения и т.п. могут быть описаны посредством задания на множестве операций специфических бинарных отношений.

В дальнейшем разность $t_j^0 - t_i^0$ будем обозначать через Δ_{ij} . Если по условию задачи операция i должна предшествовать операции j , то должно выполняться неравенство $\Delta_{ij} \geq t_i$. В ряде случаев приходится принимать во внимание временные затраты, связанные с переналадкой приборов, транспортировкой требований и т.п. Если при переходе от операции i к операции j эти затраты равны $\tau_{ij} > 0$, то должно выполняться неравенство $\Delta_{ij} \geq t_i + \tau_{ij}$. Если разрешено частичное или полное совмещение во времени операций i и j , то должно выполняться неравенство вида $\Delta_{ij} \geq \delta_{ij}$, где δ_{ij} — заданное число, $0 \leq \delta_{ij} < t_i$.

Естественным обобщением этих ситуаций является условие $\Delta_{ij} \geq a_{ij}$, где a_{ij} — заданное действительное (возможно, отрицательное) число. Иногда это условие удобно записывать в виде $\Delta_{ij} \notin (-\infty, a_{ij})$.

Отметим, что если задано несколько неравенств вида $\Delta_{ij} \geq a_{ij}$ (для одних и тех же операций i и j), то достаточно оставить наиболее сильное из них (исключив из рассмотрения остальные). Поскольку $\Delta_{ij} = t_j^0 - t_i^0 = -(t_i^0 - t_j^0) = -\Delta_{ji}$, то неравенства $\Delta_{ij} \geq a_{ij}$ и $\Delta_{ji} \geq a_{ji}$ совместны тогда и только тогда, когда $a_{ij} + a_{ji} \leq 0$.

Если операции i и j могут выполняться в любой последовательности одним и тем же прибором и этот прибор одновременно может выполнять не более одной операции, то должно выполняться либо неравенство $\Delta_{ij} \geq t_i$, либо неравенство $\Delta_{ji} \geq t_j$. Если при переходе от одной операции к другой требуется переналадка прибора, то должно выполняться одно из

неравенств $\Delta_{ij} \geq t_i + \tau_{ij}$ или $\Delta_{ji} \geq t_j + \tau_{ji}$, где τ_{ij} и τ_{ji} — соответствующие длительности переналадок прибора.

В общем случае можно потребовать, чтобы выполнялось хотя бы одно из неравенств $\Delta_{ij} \geq a_{ij}$ или $\Delta_{ji} \geq a_{ji}$, где a_{ij} и a_{ji} — заданные действительные (возможно, отрицательные) числа. Это условие может быть записано в виде $\Delta_{ij} \notin (-a_{ji}, a_{ij})$. Оно является *существенным, если $a_{ij} + a_{ji} > 0$* .

Отметим, что если для одной и той же пары операций i и j задано несколько условий вида $\Delta_{ij} \notin (-a_{ji}, a_{ij})$, то можно найти объединение всех интервалов вида $(-a_{ji}, a_{ij})$. В результате будет получена некоторая совокупность, вообще говоря, новых интервалов, для каждого из которых следует записать соответствующее условие. Совокупность этих условий, очевидно, эквивалентна исходным. В частности, *если все a_{ij} и a_{ji} неотрицательны, то будет получено единственное условие вида $\Delta_{ij} \notin (-a_{ji}, a_{ij})$* .

Аналогично можно поступить и в том случае, когда наряду с условиями вида $\Delta_{ij} \notin (-a_{ji}, a_{ij})$ заданы условия вида $\Delta_{ij} \notin (-\infty, a_{ij})$ и $\Delta_{ij} \notin (-a_{ji}, \infty)$ (последнее условие эквивалентно $\Delta_{ji} \notin (-\infty, a_{ji})$).

Множество операций с заданными на нем бинарными отношениями указанного вида удобно представить взвешенным смешанным (мульти) графом $G = (Q, U, V)$. Здесь Q — множество вершин (операций); U — множество дуг, каждой из которых (i, j) приписан вес a_{ij} ; V — множество ребер, каждому из которых $[i, j]$ сопоставлена пара весов a_{ij} и a_{ji} . Дуга (i, j) определяет условие $\Delta_{ij} \geq a_{ij}$ или, что то же самое, условие $\Delta_{ij} \notin (-\infty, a_{ij})$. Ребро $[i, j]$ определяет условие: либо $\Delta_{ij} \geq a_{ij}$, либо $\Delta_{ji} \geq a_{ji}$ (иначе говоря, $\Delta_{ij} \notin (-a_{ji}, a_{ij})$).

Из приведенных выше рассуждений следует, что *граф G не содержит петель, его подграф (Q, V) может содержать кратные ребра, в то время как подграф (Q, U) не содержит кратных дуг*.

1.2. П р и м е р ы. а) Пусть заданы условия $\Delta_{ij} \notin (-\infty, 3)$, $\Delta_{ij} \notin (7, 10)$, $\Delta_{ij} \notin (9, 15)$, $\Delta_{ij} \notin (18, 22)$, $\Delta_{ij} \notin (21, \infty)$. Объединением перечисленных интервалов является совокупность $\{(-\infty, 3), (7, 15), (18, \infty)\}$. Описанная ситуация представлена на рис. 1.1, а. Вес a_{ij} указан над ребром $[i, j]$, а вес a_{ji} — под ним.

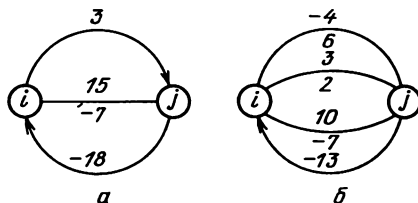


Рис. 1.1

б) На рис. 1.1, б заданы следующие условия: $\Delta_{ij} \notin (-6, -4)$, $\Delta_{ij} \notin (-2, 3)$, $\Delta_{ij} \notin (7, 10)$, $\Delta_{ij} \notin (13, \infty)$.

в) Условия $\Delta_{ij} \notin (-\infty, -7)$, $\Delta_{ij} \notin (-8, -1)$, $\Delta_{ij} \notin (-2, 2)$, $\Delta_{ij} \notin (1, 8)$, $\Delta_{ij} \notin (5, \infty)$ противоречивы, поскольку объединением указанных интервалов является $(-\infty, \infty)$.

1.3. Как уже отмечалось, расписание выполнения операций множества Q однозначно определяется заданием моментов $t_i^0 \geq 0$ их начала (либо моментов $\bar{t}_i > 0$ их завершения).

Допустимое относительно графа $G = (Q, U, V)$ расписание должно удовлетворять следующим условиям:

а) $t_j^0 - t_i^0 \geq a_{ij}$ для всех $(i, j) \in U$;

б) $t_j^0 - t_i^0 \geq a_{ij}^k$ либо $t_i^0 - t_j^0 \geq a_{ji}^k$ для всех $[i, j]^k \in V$, где индекс k отражает возможное наличие в графе G кратных ребер.

Если дано некоторое (допустимое) расписание, то для каждого ребра $[i, j]^k \in V$ известно, какое именно из двух указанных в условии б) неравенств имеет место. Если справедливо первое неравенство, то заменим ребро $[i, j]^k$ дугой $(i, j)^k$, приписав ей вес a_{ij}^k . В противном случае заменим ребро $[i, j]^k$ дугой $(j, i)^k$, приписав ей вес a_{ji}^k . Удалим все кратные дуги вида (i, j) за исключением одной, имеющей наибольший вес.

Обозначим через $\tilde{P}(G)$ множество всех ориентированных графов, порождаемых смешанным графом $G = (Q, U, V)$ в результате замены каждого ребра $[i, j]^k \in V$ дугой $(i, j)^k$ с весом a_{ij}^k или дугой $(j, i)^k$ с весом a_{ji}^k и последующего удаления всех кратных дуг за исключением дуг наибольшего веса для каждого набора дуг вида (i, j) .

Каждое (допустимое относительно G) расписание однозначно определяет некоторый граф из множества $\tilde{P}(G)$. Обратное, вообще говоря, не верно. Покажем, что для существования расписания, допустимого относительно некоторого графа (Q, U') из $\tilde{P}(G)$, необходимо и достаточно, чтобы этот граф не содержал контуров положительного веса. Под весом контура или пути будем понимать, как обычно, сумму весов дуг, входящих в этот контур или путь.

Действительно, если в графе (Q, U') содержится контур $(i_1, i_2), (i_2, i_3), \dots, (i_{r-1}, i_r), (i_r, i_1)$ веса v , то должны выполняться соотношения

$$t_{i_{k+1}}^0 - t_{i_k}^0 \geq a_{i_k i_{k+1}}, \quad k = 1, \dots, r-1, \quad t_{i_1}^0 - t_{i_r}^0 \geq a_{i_r i_1}.$$

Просуммировав эти соотношения, получаем $v \leq 0$.

Пусть граф (Q, U') не содержит контуров положительного веса. Для каждой упорядоченной пары вершин i и j выделим все пути из вершины i в вершину j и выберем среди них путь наибольшего веса. Этот вес обозначим v_{ij} . Если в графе (Q, U') не существует пути из i в j , то положим $v_{ij} = -\infty$. Поскольку в графе (Q, U') не содержится контуров положительного веса, то значения $v_{ij} < \infty$ для всех пар вершин $i, j \in Q$.

Для каждой вершины i положим $t_i^0 = \max_{1 \leq j \leq q} \{0, v_{ji}\}$. Значения t_i^0 , $i = 1, \dots, q$, очевидно, удовлетворяют условиям $t_j^0 - t_i^0 \geq a_{ij}$ для каждой дуги $(i, j) \in U'$ и тем самым определяют искомое расписание.

Нетрудно видеть, что в этом случае графу (Q, U') соответствует бесконечно много (допустимых) расписаний. Построенное выше расписание принято называть *активным*, оно однозначно определено для каждого графа (Q, U') , не содержащего контуров положительного веса. Поскольку при активном расписании выполнение операций начинается в *наиболее ранние сроки*, то оно является наилучшим среди всех допустимых относительно графа (Q, U') расписаний.

Таким образом, существует взаимно однозначное соответствие между множеством всех активных расписаний, допустимых относительно графа G , и множеством $\tilde{P}(G)$ всех графов из множества $\tilde{P}(G)$, которые не содержат контуров положительного веса. При поиске оптимального расписания дос-

таточно выделить из множества ориентированных графов, порождаемых смешанным графом G , все графы, не содержащие контуров положительного веса, построить для каждого из этих графов активное расписание и выбрать среди них наилучшее (относительно заданного критерия оптимальности).

1.4. Пусть дан граф $G = (Q, U, V)$, у которого $U \neq \emptyset$, $V = \emptyset$ и $a_{ij} > 0$ для всех дуг $(i, j) \in U$.

Этот граф принято называть *временной моделью*, а его графическое изображение — *сетевым графиком*.

Поскольку все $a_{ij} > 0$, то в графе G могут содержаться контуры только положительного веса. Следовательно, для существования расписания, допустимого относительно графа G , необходимо и достаточно, чтобы он не содержал контуров.

Покажем, что в рассматриваемом случае активное расписание (допустимое относительно бесконтурного графа G) может быть построено в результате выполнения не более чем $O(q^2)$ элементарных действий.

За один просмотр списка дуг графа G сформируем списки $A(i) = \{j \mid (i, j) \in U\}$ потомков и списки $B(i) = \{j \mid (j, i) \in U\}$ предшественников каждой вершины $i \in Q$. Распределим вершины графа G по рангам. Для этого каждой вершине $i \in Q$ сопоставим признак g_i , равный числу вершин из множества $B(i)$, ранг которых еще не определен. Вначале положим $g_i = |B(i)|$ для всех $i = 1, \dots, q$.

К нулевому рангу отнесем вершины i , для которых $g_i = 0$ (если $g_i \neq 0$ для всех $i = 1, \dots, q$, то в графе G содержится контур). Для каждой вершины i нулевого ранга уменьшим на единицу значение g_j , если $j \in A(i)$. В результате значение g_j для некоторых вершин $j \in Q$ станет равным нулю (в противном случае в графе G содержится контур). Все такие вершины j отнесем к первому рангу и уменьшим на единицу значения g_k для всех $k \in A(j)$. Процесс продолжается до тех пор, пока все вершины множества Q не будут распределены по рангам.

Для всех вершин i нулевого ранга положим $t_i^0 = 0$. Вычисление значений t_j^0 для остальных вершин $j \in Q$ выполним последовательно в порядке убывания рангов вершин по формуле

$$t_j^0 = \max_{i \in B(j)} \{t_i^0 + a_{ij}\}.$$

Нетрудно видеть, что полученные значения t_i^0 , $i = 1, \dots, q$, определяют искомое расписание, и *трудоемкость описанного алгоритма не превышает $O(|U|)$ для связного графа G , $O(|U| + q)$ для несвязного графа или, в общем случае, $O(q^2)$.*

1.5. Пусть дан граф $G = (Q, U, V)$, у которого $U \neq \emptyset$, $V \neq \emptyset$ и веса всех дуг и ребер положительны.

Как уже отмечалось, в этом случае совокупность нескольких условий вида $\Delta_{ij} \notin (-a_{ji}, a_{ij})$ эквивалентна единственному условию того же вида, и следовательно, граф $G = (Q, U, V)$ не содержит кратных ребер. Пара условий $\Delta_{ij} \notin (-\infty, a_{ij})$ и $\Delta_{ij} \notin (-a'_{ji}, a'_{ji})$ эквивалентна условию $\Delta_{ij} \notin (-\infty, \max\{a_{ij}, a'_{ji}\})$, а пара условий $\Delta_{ij} \notin (-\infty, a_{ij})$ и $\Delta_{ji} \notin (-\infty, a_{ji})$ — условию $\Delta_{ij} \notin (-\infty, \infty)$. Таким образом, если $(i, j) \in U$, то $(j, i) \notin U$ и $[i, j] \notin V$.

Поскольку веса дуг множества U положительны, то для существования расписания, допустимого относительно графа G , необходимо, чтобы подграф (Q, U) графа G не содержал контуров. Покажем, что это условие является и достаточным.

Пусть граф (Q, U) не содержит контуров. Для построения бесконтурного графа, принадлежащего множеству $\tilde{P}(G)$, воспользуемся следующим преобразованием графа G .

Пусть G^l — смешанный граф, у которого $l \geq 0$ вершин отмечено и существует неотмеченная вершина i , в которую не заходит ни одной дуги, исходящей из неотмеченной вершины. Отметим вершину i , и в случае, если в графе G^l есть ребра, инцидентные этой вершине, заменим их исходящими из нее дугами. Полученный в результате граф обозначим через G^{l+1} .

Нетрудно убедиться, что бесконтурный граф множества $\tilde{P}(G)$ может быть получен из исходного графа $G^0 = G$ в результате конечного числа указанных преобразований. Если для каждого графа G^l рассматривать все возможные варианты выбора вершины i , то получим множество $P(G)$ всех бесконтурных ориентированных графов, порождаемых смешанным графом G в результате замены каждого ребра $[i, j] \in V$ дугой (i, j) с весом a_{ij} или дугой (j, i) с весом a_{ji} .

Трудоемкость описанного алгоритма, очевидно, не превосходит $O(|U| + |V| + q)$ и, следовательно, общая трудоемкость построения активного расписания (допустимого относительно графа G) не превосходит $O(q^2)$. Для того чтобы получить все активные расписания выполнения операций множества Q , требуется не более чем $O(q^2|P(G)|)$ элементарных действий. Алгоритмы вычисления и оценки величины $|P(G)|$ рассматриваются в следующем параграфе.

1.6. Пример. Рассмотрим задачу построения оптимального расписания обслуживания требований множества $N = \{1, 2\}$ приборами A, B, C и D . В табл. 1.1 и 1.2 § 1 гл. 2 заданы маршруты $L^1 = (A, B, C, D, A)$ и $L^2 = (A, B, A, C, A, C)$ прохождения приборов при обслуживании первого и второго требования соответственно и указаны длительности выполнения всех операций.

Пусть $Q_1 = \{1, 2, \dots, 5\}$ и $Q_2 = \{6, 7, \dots, 11\}$ — множества операций по обслуживанию первого и второго требований соответственно. Операция 1 состоит в обслуживании требования 1 прибором A , операция 2 — в обслуживании требования 1 прибором B и т.д., операция 11 состоит в обслуживании требования 2 прибором C . Введем в рассмотрение фиктивную операцию 12 — завершение обслуживания обоих требований.

На рис. 1.2, а представлен смешанный граф $G = (Q, U, V)$. Множество операций $Q = Q_1 \cup Q_2 \cup \{12\}$. Множество дуг U определяет последовательности выполнения операций (маршруты L^1 и L^2 обслуживания требований). Множество ребер V задает невозможность одновременного обслуживания двух требований одним и тем же прибором. Каждой дуге $(i, j) \in U$ приписан вес a_{ij} , равный длительности t_i выполнения операции $i \in Q$. Каждому ребру $[i, j] \in V$ сопоставлена пара весов $a_{ij} = t_i$ и $a_{ji} = t_j$ (веса ребер на рисунке не указаны).

На рис. 1.2, б представлен один из графов множества $P(G)$. Используя алгоритм, описанный в п. 1.4 этого параграфа, построим активное расписание, допустимое относительно этого графа. Построенное расписание яв-

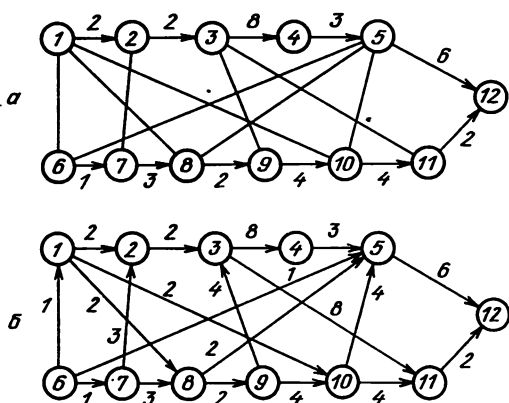


Рис. 1.2

ляется оптимальным для целевой функции $F(x_1, x_2, \dots, x_{12}) = 30x_5 + x_{11}^2$ (см. § 1, гл. 2).

1.7. Пусть дан граф $G = (Q, U, V)$, у которого $U \neq \emptyset$, $V = \emptyset$ и не для всех дуг $(i, j) \in U$ веса a_{ij} положительны.

Этот граф обычно называют *обобщенной временной моделью*, а его графическое изображение — *обобщенным сетевым графиком*.

Если граф G не содержит контуров, то для построения активного расписания (допустимого относительно графа G) можно воспользоваться алгоритмом трудоемкости $O(q^2)$, описанным в п. 1.4 этого параграфа. Если граф G содержит контуры, то можно воспользоваться описанным ниже алгоритмом (трудоемкости $O(q^3)$), который позволяет построить допустимое активное расписание либо убедиться в невозможности его построения (из-за наличия в графе G контуров положительного веса).

Как показано в п. 1.3, если в графе G нет контуров положительного веса, то искоемое (активное) расписание определяется значениями $t_i^0 = \max_{1 \leq j \leq q} \{0, v_{ji}\}$, где, как и прежде, v_{ji} — максимальный вес пути из вершины j в вершину i графа G (значение v_{ji} равно $-\infty$, если не существует пути из j в i).

Дугу (i, j) будем называть *базисной*, если ее вес равен v_{ij} . Для вычисления значений v_{ij} для всех упорядоченных пар вершин i и j из множества Q будем последовательно добавлять в граф G базисные дуги.

Рассмотрим квадратную матрицу порядка q с элементами

$$a_{ij}^{[0]} = \begin{cases} a_{ij}, & \text{если } (i, j) \in U, \\ 0, & \text{если } i = j, \\ -\infty, & \text{если } (i, j) \notin U, i \neq j. \end{cases}$$

Последовательно для каждого индекса $l = 1, 2, \dots, q$ вычислим значения элементов матрицы $\|a_{ij}^{[l]}\|$, используя соотношение

$$a_{ij}^{[l]} = \max \{a_{ij}^{[l-1]}, a_{il}^{[l-1]} + a_{lj}^{[l-1]}\}. \quad (1.1)$$

Каждая матрица $\|a_{ij}^{[l]}\|$, $1 \leq l \leq q$, однозначно определяет граф $(Q, U^{[l]})$ следующим образом: если $a_{ij}^{[l]} > -\infty$, то дуга (i, j) с весом $a_{ij}^{[l]}$ принадлежит множеству $U^{[l]}$, в противном случае $(i, j) \notin U^{[l]}$.

Очевидно, что трудоемкость построения всех матриц $\|a_{ij}^{[l]}\|$, $1 \leq l \leq q$, не превосходит $O(q^3)$.

Покажем, что если $a_{ij}^{[l]} = 0$ для всех $i = 1, \dots, q$, то значение $a_{ij}^{[q]} = v_{ij}$ при $i \neq j$ и $a_{ij}^{[q]} > -\infty$. Рассмотрим произвольную пару вершин $i, j \in Q$, для которой $a_{ij}^{[q]} > -\infty$. Если дуга (i, j) в графе G является базисной, то согласно (1.1) $a_{ij}^{[l]} = v_{ij}$ для всех $l = 1, \dots, q$.

Пусть $\mu = (i = i_1, i_2, \dots, i_{r-1}, i_r = j)$ — максимальный по весу путь из вершины i в вершину j в графе G , $r > 2$ и $i_k = \min\{i_2, \dots, i_{r-1}\}$. Очевидно, что все дуги пути μ являются базисными, и их вес в графе G и в графах $(Q, U^{[l]})$, $l = 1, \dots, q$, один и тот же.

При вычислении значения $a_{i_{k-1}i_{k+1}}^{[t_k]}$ по формуле (1.1) имеют место соотношения

$$a_{i_{k-1}i_{k+1}}^{[t_k-1]} \leq a_{i_{k-1}i_k}^{[t_k-1]} + a_{i_ki_{k+1}}^{[t_k-1]} = a_{i_{k-1}i_k}^{[t_k]} + a_{i_ki_{k+1}}^{[t_k]}.$$

Следовательно, в графе $(Q, U^{[t_k]})$ содержится дуга (i_{k-1}, i_{k+1}) веса $a_{i_{k-1}i_{k+1}}^{[t_k]} = a_{i_{k-1}i_k}^{[t_k]} + a_{i_ki_{k+1}}^{[t_k]}$, которая является базисной (в противном случае путь μ не был бы максимальным по весу).

Рассмотрим путь, полученный из μ в результате замены пары дуг (i_{k-1}, i_k) и (i_k, i_{k+1}) дугой (i_{k-1}, i_{k+1}) . Этот путь также имеет вес v_{ij} , но состоит из меньшего числа дуг по сравнению с путем μ .

Повторяя эти рассуждения конечное число раз, получим путь веса v_{ij} из вершины i в вершину j , состоящий из единственной (базисной) дуги, и $a_{ij}^{[q]} = v_{ij}$.

Если при некоторых $l \leq q$ и $i \in Q$ значение $a_{ii}^{[l]} > 0$, то в графе $(Q, U^{[l]})$, а следовательно, и в графе G содержится контур положительного веса, проходящий через вершину i .

1.8. Использование неположительных весов дуг и ребер позволяет естественным образом описывать многие типичные для задач теории расписаний ситуации. В частности, можно задавать директивные сроки обслуживания

требований, ограничения на суммарную длительность выполнения операций, необходимость обслуживания требований без задержек или без простоев приборов и т.п. Рассмотрим несколько примеров.

На рис. 1.3,а представлено требование, которое необходимо обслужить в течение интервала длительности t . Иными словами, общее время выполнения множества операций $\{1, 2, 3\}$ по обслуживанию этого требования не должно быть больше t . Если $t = t_1 + t_2 + t_3$, то требование будет обслуживаться непрерывно без задержек при любом (допустимом) расписании.

Выполнение операций, представленных на рис. 1.3,б, должно быть начато одновременно.

На рис. 1.3,в задан момент d_k поступления и директивный срок D_k обслуживания требования k . Операции по обслуживанию требования k обозначены 1, 2, 3 и 4. Фиктивная операция 0 означает начало обслуживания всех требований, значение $t_0^0 = 0$.

На рис. 1.3,г представлена ситуация, в которой прибор не должен простаивать при выполнении операций множества $\{1, 2, 3\}$. Предполагается, что значения $a_{ij} = t_i$ и $a_{ji} = t_j$ для каждого ребра $[i, j] \in \{[1, 2], [1, 3], [2, 3]\}$ и значение $t = t_1 + t_2 + t_3$.

1.9. Пусть дан граф $G = (Q, U, V)$, у которого $U = \phi$ и a_{ij} — произвольные числа.

В этом случае построение расписания, допустимого относительно графа G , не вызывает особых затруднений.

Действительно, если $V = \phi$, то положим $t_i^0 = 0$ для всех $i \in Q$. Пусть $V \neq \phi$. Используя алгоритм, описанный в п. 1.5, построим бесконтурный граф $(Q, U') \in P(G)$, где $|U'| = |V|$ (нетрудно видеть, что этот алгоритм можно применять и в случае, когда граф G содержит кратные ребра). Для построения активного расписания, допустимого относительно графа (Q, U') и графа G , воспользуемся алгоритмом, описанным в п. 1.4. Общая трудоемкость построения этого расписания не превосходит $O(|V| + q)$.

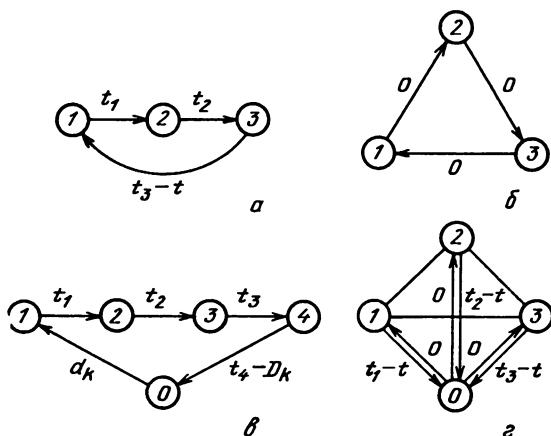


Рис. 1.3

1.10. Перейдем к рассмотрению общего случая. Пусть дан граф $G = (Q, U, V)$, у которого $U \neq \emptyset$, $V \neq \emptyset$ и a_{ij} — произвольные числа.

Поскольку задача проверки существования расписания, допустимого относительно графа G , сводится к задаче проверки существования во множестве $\tilde{P}(G)$ графа без контуров положительного веса, то для существования указанного расписания достаточно, чтобы граф (Q, U) не содержал контуров, и необходимо, чтобы этот граф не содержал контуров положительного веса.

Покажем, что при наличии в графе (Q, U) контуров неположительного веса задача проверки существования допустимого расписания является NP -трудной в сильном смысле даже в том случае, когда все веса положительны за исключением одного, приписанного некоторой дуге графа G .

В § 2 гл. 1 доказана NP -трудность в сильном смысле следующей задачи распознавания. В обслуживающую систему, состоящую из трех приборов A, B и C , в момент времени d поступают требования множества $N = \{1, 2, \dots, n\}$. Каждое требование $k \in N$ обслуживается приборами в одной и той же последовательности (A, B, C) . Необходимо определить, существует ли расписание s^0 (без прерываний) обслуживания требований множества N такое, что общее время обслуживания требований $\bar{t}_{\max}(s^0)$ не превосходит заданного положительного числа y . Можно считать, что $d > 0$, хотя доказательство NP -трудности в сильном смысле этой задачи приведено для $d = 0$.

Введем в рассмотрение смешанный граф (Q, U^0, V) , представленный на рис. 1.4. Здесь Q — множество операций по обслуживанию требований приборами. Операциям по обслуживанию требования $k \in N$ приборами

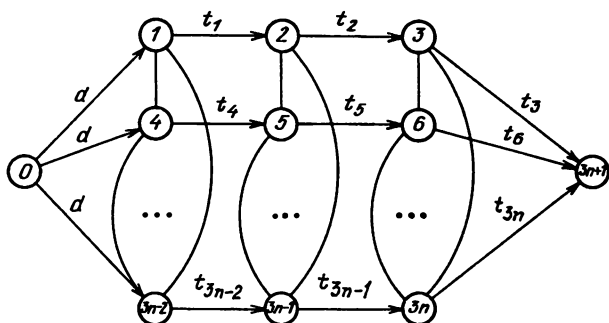


Рис. 1.4

A, B и C присвоены номера $3k - 2$, $3k - 1$ и $3k$ соответственно. Фиктивные операции 0 и $3n + 1$ означают начало и окончание обслуживания требований N . Множество дуг U^0 задает маршрут прохождения приборов при обслуживании каждого требования, а множество ребер V указывает на невозможность одновременного выполнения двух и более операций одним прибором. Каждой дуге $(i, j) \in U$ приписан вес $a_{ij} = t_i$, а каждому ребру $[i, j] \in V$ — пара весов $a_{ij} = t_i$ и $a_{ji} = t_j$.

Очевидно, что для построения графа (Q, U^0, V) требуется не более $O(n^2)$ элементарных действий.

Через $G = (Q, U, V)$ обозначим граф, полученный из графа (Q, U^0, V) в результате добавления дуги $(3n+1, 0)$ с отрицательным весом $a_{3n+1,0} = -\gamma$.

Нетрудно видеть, что расписание s^0 существует тогда и только тогда, когда множество $\tilde{P}(G)$ содержит граф без контуров положительного веса. Следовательно, задача проверки существования допустимого относительно графа G расписания является NP -трудной в сильном смысле.

§ 2. Смешанные графы

В данном параграфе исследуются свойства смешанных графов, которые можно использовать для построения допустимых относительно этих графов расписаний.

2.1. Пусть $G = (Q, U, V)$ — конечный смешанный граф без петель и кратных дуг. Заменяем в G каждое ребро $[i, j]^k \in V$ парой дуг $(i, j)^k$ и $(j, i)^k$. Индекс k отражает возможность наличия в графе G кратных ребер вида $[i, j]$. Полученный ориентированный граф обозначим через $\tilde{G}$, а множество всех его контуров — через $\tilde{H}$.

Каждый из контуров множества $\tilde{H}$ в общем случае содержит дуги двух видов: собственно дуги множества U и дуги $(i, j)^k$, которым в смешанном графе G соответствует ребро $[i, j]^k \in V$. В последнем случае будем говорить, что контур содержит ребро $[i, j]^k$ независимо от того, содержит ли этот контур дугу $(i, j)^k$ или дугу $(j, i)^k$.

Обозначим через $P(G, H)$ множество всех ориентированных графов, порождаемых смешанным графом G в результате замены каждого ребра $[i, j]^k \in V$ одной из дуг $(i, j)^k$ или $(j, i)^k$ и не содержащих ни одного контура из заданного множества $H \subseteq \tilde{H}$. Мощность множества $P(G, H)$ обозначим через $\rho(G, H)$. Очевидно, что $\rho(G, \emptyset) = |\tilde{P}(G)| = 2^{|V|}$ и $\rho(G, \tilde{H}) = |P(G)|$.

Покажем, что задача проверки неравенства $\rho(G, H) > 0$ является NP -полной даже в том случае, когда в графе G нет кратных ребер и каждый контур множества H содержит не более трех ребер и не более трех дуг. Для этого построим полиномиальное сведение к ней задачи о 3-в-ы-полнимо-сти (см. вводную часть данной главы).

Будем использовать следующее обозначение. Пусть $\nu \in \{0, 1\}$, тогда

$$x^\nu = \begin{cases} \bar{x} & \text{при } \nu = 0, \\ x & \text{при } \nu = 1. \end{cases}$$

Рассмотрим множество дизъюнкций $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_\omega\}$, содержащих по три литерала каждая: $\Omega_k = \{x_{k_1}^{\nu_{k_1}}, x_{k_2}^{\nu_{k_2}}, x_{k_3}^{\nu_{k_3}}\}$, где $\nu_{k_l} \in \{0, 1\}$, $x_{k_l} \in X$, $l \in \{1, 2, 3\}$, $k = 1, \dots, \omega$.

Построим смешанный граф (Q, U^0, V) следующим образом. Каждой переменной $x_l \in X$ поставим в соответствие пару вершин x_l^0, x_l^1 и ребро $[x_l^0, x_l^1]$. Каждой дизъюнкции $\Omega_k, k = 1, \dots, \omega$, сопоставим три дуги: $(x_{k_1}^{1-\nu_{k_1}}, x_{k_2}^{\nu_{k_2}})$, $(x_{k_2}^{1-\nu_{k_2}}, x_{k_3}^{\nu_{k_3}})$ и $(x_{k_3}^{1-\nu_{k_3}}, x_{k_1}^{\nu_{k_1}})$.

В построенном графе нет петель и кратных ребер. Через $G = (Q, U, V)$ обозначим граф, полученный из (Q, U^0, V) в результате отождествления кратных дуг. Каждой дизъюнкции $\Omega_k, k = 1, \dots, \omega$, соответствует замкнутый путь

$$\mu_k = (x_{k_1}^{\nu_{k_1}}, x_{k_1}^{1-\nu_{k_1}}, x_{k_2}^{\nu_{k_2}}, x_{k_2}^{1-\nu_{k_2}}, x_{k_3}^{\nu_{k_3}}, x_{k_3}^{1-\nu_{k_3}}, x_{k_1}^{\nu_{k_1}}).$$

Поскольку в Ω_k все переменные различны, то μ_k — простой замкнутый путь, т.е. контур.

Положим $H = \{\mu_1, \mu_2, \dots, \mu_\omega\}$ и рассмотрим задачу проверки неравенства $\rho(G, H) > 0$. Трудоемкость построения графа G и множества контуров H , очевидно, ограничена полиномом от m и ω .

Покажем, что если множество дизъюнкций Ω выполнимо, то $\rho(G, H) > 0$. Пусть существует набор значений переменных $x_l = a_l, a_l \in \{0, 1\}, l = 1, \dots, m$, при котором выполнены все дизъюнкции $\Omega_k, k = 1, \dots, \omega$. Иными словами, для каждого $k = 1, \dots, \omega$ существует индекс $l \in \{1, 2, 3\}$ такой, что $x_{k_l}^{\nu_{k_l}} = 1$ при $x_{k_l} = a_{k_l}$.

По набору значений переменных $a_1, a_2, \dots, a_m$ построим ориентированный граф $(Q, U') \in P(G, \phi)$, положив $U' = U \cup \{(x_l^0, x_l^1) | a_l = 1\} \cup \{(x_l^1, x_l^0) | a_l = 0\}$. Граф (Q, U') не содержит контура $\mu_k, k = 1, \dots, \omega$, поскольку множество U' не содержит дуги $(x_{k_l}^{\nu_{k_l}}, x_{k_l}^{1-\nu_{k_l}})$. Следовательно, $(Q, U') \in P(G, H)$.

Покажем, что при $\rho(G, H) > 0$ множество дизъюнкций Ω выполнимо. Если существует граф $(Q, U'') \in P(G, H)$, то для любого контура $\mu_k \in H$ во множестве U'' найдется дуга $(x_{k_r}^{1-\nu_{k_r}}, x_{k_r}^{\nu_{k_r}})$ такая, что дуга $(x_{k_r}^{\nu_{k_r}}, x_{k_r}^{1-\nu_{k_r}}) \in \mu_k$.

Для каждого $k = 1, \dots, \omega$ положим $x_{k_r} = \nu_{k_r}$. Остальным переменным присвоим произвольные значения из множества $\{0, 1\}$. Тогда по определению $x_{k_r}^{\nu_{k_r}} = 1$ и каждая дизъюнкция $\Omega_k, k = 1, \dots, \omega$, выполнена.

Таким образом, задача о 3-выполнимости полиномиально сводится к задаче проверки неравенства $\rho(G, H) > 0$, и следовательно, последняя является NP -трудной задачей. Отметим, что задача проверки неравенства $\rho(G, H) > 0$ остается сложной и в том случае, когда мощность множества H сравнительно невелика (например, когда значение $|H|$ не превосходит $O(q)$, где $q = |Q|$).

Для доказательства NP -полноты задачи проверки неравенства $\rho(G, H) > 0$ покажем, что она полиномиально сводится к задаче о выполнимости. Пусть $G = (Q, U, V)$ — произвольный смешанный граф без петель и кратных дуг и $H = \{\mu_1, \mu_2, \dots, \mu_h\}$.

Пронумеруем элементы множества V : $V = \bigcup_{l=1}^{|V|} [i, j]_l$. Соответственно

ориентациям ребра $[i, j]_l$ припишем тот же индекс l : $(i, j)_l$ и $(j, i)_l$.

Каждому ребру $[i, j]_l$ сопоставим булеву переменную x_l , которая принимает значение 1, если ребро $[i, j]_l$, $i < j$, заменяется дугой $(i, j)_l$, и значение 0, если это ребро заменяется дугой $(j, i)_l$. Таким образом, каждый граф $(Q, U') \in P(G, \phi)$ однозначно определяет некоторый набор значений переменных $x_l = b_l$, $l = 1, \dots, |V|$, и наоборот.

Если контур μ_k из множества H содержит ребра с номерами $k_1, k_2, \dots, k_\delta$, то он определяет набор значений переменных $x_{k_r} = c_{k_r}$, $r = 1, \dots, \delta$, где

$$c_{k_r} = \begin{cases} 1, & \text{если } (i, j)_{k_r} \in \mu_k, \ i < j, \\ 0, & \text{если } (j, i)_{k_r} \in \mu_k. \end{cases}$$

Очевидно, что граф (Q, U') не содержит контура μ_k тогда и только тогда, когда $b_l \neq c_{k_r}$ хотя бы для одного индекса $l = k_r$. Следовательно, для того чтобы $P(G, H) \neq \phi$, необходимо и достаточно, чтобы было выполнимым множество дизъюнкций $\Omega = \{\Omega_1, \Omega_2, \dots, \Omega_h\}$, где $\Omega_m = \{x_{k_1}^{1-c_{k_1}}, x_{k_2}^{1-c_{k_2}}, \dots, x_{k_\delta}^{1-c_{k_\delta}}\}$, $m = 1, \dots, h$.

Отметим, что в формулировке задачи проверки неравенства $\rho(G, H) > 0$ в виде задачи о выполнимости длина записи входных данных зависит только от $|V|$ и h и не зависит от $|U|$ и q . Для построения графов из множества $P(G, H)$ можно использовать известные алгоритмы решения задачи о выполнимости. В частности, если в каждом контуре μ_k , $k = 1, \dots, h$, содержится не более двух ребер, то алгоритм решения задачи о 2-выполнимости позволяет построить граф $(Q, U') \in P(G, H)$ или установить, что $P(G, H) = \phi$, за $O(h)$ элементарных действий.

2.2. Для построения всех графов множества $P(G, H)$ можно воспользоваться следующей процедурой последовательного ориентирования ребер смешанного графа G .

Пусть μ^1 — один из контуров множества H , содержащий наименьшее число ребер множества V , а $[i, j]^k$ — содержащееся в контуре μ^1 ребро (для определенности контур μ^1 содержит дугу $(i, j)^k$), которое принадлежит наибольшему числу контуров множества H по сравнению с другими ребрами данного контура. Если контур μ^1 не содержит ребер множества V , отличных от ребра $[i, j]^k$, то заменим в G ребро $[i, j]^k$ дугой $(j, i)^k$. Полученный в результате граф обозначим через G^2 . В противном случае наряду с G^2 построим граф G^3 , заменив в G ребро $[i, j]^k$ дугой $(i, j)^k$. Таким образом, получим граф G^2 и, возможно, граф G^3 , отличающиеся от графа G тем, что одно из его ребер заменено дугой. Построим множество контуров $H^2 \subseteq H$ графа $\tilde{G}^2$ и множество контуров $H^3 \subseteq H$ графа $\tilde{G}^3$, удалив из множества H все контуры, содержащие дугу $(i, j)^k$, и все контуры, содержащие дугу $(j, i)^k$, соответственно. (Напомним, что $\tilde{G}^i$ обозначает граф, полученный из $G^i =$

$= (Q, U^i, V^i)$ в результате замены каждого ребра $[i, j]^k \in V^i$ парой дуг $(i, j)^k$ и $(j, i)^k$.)

Пусть построены граф G^1 и множество контуров $H^1 \subseteq H$. Если $H^1 = \emptyset$, то переходим к рассмотрению следующего по порядку графа. Если $H^1 \neq \emptyset$, то среди элементов множества H^1 выделим контур, содержащий наименьшее число ребер множества V (без учета ребер, которые уже заменены дугами). Если таких контуров несколько, то выберем любой из них. Пусть $[a, b]^c$ — ребро выбранного контура, которое принадлежит наибольшему числу контуров множества H^1 по сравнению с другими ребрами, содержащимися в выбранном контуре. Относительно графа G^1 , ребра $[a, b]^c$ и множества H^1 проведем рассуждения, аналогичные тем, что проводились относительно графа G , ребра $[i, j]^k$ и множества H .

Нетрудно заметить, что в каждом построенном графе не содержится контура из множества H , в котором все ребра заменены дугами.

Процесс заканчивается, когда каждому полученному графу $G^m = (Q, U^m, V^m)$ соответствует пустое множество контуров H^m . Смешанный граф G^m определяет множество, состоящее из $2^{|V^m|}$ искомым ориентированных графов, каждый из которых получается из графа G^m в результате замены всех ребер множества V^m дугами, ориентированными произвольным образом. Поскольку на каждом шаге реализуются все возможные альтернативы, то получаем все графы множества $P(G, H)$.

2.3. При вычислении значения $\rho(G, H)$ можно использовать следующее обобщение описанной процедуры ориентирования ребер множества V .

Пусть μ^1 — один из контуров множества H с наименьшим числом ребер. В смешанном графе G заменим дугой каждое ребро $[i, j]^k$, принадлежащее контуру μ^1 . Если μ^1 содержит v_1 ребер, то в результате получим 2^{v_1} различных графов: $G_l = (Q, U_l, V_l)$, $l = 1, 2, \dots, 2^{v_1}$. Пусть контур μ^1 содержится в графе $G_{2^{v_1}}$. Число графов множества $P(G, \phi)$, со-

держащих контур μ^1 , равно $2^{|V_{2^{v_1}}|}$. Для каждого $l = 1, 2, \dots, 2^{v_1} - 1$ отметим граф G_l и определим подмножество H_l контуров множества H , которые содержатся в графе $\tilde{G}_l$.

Таким образом, на первом шаге алгоритма рассмотрен исходный граф G , отмечены графы $G_1, G_2, \dots, G_{2^{v_1}-1}$ и отобран граф $G_{2^{v_1}}$. На втором шаге аналогично из графа G_1 выбирается контур $\mu^2 \in H$, содержащий наименьшее число v_2 ребер графа G_1 , и каждое ребро этого контура заменяется дугой. В результате граф G_1 оказывается рассмотренным, некоторые графы $G_{2^{v_1}+1}, \dots, G_{2^{v_1}+v_2-1}$ — отмеченными, а граф $G_{2^{v_1}+v_2}$, содержащий контур μ^2 , — отобранным. Очевидно, число всех графов множества $P(G, \phi)$, содержащих контур μ^2 , равно $2^{|V_{2^{v_1}+v_2}}|$.

Таким образом, после выполнения m шагов получаем равенство

$$\rho(G, H) = 2^{|V|} - \sum_{i=1}^m 2^{|V_{2^{v_i}}|} - \sum_{l \in \Lambda} [2^{|V_l|} - \rho(G_l, H_l)],$$

где $w_i = \sum_{j=1}^i v_j$ и Λ — множество индексов отмеченных, но не рассмотренных до m -го шага включительно графов.

Поскольку процедуру замены дугами ребер, содержащихся в контуре $\mu^l \in H$, требуется выполнить не более чем для $\prod_{\substack{k=1 \\ v_k \neq 0}}^{l-1} (2^{v_k} - 1)$ рассмотренных графов, то справедлива оценка $O(\sum_{l=1}^h v_l \prod_{\substack{k=1 \\ v_k \neq 0}}^{l-1} (2^{v_k} - 1))$ трудоем-

кости описанного алгоритма вычисления $\rho(G, H)$. При выполнении условия $v_k \leq 1$ для всех $k = 1, \dots, h-1$ получаем оценку $O(h)$.

2.4. При сравнительно небольшой мощности множества контуров $H = \{\mu_1, \mu_2, \dots, \mu_h\}$ более эффективным является следующий алгоритм вычисления значения $\rho(G, H)$.

Обозначим через $\Theta(l)$ число графов множества $P(G, \phi)$, содержащих контур $\mu_l \in H$, и в общем случае через $\Theta(l_1, l_2, \dots, l_m)$ — число графов множества $P(G, \phi)$, каждый из которых содержит контуры $\mu_{l_1}, \mu_{l_2}, \dots, \mu_{l_m}$ из множества H . Контуры μ_a и μ_b будем называть *совместимыми*, если не существует ребра $[i, j]^k \in V$ такого, что дуга $(i, j)^k$ принадлежит одному из этих контуров, а дуга $(j, i)^k$ — второму.

Очевидно, что для того чтобы $\Theta(l_1, l_2, \dots, l_m) \neq 0$, необходимо и достаточно, чтобы контуры $\mu_{l_1}, \mu_{l_2}, \dots, \mu_{l_m}$ были попарно совместимыми.

Составление всех множеств попарно совместимых контуров можно выполнить следующим образом. Пусть составлены все множества попарно совместимых контуров мощности $m-1$: $\{\mu_{l_1^k}, \mu_{l_2^k}, \dots, \mu_{l_{m-1}^k}\}$, $k = 1, \dots, u$. Для определенности можно считать, что $l_1^k < l_2^k < \dots < l_{m-1}^k$, $k = 1, \dots, u$.

Для того чтобы составить все множества попарно совместимых контуров мощности m , достаточно рассмотреть пары совместимых множеств вида $\{\mu_{l_1^k}, \mu_{l_2^k}, \dots, \mu_{l_{m-2}^k}, \mu_{l_{m-1}^k}\}$ и $\{\mu_{l_1^p}, \mu_{l_2^p}, \dots, \mu_{l_{m-2}^p}, \mu_{l_{m-1}^p}\}$, где $l_{m-1}^k < l_{m-1}^p$. Если контуры $\mu_{l_{m-1}^k}$ и $\mu_{l_{m-1}^p}$ совместимы, то и контуры с индексами $l_1^k, l_2^k, \dots, l_{m-2}^k, l_{m-1}^k, l_{m-1}^p$ попарно совместимы. Составление списка пар совместимых контуров осуществляется непосредственной проверкой условия совместимости.

Используя известный в комбинаторном анализе принцип включения и исключения, получаем

$$\rho(G, H) = 2^{|V|} - \sum \Theta(a) + \sum \Theta(a, b) - \sum \Theta(a, b, c) + \dots,$$

где суммирование проводится соответственно по всем контурам μ_a , по всем парам совместимых контуров μ_a и μ_b , по всем тройкам попарно совместимых контуров μ_a, μ_b и μ_c и т.д.

Пусть w — максимальное число попарно совместимых контуров множества H , а $V(l_1, l_2, \dots, l_m)$ — число различных ребер, каждое из

которых содержится хотя бы в одном из попарно совместимых контуров $\mu_{l_1}, \mu_{l_2}, \dots, \mu_{l_m}$, где $l_1 < l_2 < \dots < l_m$. Тогда, очевидно,

$$\Theta(l_1, l_2, \dots, l_m) = 2^{|V| - V(l_1, l_2, \dots, l_m)}$$

и, следовательно,

$$\rho(G, H) = 2^{|V|} + \sum_{m=1}^w (-1)^m \sum_{\{l_1, l_2, \dots, l_m\}} 2^{|V| - V(l_1, l_2, \dots, l_m)}.$$

Легко показать, что оценка трудоемкости описанного алгоритма вычисления значения $\rho(G, H)$ равна $O(v \cdot 2^{w+1})$, где v — максимальное число ребер, содержащихся в одном контуре из множества H . Сравнивая ее с оценкой, приведенной для алгоритма из п.2.3, можно предложить следующую комбинацию алгоритмов. Вначале, используя алгоритм из п.2.3, подсчитать графы из множества $P(G, \phi)$ без контуров из множества H со сравнительно небольшим числом ребер, а затем, используя алгоритм из данного пункта, вычислить $\rho(G_l, H_l)$ при сравнительно небольших значениях $|H_l|$.

2.5. Очевидно, $P(G, \tilde{H})$ представляет собой множество $P(G)$ всех бесконтурных графов, порождаемых смешанным графом G в результате замены каждого ребра множества V дугой. В этом случае наряду с $\rho(G, \tilde{H})$ будем использовать обозначение $\rho(G)$.

Отметим, что в любом графе $(Q, U') \in P(G)$ все кратные ребра $[i, j]^k \in V$ графа G имеют одну и ту же ориентацию: либо $(i, j)^k \in U'$, либо $(j, i)^k \in U'$ для всех k . Таким образом, при изучении множества $P(G)$ ограничимся рассмотрением смешанных графов G , которые не содержат кратных ребер и удовлетворяют следующему условию: если $(i, j) \in U$, то $(j, i) \notin U$ и $[i, j] \notin V$.

Пусть $G' = (Q, U')$ и $G'' = (Q, U'')$ — различные графы из множества $P(G)$. Обозначим через $\epsilon(G', G'')$ число ребер $[i, j] \in V$ таких, что $(i, j) \in U'$, а $(j, i) \in U''$. Пусть $E(G', G'')$ — наименьшая из абсолютных величин разностей рангов вершин i и j , вычисленных для всех указанных пар вершин $i, j \in Q$ относительно графа G' .

Покажем, что если $G' \in P(G)$, $G'' \in P(G)$ и $\epsilon(G', G'') = m$, то существует такая последовательность графов $G' = G_0, G_1, \dots, G_m = G''$ из множества $P(G)$, что $\epsilon(G_{l-1}, G_l) = 1$ и $E(G_{l-1}, G_l) = E(G_{l-1}, G'')$, $l = 1, \dots, m$.

Пусть граф G' содержит дугу (a, b) , граф G'' — дугу (b, a) , граф G_1 получен из графа G' в результате замены дуги (a, b) дугой (b, a) , причем $E(G', G_1) = E(G', G'')$. Для доказательства утверждения достаточно показать, что $G_1 \in P(G)$. Допустим от противного, что существует контур в графе G_1 , а следовательно, не содержащий дуги (a, b) путь из вершины a в вершину b в графе G' . Поскольку абсолютные величины разности рангов вершин, инцидентных каждой дуге этого пути, меньше $E(G', G'')$, то в графе G'' существует тот же путь из вершины a в вершину b , и следовательно, G'' не принадлежит $P(G)$. Получаем противоречие.

Если $G', G'' \in P(G)$ и $U = \phi$, то существует последовательность графов $G' = G_0, G_1, \dots, G_r = G''$ из множества $P(G)$, такая, что $\epsilon(G_{l-1}, G_l) = E(G_{l-1}, G_l) = 1$, $l = 1, \dots, r$.

При доказательстве этого утверждения достаточно рассмотреть случай, когда $\epsilon(G', G'') = 1$. Пусть графы G' и G'' отличаются ориентацией ребра $[a, b]$, причем граф G' содержит дугу (a, b) , а граф G'' — дугу (b, a) . Возможны два случая: либо граф G' содержит вершину $c \neq b$; полустепень исхода которой равна нулю, либо он не содержит такой вершины.

В первом случае предположим, что утверждение имеет место при числе вершин, равном $q \geq 3$, и докажем его справедливость при числе вершин, равном $q + 1$. При $q = 2$ утверждение очевидно.

Удалим из графов G' и G'' вершину c и все инцидентные ей дуги. Полученные подграфы обозначим соответственно через $\bar{G}'$ и $\bar{G}''$. По предположению существует последовательность бесконтурных графов $\bar{G}' = \bar{G}_0, \bar{G}_1, \dots, \bar{G}_r = \bar{G}''$ таких, что $\epsilon(\bar{G}_{l-1}, \bar{G}_l) = E(\bar{G}_{l-1}, \bar{G}_l) = 1, l = 1, \dots, r$. Присоединив вершину c вместе с инцидентными ей дугами к каждому графу $G_l, l = 0, \dots, r$, получим искомую последовательность.

Во втором случае единственной вершиной в графе G' , полустепень исхода которой равна нулю, является вершина b . Выберем в G' вершину d , для которой $(d, b) \in U'$ и $R(b) - R(d) = 1$, где $R(k)$ — ранг вершины k в графе G' . Будем рассматривать нетривиальный случай, когда $R(b) - R(a) > 1$ и, следовательно, $d \neq a$. Обозначим через $\hat{G}'$ и $\hat{G}''$ графы, отличающиеся соответственно от G' и G'' только ориентацией ребра $[b, d]$. Нетрудно видеть, что графы $\hat{G}'$ и $\hat{G}''$ не содержат контуров, отличаются ориентацией ребра $[a, b]$ и содержат вершину $d \neq b$, полустепень исхода которой равна нулю. Следовательно, к этим графам применима доказанная часть утверждения. Аналогичное заключение может быть сделано относительно бесконтурных графов $\hat{G}''$ и G'' . Действительно, они отличаются ориентацией ребра $[b, d]$, и поскольку в них не существует пути, соединяющего вершины a и d , то эти графы содержат вершину $e \neq d$, полустепень исхода которой равна нулю. Утверждение доказано.

2.6. Исследуем влияние на величину $\rho(G)$ некоторых преобразований смешанного графа G .

Пусть $[i, j]$ — некоторое ребро, возможно, $[i, j] \notin V$. Через $_{ij}G$ обозначим граф, полученный из G в результате отождествления вершин i и j с последующим отождествлением всех кратных ребер, всех кратных дуг, удалением всех петель и всех ребер $[a, b]$, для которых в полученном графе есть дуга, инцидентная вершинам a и b . Нетрудно обобщить эту процедуру на множество W ребер. Через $_wG$ обозначим граф, полученный из G в результате последовательного применения описанной процедуры отождествления пар вершин i и j для каждого ребра $[i, j] \in W$. Если $W = \emptyset$, то будем полагать $_wG = G$.

Вершины i и j графа G будем называть *несмежными*, если они не инцидентны одному и тому же ребру или дуге. Несмежные вершины i и j будем называть *сравнимыми*, если в графе (Q, U) существует путь, содержащий эти вершины.

Покажем, что если i и j — несмежные вершины графа G , то

$$\rho(G) = \rho(Q, U, V \cup [i, j]) - \rho(_{ij}G). \quad (2.1)$$

Доказательство этого утверждения проведем по индукции относительно числа ребер графа G .

Пусть $V = \emptyset$. Если в графе (Q, U) имеется контур, то $\rho(G) = \rho(Q, U, [i, j]) = \rho(ijG) = 0$. Пусть граф (Q, U) не содержит контуров. Если вершины i и j сравнимы в G , то $\rho(G) = \rho(Q, U, [i, j]) = 1$ и $\rho(ijG) = 0$, в противном случае $\rho(G) = \rho(ijG) = 1$ и $\rho(Q, U, [i, j]) = 2$.

Итак, для $V = \emptyset$ равенство (2.1) выполняется. Предположим, что утверждение имеет место при $|V| = m$, и докажем его при $|V| = m + 1$.

Выберем произвольное ребро $[a, b] \in V$. Очевидно, что

$$\rho(Q, U, V \cup [i, j]) = \rho(Q, U \cup (a, b), V \cup [i, j] \setminus [a, b]) + \\ + \rho(Q, U \cup (b, a), V \cup [i, j] \setminus [a, b]).$$

По предположению

$$\rho(Q, U \cup (a, b), V \cup [i, j] \setminus [a, b]) = \rho(Q, U \cup (a, b), V \setminus [a, b]) + \\ + \rho(ij(Q, U \cup (a, b), V \setminus [a, b]))$$

и

$$\rho(Q, U \cup (b, a), V \cup [i, j] \setminus [a, b]) = \rho(Q, U \cup (b, a), V \setminus [a, b]) + \\ + \rho(ij(Q, U \cup (b, a), V \setminus [a, b])).$$

Наконец, $\rho(G) = \rho(Q, U \cup (a, b), V \setminus [a, b]) + \rho(Q, U \cup (b, a), V \setminus [a, b])$ и $\rho(ijG) = \rho(ij(Q, U \cup (a, b), V \setminus [a, b])) + \rho(ij(Q, U \cup (b, a), V \setminus [a, b]))$.

Утверждение доказано.

Пусть элементы множества $V' \subseteq V$ пронумерованы: $V' = \bigcup_{l=1}^{|V'|} [i, j]_l$.

Применяя равенство (2.1) конечное число раз, можно получить следующее соотношение:

$$\rho(G) = \rho(Q, U, V \setminus V') + \sum_{[i, j]_k \in V'} \rho(ij(Q, U, V \setminus \bigcup_{l=1}^{k-1} [a, b]_l)).$$

Это равенство позволяет свести вычисление значений $\rho(G)$ при $|Q| = k$ к вычислению значения $\rho(Q, U)$ и значений $\rho(G)$ при $|Q| = k - 1$.

Аналогично, если $V' \subseteq V$ и граф (Q, U') ациклический, то $\rho(G) = \sum \rho(w(Q, U, V \setminus V'))$, где суммирование проводится по всем $W \subseteq V$, для которых в графе (Q, W) нет цепи, соединяющей смежные вершины графа $(Q, U, V \setminus V')$.

2.7. В заключение отметим, что если i и j — несмежные вершины графа G , то

$$2\rho(G) \geq \rho(Q, U, V \cup [i, j]) \geq \rho(G) \geq \rho(Q, U \cup (i, j), V) \geq 0.$$

Сформулируем условия, при которых в этом соотношении имеет место знак равенства.

Если i и j — несмежные вершины графа G и $\rho(G) \neq 0$, то следующие утверждения эквивалентны:

а) $\rho(Q, U, V \cup [i, j]) = \rho(G)$;

б) $\rho(ijG) = 0$;

в) i и j — сравнимые вершины графа G .

Действительно, условия а) и в) эквивалентны согласно равенству (2.1). Поскольку граф (Q, U) не содержит контуров, для б) необходимо и достаточно, чтобы выполнялось условие в).

Нетрудно убедиться в справедливости и следующих утверждений.

Если i и j — несмежные и несравнимые вершины графа G , то $\rho(Q, U, V \cup [i, j]) = 2\rho(G)$ и $\rho(G) = \rho(ijG)$.

Если i и j — несмежные вершины графа G и в графе $\tilde{G}$ не существует пути из j в i , то $\rho(Q, U \cup (i, j), V) = \rho(G)$ и $\rho(Q, U \cup (j, i), V) = \rho(ijG)$.

Если i и j — несмежные вершины графа G и $\rho(G) \neq 0$, то следующие утверждения эквивалентны:

а) $\rho(Q, U \cup (i, j), V) = 0$;

б) $\rho(Q, U \cup (j, i), V) = \rho(G) + \rho(ijG)$;

в) в графе (Q, U) существует путь из вершины j в вершину i .

Приведенные утверждения могут быть использованы при разработке методов вычисления значения $\rho(G)$.

§ 3. Последовательный анализ вариантов

В этом параграфе рассматриваются основанные на идеях последовательного анализа вариантов точные и приближенные алгоритмы построения допустимого относительно взвешенного смешанного графа $G = (Q, U, V)$ расписания, которое является оптимальным (т.е. заданная неубывающая по каждому из аргументов действительная функция $F(x_1, x_2, \dots, x_q)$ принимает минимальное значение при x_i , равном моменту t_i завершения выполнения операции i , $i = 1, \dots, q$, при этом расписании).

Как показано в § 1 данной главы, оптимальное расписание можно искать в классе активных расписаний. Для этого достаточно из множества $\hat{P}(G)$ ориентированных графов, порождаемых смешанным графом G , выделить множество $\hat{P}(G)$ графов, не содержащих контуров положительного веса. Каждый граф $(Q, U') \in \hat{P}(G)$ однозначно определяет активное расписание, допустимое относительно графа G (и наоборот). Для построения этого расписания достаточно определить наиболее ранний срок $\bar{t}_i = \bar{t}_i(Q, U')$ завершения выполнения каждой операции $i \in Q$ относительно графа (Q, U') .

3.1. Для организации целенаправленного перебора графов множества $\hat{P}(G)$ будем использовать процедуру последовательного разбиения множества $\hat{P}(G)$ на подмножества. При этом множество $\hat{P}(G)$ сначала разбивается на подмножества $\hat{P}(G_1), \hat{P}(G_2), \dots, \hat{P}(G_h)$, где G_k , $k = 1, \dots, h$, — графы, получаемые из графа G в результате замены одного или нескольких его ребер дугами. Затем одно из множеств $\hat{P}(G_k)$, скажем $\hat{P}(G_1)$, в свою очередь разбивается на подмножества $\hat{P}(G_{h+1}), \hat{P}(G_{h+2}), \dots, \hat{P}(G_{h+p})$. В результате получаем разбиение исходного множества $\hat{P}(G)$ на подмножества $\hat{P}(G_1), \dots, \hat{P}(G_{l-1}), \hat{P}(G_{l+1}), \dots, \hat{P}(G_{h+p})$ и т.д.

Этот процесс удобно представлять "растущим" *выходящим деревом* (Z_m, W_m) , где Z_m — множество вершин, а W_m — множество дуг. Каждой вершине дерева сопоставлен некоторый граф $G_k = (Q, U_k, V_k)$, где $U \subseteq U_k$ и $V_k \subseteq V$. В дальнейшем вершину дерева будем отождествлять с сопоставленным ей графом. Корнем дерева является граф $G_0 = G$.

Остальные вершины (графы) нумеруются последовательно по мере их получения. Множество $Z_m = \{G_0, G_1, \dots, G_m\}$. Корень G_0 дерева соединяется исходящими дугами с вершинами $G_1, G_2, \dots, G_h$, вершина G_l — с вершинами $G_{h+1}, G_{h+2}, \dots, G_{h+p}$ и т.д.

Обозначим множество висячих вершин дерева (Z_m, W_m) через $\tilde{Z}_m$. Это множество определяет текущее разбиение множества $\hat{P}(G)$ на подмножества.

Введем в рассмотрение функцию

$$f(G_k) = \begin{cases} \min \{F(\bar{t}_1(Q, U'), \dots, \bar{t}_q(Q, U')) \mid (Q, U') \in \hat{P}(G_k)\}, & \text{если } \hat{P}(G_k) \neq \phi, \\ \infty, & \text{если } \hat{P}(G_k) = \phi. \end{cases}$$

Очевидно, если $V_k = \phi$ и граф (Q, U_k) не содержит контуров положительного веса, то $f(G_k) = F(\bar{t}_1(Q, U_k), \dots, \bar{t}_q(Q, U_k))$.

Через $f_0(G_k)$ и $f^0(G_k)$ будем обозначать соответственно оценки снизу и сверху значения $f(G_k)$. Число $r_m = \min \{f^0(G_k) \mid G_k \in \tilde{Z}_m\}$ будем называть (текущим) рекордом, а граф $(Q, U^m) \in \hat{P}(G_k)$, для которого $r_m = f(Q, U^m)$ и $G_k \in \tilde{Z}_m$, — рекордным.

Отношение строгого порядка $\rightarrow$, заданное на множестве Z_m , называется отношением доминирования, если из соотношения $G_k \rightarrow G_l$ следует, что $f(G_k) \leq f(G_l)$ и в дереве (Z_m, W_m) нет пути из G_k в G_l .

Вершину $G_l \in \tilde{Z}_m$ будем называть конечной, если выполняется хотя бы одно из условий:

а) $f_0(G_l) = f^0(G_l)$;

б) $f_0(G_l) \geq r_m$;

в) существует граф $G_k \in Z_m$ такой, что $G_k \rightarrow G_l$.

Множество всех конечных вершин дерева (Z_m, W_m) будем обозначать через $\bar{Z}_m$.

При выполнении условия а) получаем точное значение $f(G_l)$. Из условия б) следует соотношение $f(Q, U^m) \leq f(Q, U')$ для любого графа $(Q, U') \in \hat{P}(G_l)$. При выполнении условия в) существует граф $(Q, U'') \in \hat{P}(G_k)$ такой, что $f(Q, U'') \leq f(G_l)$. Следовательно, если выполняется условие б) или в), то при поиске оптимального расписания граф G_l можно исключить из рассмотрения.

3.2. Опишем одну из типичных схем поиска оптимального расписания — схему ветвей и границ.

1) Полагаем $m = 0$, $G_0 = G$ и строим рекордный граф $(Q, U^m) \in \hat{P}(G_0)$. Если $\hat{P}(G_0) = \phi$, то выполняем шаг 10. В противном случае полагаем $Z_m = \bar{Z}_m = \{G_0\}$, $W_m = \phi$, $\bar{Z}_m = \phi$, $r_m = f^0(G_0) = f(Q, U^m)$ и выполняем шаг 2).

2) Из множества $\tilde{Z}_m \setminus \bar{Z}_m$ выбираем граф $G_l = (Q, U_l, V_l)$ и выполняем шаг 3). Если $\tilde{Z}_m \setminus \bar{Z}_m = \phi$, то выполняем шаг 10).

3) Вычисляем оценку $f_0(G_l)$ значения функции $f(G_l)$. Если $f_0(G_l) = f^0(G_l)$, то выполняем шаг 8), в противном случае — шаг 4).

4) Если справедливо соотношение

$$r_m \leq f_0(G_l), \quad (3.1)$$

то выполняем шаг 9), в противном случае — шаг 5).

5) Если существует граф $G_k \in Z_m$, для которого выполняется соотношение $G_k \rightarrow G_l$, то включаем G_l во множество $\bar{Z}_m$ и выполняем шаг 2). В противном случае выполняем шаг 6).

6) Строим разбиение множества $\hat{P}(G_l)$ на $p \geq 2$ подмножеств $\hat{P}(G_{m+1}), \dots, \hat{P}(G_{m+p})$. Полагаем $Z_{m+p} = Z_m \cup \{G_{m+1}, \dots, G_{m+p}\}$, $W_{m+p} = W_m \cup \{(G_l, G_{m+1}), \dots, (G_l, G_{m+p})\}$, $\tilde{Z}_{m+p} = \tilde{Z}_m \cup \{G_{m+1}, \dots, G_{m+p}\} \setminus \{G_l\}$. Выполняем шаг 7).

7) Вычисляем оценки $f^0(G_{m+1}), \dots, f^0(G_{m+p})$ и выбираем среди них наименьшую $f^0(G_h)$. Выполняем шаг 8), полагая l и m равными h и $m+p$ соответственно.

8) Если $r_m > f^0(G_l)$, то полагаем рекорд $r_m = f^0(G_l)$ и строим новый рекордный граф $(Q, U^m) \in \hat{P}(G_l)$, для которого $f^0(G_l) = f(Q, U^m)$. В любом случае выполняем шаг 9).

9) Во множество конечных вершин $\bar{Z}_m$ включаем все вершины G_k из множества $\tilde{Z}_m$, для которых $f_0(G_k) \geq r_m$, и выполняем шаг 2).

10) Если $m = 0$, то не существует расписания, допустимого относительно графа G . Если $m > 0$, то оптимальным является расписание, при котором $\bar{t}_i = \bar{t}_i(Q, U^m)$, $i = 1, \dots, q$. Оптимальное значение целевой функции равно рекорду r_m .

Полученный на последнем шаге граф (Z_m, W_m) обозначим через (Z, W) , а множество $\tilde{Z}_m$ — через $\tilde{Z}$.

При практической реализации этой схемы описанные шаги и последовательность их выполнения могут меняться в зависимости от особенностей решаемой задачи.

3.3. Перейдем к более детальному описанию основных фрагментов предлагаемой схемы.

Для проверки условия $\hat{P}(G) \neq \emptyset$ и построения рекордного графа $(Q, U^m) \in \hat{P}(G)$ можно использовать алгоритмы, описанные в § 1 и § 2 этой главы. Напомним, что если граф (Q, U) содержит контур положительного веса, то $\hat{P}(G) = \emptyset$. Если же граф (Q, U) содержит контуры и все они неположительного веса, то задача проверки условия $\hat{P}(G) \neq \emptyset$ является *NP-трудной* в сильном смысле.

Опишем алгоритм построения активного расписания, допустимого относительно заданного графа $G = (Q, U, V)$.

а) Вычисляем наиболее ранний срок $t_i^0(Q, U)$ начала выполнения каждой операции $i \in Q$. Если при этом в графе (Q, U) будет обнаружен контур положительного веса, то $\hat{P}(G) = \emptyset$. В противном случае выполняем п. б).

б) Во множестве V выбираем *конфликтное ребро*, т.е. такое ребро $[i, j]^k \in V$, для которого выполняется соотношение

$$-a_{ji}^k < t_j^0(Q, U) - t_i^0(Q, U) < a_{ij}^k, \quad (3.2)$$

и выполняем п.в). Если в графе (Q, U, V) нет конфликтных ребер, то искомые времена начала выполнения операций $t_l^0 = t_l^0(Q, U), l = 1, \dots, q$.

в) Вычисляем наиболее ранние сроки $t_l^0(Q, U \cup (i, j)^k), l = 1, \dots, q$, и выполняем п. б), заменяя граф (Q, U, V) графом $(Q, U \cup (i, j)^k, V \setminus [i, j]^k)$. Если в графе $(Q, U \cup (i, j)^k)$ содержится контур положительного веса, то вычисляем наиболее ранние сроки $t_l^0(Q, U \cup (j, i)^k), l = 1, \dots, q$, и выполняем п. б), заменяя граф (Q, U, V) графом $(Q, U \cup (j, i)^k, V \setminus [i, j]^k)$. Если в графе $(Q, U \cup (j, i)^k)$ содержится контур положительного веса, то допустимого относительно графа G расписания не существует.

Через $\hat{V}$ обозначим множество всех конфликтных ребер графа $G = (Q, U, V)$. Очевидно, что при $\hat{V} = \phi$ для каждого ребра $[i, j]^k \in V$ выполняется одно из соотношений $t_j^0(Q, U) - t_i^0(Q, U) \geq a_{ij}^k$ или $t_i^0(Q, U) - t_j^0(Q, U) \geq a_{ji}^k$. Для каждой дуги $(i, j) \in U$ имеет место соотношение $t_j^0(Q, U) - t_i^0(Q, U) \geq a_{ij}$. Следовательно, значения $t_l^0 = t_l^0(Q, U), l = 1, \dots, q$, определяют допустимое относительно графа G расписание и $f(G) = f(Q, U, \phi)$.

Если $\hat{V} \neq \phi$, то для улучшения качества получаемого расписания целесообразно при выборе очередного ребра $[i, j]^k \in \hat{V}$ использовать те или иные *правила предпочтения (приоритеты)* в зависимости от вида целевой функции.

3.4. Если граф (Q, U) не содержит контуров, то $\hat{P}(G) \neq \phi$ и для построения активного расписания, допустимого относительно графа G , можно воспользоваться следующим *алгоритмом*, трудоемкость которого не превосходит $O(q^2 + |V|)$ элементарных действий.

а) Полагаем $p = 0, \bar{Q}^0 = \phi, (Q, U^0, V^0) = (Q, U, V)$ и вычисляем наиболее ранние сроки $t_i^0(Q, U^0)$ начала всех операций $i \in Q$. Очевидно, искомые времена начала выполнения операций $t_i^0 \geq t_i^0(Q, U^0), i = 1, \dots, q$.

б) В графе (Q, U^p) выбираем одну из вершин j , для которой не существует дуги $(i, j) \in U^p$ при $i \in Q \setminus \bar{Q}^p$. Определяем момент начала выполнения операции j в искомом расписании, полагая $t_j^0 = t_j^0(Q, U^p)$. Вершину j включаем во множество упорядоченных вершин: $\bar{Q}^{p+1} = \bar{Q}^p \cup \{j\}$.

в) Заменяем в графе (Q, U^p, V^p) все ребра, инцидентные вершине j , на исходящие из нее дуги. Полученный граф обозначаем через (Q, U^{p+1}, V^{p+1}) .

г) Используя вычисленные значения $t_i^0, i \in \bar{Q}^{p+1}$, и $t_i^0(Q, U^p), i \in Q \setminus \bar{Q}^{p+1}$, определяем наиболее ранние сроки $t_i^0(Q, U^{p+1}), i \in Q \setminus \bar{Q}^{p+1}$.

д) Если $V^{p+1} = \phi$, то полагаем $t_i^0 = t_i^0(Q, U^{p+1})$ для всех $i \in Q \setminus \bar{Q}^{p+1}$. В противном случае переходим к п. б), заменяя граф (Q, U^p, V^p) графом (Q, U^{p+1}, V^{p+1}) , а множество $\bar{Q}^p$ — множеством $\bar{Q}^{p+1}$.

Применяя эту процедуру к исходному графу G не более чем q раз, получаем значения $t_i^0, i \in Q$, определяющие некоторое активное расписание.

При выборе очередной вершины j в п. б) обычно используют соответствующие *правила предпочтения*.

3.5. Если на шаге 2) приведенной выше схемы ветвей и границ $|\tilde{Z}_m \setminus \bar{Z}_m| \geq 2$, то необходимо выбрать очередной граф G_l из множества $\tilde{Z}_m \setminus \bar{Z}_m$ возможных кандидатов.

Пусть на множестве Z задана некоторая действительная функция $\psi(G_k)$, $G_k \in Z$. Будем говорить, что используется *стратегия, основанная на функции* $\psi(G_k)$, если выбирается граф $G_l \in \tilde{Z}_m \setminus \bar{Z}_m$, для которого $\psi(G_l) = \min \{ \psi(G_k) \mid G_k \in \tilde{Z}_m \setminus \bar{Z}_m \}$.

Стратегию, основанную на функции $\psi(G_k)$, называют *поиском в ширину*, если значение функции $\psi(G_k)$ равно рангу $R(G_k)$ вершины G_k в графе (Z_m, W_m) ;

поиском в глубину, если $\psi(G_k) = -R(G_k)$;

поиском по наилучшей оценке снизу, если $\psi(G_k) = f_0(G_k)$;

поиском по наилучшей оценке сверху, если $\psi(G_k) = f^0(G_k)$.

Эффективность той или иной стратегии определяется тем, насколько полно функция $\psi(G_k)$ отражает специфику решаемой задачи.

3.6. При выборе *алгоритма вычисления оценок* $f_0(G_l)$ на шаге 3) необходимо учитывать два противоречивых обстоятельства. С одной стороны, уточнение оценок позволяет обычно сократить процесс ветвления. С другой стороны, уточнение оценок сопряжено с увеличением объема вычислений, что может компенсировать выигрыш от сокращения процесса ветвления.

Пусть $(G_k, G_l) \in W_m$. Поскольку $\hat{P}(G_l) \subseteq \hat{P}(G_k)$, то $f(G_k) \leq f(G_l)$ и значение $f_0(G_l)$ можно положить равным $f_0(G_k)$. Поэтому в дальнейшем будем предполагать, что для любой дуги $(G_k, G_l) \in W_m$ имеет место соотношение

$$f_0(G_k) \leq f_0(G_l). \quad (3.3)$$

При получении оценок снизу обычно используется прием *релаксации*, т.е. замены исходной задачи более простой задачей путем исключения из рассмотрения некоторых ограничений.

Поскольку $F(x_1, x_2, \dots, x_q)$ не убывает по каждому из аргументов, то *релаксированную задачу* можно получить, например, в результате удаления из графа $G_l = (Q, U_l, V_l)$ всех или части ребер.

Если удалены все ребра графа G_l , то получаем релаксированную задачу, состоящую в определении ранних сроков $t_i^0(Q, U_l)$ начала выполнения операций $i \in Q$ относительно графа (Q, U_l) , и полагаем $f_0(G_l) = F(t_1^0(Q, U_l), \dots, t_q^0(Q, U_l))$.

Пусть некоторый порожденный подграф (Q', U', V') графа G_l является полным (в качестве $Q' \subseteq Q$ обычно выбирается множество операций, выполняемых одним и тем же прибором). Удаляя из графа G_l все ребра за исключением ребер множества V' , получаем релаксированную задачу построения допустимого относительно графа (Q, U_l, V') расписания, при котором минимально значение заданной целевой функции.

Например, если все $a_{ij} = t_i$ и требуется минимизировать общее время выполнения операций, т.е. $F(x_1, x_2, \dots, x_q) = \max \{x_i \mid i \in Q\}$, то указанная релаксированная задача по существу представляет собой задачу минимизации максимального временного смещения при обслуживании

одним прибором частично упорядоченного множества требований Q' , для каждого из которых задан момент $d_i = t_i^0(Q, U_i)$ поступления в систему, длительность обслуживания t_i и директивный срок $D_i = v(Q, U_i) + t_i - v_i(Q, U_i)$, где $v_i(Q, U_i)$ — максимальный вес пути в графе (Q, U_i) с началом в вершине i и $v(Q, U_i) = \max \{v_i(Q, U_i) \mid i \in Q\}$.

3.7. Рассмотрим некоторые *способы разбиения* множества $\hat{P}(G_I)$ на подмножества на шаге 6) описанной в п. 3.2. схемы при различных предположениях относительно графа G .

Пусть $\hat{V}_I$ — множество конфликтных ребер графа $G_I = (Q, U_I, V_I)$. Как уже отмечалось, если $\hat{V} = \emptyset$, то $f(G_I) = f(Q, U_I, \emptyset)$ и вершина G_I является конечной.

Если $\hat{V}_I \neq \emptyset$, то разбиение множества $\hat{P}(G_I)$ определяется выбором некоторого конкретного подмножества $\bar{V}_I \subseteq \hat{V}_I$. В результате замены каждого ребра $[i, j]^k \in \bar{V}_I$ дугой $(i, j)^k$ или дугой $(j, i)^k$ получаем $p = 2^{|\bar{V}_I|}$ различных графов $G_{m+1}, \dots, G_{m+p}$, причем $\hat{P}(G_I) = \hat{P}(G_{m+1}) \cup \dots \cup \hat{P}(G_{m+p})$. Как правило, в качестве $\bar{V}_I$ выбирают множество сравнительно небольшой мощности (обычно $|\bar{V}_I| = 1$).

Рассмотрим несколько *правил предпочтения* при выборе множества $\bar{V}_I$ на примере задачи минимизации общего времени выполнения операций, сформулированной в п. 3.6. Для каждого ребра $[i, j] \in \bar{V}_I$ вычислим величину $h_{ij} = t_{ij}^0(Q, U_I) + a_{ij} + v_j(Q, U_I) - v(Q, U_I)$. Поскольку ребро $[i, j]$ конфликтное и вес $a_{bc} = t_b$ для каждой дуги $(b, c) \in U_I$, то в графе (Q, U_I) не существует пути из вершины j в вершину i . Следовательно, в графе $(Q, U_I \cup (i, j))$ найдется путь веса $v(Q, U_I) + h_{ij}$.

Таким образом, при выборе ребер $[i, j] \in \bar{V}_I$ для включения во множество V_I можно использовать одну из следующих величин: $h_{ij} + h_{ji}$, $\min \{h_{ij}, h_{ji}\}$ и $\max \{h_{ij}, h_{ji}\}$, отдавая предпочтение тем ребрам, для которых соответствующая величина больше. Использование приоритетов при выборе множества $\bar{V}_I$ позволяет на ранних итерациях метода рассмотреть „наиболее существенные“ из конфликтных ребер, после чего остальные ребра уже не будут конфликтными.

3.8. Если в графе $G = (Q, U, V)$ все веса $a_{ij} > 0$, то каждый граф из $\hat{P}(G)$ не содержит контуров. В этом случае обычно используют *процедуру разбиения* множества $\hat{P}(G)$ на подмножества, основанную на последовательном выборе вершин графа G .

Пусть $\pi = (i_1, i_2, \dots, i_c)$ — некоторая перестановка элементов (не обязательно всех) из множества Q . Обозначим через $G(\pi)$ граф, полученный из G следующим образом. Выбираем в G вершину i_1 и заменяем все инцидентные ей ребра исходящими из нее дугами. В полученном графе выбираем вершину i_2 и заменяем все инцидентные ей ребра исходящими из нее дугами и т.д., пока не будет рассмотрена вершина i_c . Очевидно, если для каждого $h = 1, \dots, c$ вершина i_h имеет нулевую полустепень захода в подграфе графа $G(i_1, i_2, \dots, i_{h-1})$, порожденном множеством вершин $Q \setminus \{i_1, i_2, \dots, i_{h-1}\}$, то граф $G(\pi)$ не содержит контуров (предполагается, что $G(\emptyset) = G$).

Соответствующую процедуру можно описать следующим образом. Полагаем $\bar{Q}_0 = \phi$, $G_0 = (Q, U_0, V_0) = G$. Пусть $\hat{Q}_0 = \{i'_1, i'_2, \dots, i'_{q_0}\}$ — множество всех вершин с нулевой полустепенью захода в графе (Q, U_0) . Для каждой вершины $i'_k \in \hat{Q}_0$ полагаем $\bar{Q}_k = \{i'_k\}$, $\pi_k = (i'_k)$ и строим граф $G_k = G(\pi_k)$. Получаем разбиение множества $\hat{P}(G_0)$ на подмножества $\hat{P}(G_1), \dots, \hat{P}(G_{q_0})$. Очевидно, что при любом активном расписании, допустимом относительно графа G_k , имеет место равенство $t_{i'_k}^0 = 0$.

Пусть $G_l = (Q, U_l, V_l) = G(\pi_l)$ — граф, который рассматривается на очередной итерации, π_l — некоторая перестановка элементов множества $\bar{Q}_l \subset Q$ и $\hat{Q}_l = \{j_1, j_2, \dots, j_{q_l}\}$ — множество всех вершин с нулевой полустепенью захода в подграфе графа (Q, U_l) , порожденном множеством вершин $Q \setminus \bar{Q}_l$. Для каждой вершины $j_k \in \hat{Q}_l$ строим граф $G_{m+k} = G(\pi_{m+k})$ и полагаем $\bar{Q}_{m+k} = \bar{Q}_l \cup \{j_k\}$, где $\pi_{m+k} = (\pi_l, j_k)$ и m — число ранее построенных графов. Очевидно, $\hat{P}(G_l) = \hat{P}(G_{m+1}) \cup \dots \cup \hat{P}(G_{m+q_l})$ и $t_{j_k}^0 = t_{j_k}^0(Q, U_l)$ при любом активном расписании, допустимом относительно графа G_{m+k} .

Если $\bar{Q}_l = Q$, то $V_l = \phi$ и вершина G_l является конечной. При таком способе разбиения множества $\hat{P}(G)$ максимальный ранг вершин дерева (Z, W) , очевидно, не превосходит q .

3.9. Описанная в предыдущем пункте процедура разбиения множества $\hat{P}(G)$ на подмножества может быть дополнена *отсеиванием* некоторых из этих подмножеств. Ограничимся рассмотрением случая, когда в графе $G = (Q, U, V)$ веса $a_{ij} = t_i$.

Расписание называется *компактным*, если любую из операций нельзя начать раньше, чем при этом расписании, при условии, что остальные операции начнутся не позже. Ясно, что *любое компактное расписание является активным* (но не наоборот) и *в классе компактных расписаний содержится хотя бы одно оптимальное*.

Пусть на очередной итерации рассматривается граф $G_l = (Q, U_l, V_l) = G(\pi_l)$. Из множества $\hat{Q}_l$ выберем вершину j_a , для которой $\bar{t}_{j_a}(Q, U_l) = \min \{\bar{t}_{j_k}(Q, U_l) \mid j_k \in \hat{Q}_l\}$. Обозначим $\tilde{Q}_l = \{j_a\} \cup \{j_k \mid j_k \in \hat{Q}_l, [j_k, j_a] \in V_l\}$ и для каждой вершины $j_k \in \tilde{Q}_l$ построим граф $G(\pi_l, j_k)$. Полученные графы обозначим через $G_{m+1}, G_{m+2}, \dots, G_{m+p}$, где $p = |\tilde{Q}_l|$ и m — число ранее построенных графов.

Если $\tilde{Q}_l$ — собственное подмножество $\hat{Q}_l$, то, возможно, $\bigcup_{k=1}^p \hat{P}(G_{m+k}) \neq \hat{P}(G_l)$. Покажем, что *активное расписание s , допустимое относительно графа $(Q, U') \in \hat{P}(G_l) \setminus \bigcup_{k=1}^p \hat{P}(G_{m+k})$, не является компактным*.

Пусть $(Q, U') = G(\pi_l, i_1, i_2, \dots, i_\omega, j_a, \pi)$, где $i_\nu \in Q \setminus \bar{Q}_l$, $\nu = 1, \dots, \omega$, и через π обозначена некоторая перестановка элементов множества

$Q \setminus \bar{Q}_1 \setminus \{i_1, i_2, \dots, i_\omega, j_a\}$. Если не существует ни одного ребра $[i_\nu, j_a] \in \in V_l$ при некотором $\nu \in \{1, 2, \dots, \omega\}$, то граф (Q, U') есть $G(\pi_l, j_a, i_1, i_2, \dots, i_\omega, \pi)$ и принадлежит множеству $\bigcup_{k=1}^p \hat{P}(G_{m+k})$.

В противном случае, если найдется вершина $i_b \in \bar{Q}_l$ такая, что $1 < b \leq \leq \omega$ и не существует ребра $[i_b, i_c] \in V_b, 1 \leq c \leq b-1$, то граф (Q, U') есть $G(\pi_l, i_b, i_1, \dots, i_{b-1}, i_{b+1}, \dots, i_\omega, j_a, \pi)$ и принадлежит множеству $\bigcup_{k=1}^p \hat{P}(G_{m+k})$. Если же такой вершины не существует, то для каждого ребра $[i_\nu, j_a] \in V_l, 1 \leq \nu \leq \omega$, в графе (Q, U') существует путь, который заканчивается в вершине i_ν и проходит через некоторую вершину $i_d \in \hat{Q}_l, 1 \leq d \leq \omega$. Следовательно, вес этого пути не меньше $\bar{t}_{id}(Q, U_l)$, причем $\bar{t}_{id}(Q, U_l) \geq \bar{t}_{ja}(Q, U_l)$. Поскольку $\bar{t}_i(Q, U'') \leq \bar{t}_i(Q, U')$, $i \in Q$, и $\bar{t}_{ja}(Q, U'') < \bar{t}_{ja}(Q, U')$ при $(Q, U'') = G(\pi_l, j_a, i_1, i_2, \dots, i_\omega, \pi)$, то расписание s не является компактным.

3.10. Рассмотрим некоторые оценки сложности метода ветвей и границ. Напомним, что через (Z, W) обозначено дерево (Z_m, W_m) , полученное на последней итерации метода, т.е. когда $\tilde{Z}_m = \bar{Z}_m$. Положим $\tilde{Z} = \tilde{Z}_m = \bar{Z}_m$.

Введем в рассмотрение дерево (Z', W') , полученное из (Z, W) в результате последовательного ветвления (на шаге б)) каждой вершины G_l из множества $\tilde{Z}$ до тех пор, пока не будут сориентированы все ребра графа G_l . Таким образом, дерево (Z, W) является порожденным подграфом дерева (Z', W') , и в качестве показателя сложности метода ветвей и границ можно использовать мощность $|Z \setminus \tilde{Z}|$ множества вершин, которые были подвергнуты ветвлению при реализации этого метода.

Вершину $G_l \in Z'$ будем называть минимальной тогда и только тогда, когда соотношение $G_k \rightarrow G_l$ не выполняется ни для какой вершины $G_k \in \in Z'$. Множество всех минимальных вершин обозначим через $\hat{Z}$. Пусть $\hat{Z} = \{G_k \in Z' \mid f_0(G_k) < f_0(G_0) \leq f_0(G_k)\}$. Нетрудно показать, что множество $\hat{Z} \cap \tilde{Z} \subseteq Z \setminus \tilde{Z}$.

Действительно, если граф $G_l \in \hat{Z} \cap \tilde{Z}$ выбран на шаге 2) описанной в п. 3.2 схемы ветвей и границ, то после шага 3) будет выполнен шаг 4), поскольку $f_0(G_l) \neq f_0(G_l)$, после шага 4) — шаг 5), поскольку $f_0(G_l) < < f(G_0) \leq r_m$, а после шага 5) — шаг 6), поскольку $G_l \in \tilde{Z}$. Таким образом, если $G_l \in Z$, то $G_l \notin \tilde{Z}$.

Покажем, что граф G_l принадлежит множеству Z . Рассмотрим путь $G_0, G_d, \dots, G_l$ в графе (Z', W') . Из соотношения (3.3) получаем $f_0(G_0) \leq \leq f_0(G_d) \leq \dots \leq f_0(G_l)$. Следовательно, все вершины этого пути принадлежат множеству $\hat{Z}$, а поскольку $G_0 \in Z$, то эти вершины принадлежат и множеству Z .

Таким образом, справедливо соотношение $|Z \setminus \tilde{Z}| \geq |\hat{Z} \cap \tilde{Z}|$.

3.11. Предположим, что используется стратегия, основанная на функции $\psi(G_h), G_h \in Z'$. Будем говорить, что отношение доминирования $\rightarrow$ *согласовано с функцией* $\psi(G_h), G_h \in Z'$, если из условия $G_k \rightarrow G_l$ следует неравенство $\psi(G_k) < \psi(G_l)$ для любого предшественника G_k вершины G_l в графе (Z', W') . Отношение $\rightarrow$ *согласовано с оценками* $f_0(G_h), G_h \in Z'$, если из $G_k \rightarrow G_l$ следует $f_0(G_k) \leq f_0(G_l)$. Отношение $\rightarrow$ *согласовано с графом* (Z', W') , если из $G_k \rightarrow G_l$ следует, что для любого потомка G_l вершины G_l найдется потомок G_k' вершины G_k такой, что $G_k' \rightarrow G_l$.

Покажем, что *если отношение доминирования $\rightarrow$ согласовано с $\psi(G_h), f_0(G_h)$ и (Z', W') , то $Z \setminus \bar{Z} \subseteq \bar{Z}$.*

Пусть для некоторой итерации на шаге 2) описанной в п. 3.2 схемы выбран граф $G_l \in \bar{Z}_m \setminus \bar{Z}_m$ и существует граф $G_k \in Z'$ такой, что $G_k \rightarrow G_l$. По определению отношения $\rightarrow$ в дереве (Z_m, W_m) не существует пути из G_k в G_l . Поскольку отношение $\rightarrow$ согласовано с функцией $\psi(G_k), G_k \in Z'$, то либо $G_k \in Z_m$, либо в графе (Z, W) найдется предшественник G_c вершины G_k , который принадлежит множеству $\bar{Z}_m$. Если $G_k \in Z_m$, то на шаге 5) граф G_l будет включен в множество $\bar{Z}_m$. Если $G_k \notin Z_m$, то имеет место одна из следующих возможностей.

а) Граф G_c включен в множество конечных вершин $\bar{Z}_{m'}$, $0 < m' < m$, в силу равенства $f_0(G_c) = f^0(G_c)$. Тогда выполняются соотношения $r_m \leq r_{m'} \leq f_0(G_c) \leq f_0(G_k)$. Поскольку отношение $\rightarrow$ согласовано с $f_0(G_h), G_h \in Z'$, то $f_0(G_k) \leq f_0(G_l)$. Получаем соотношение $f_0(G_l) \geq r_m$, из которого следует, что на шаге 4) вершина G_l должна быть включена в множество $\bar{Z}_m$.

б) Граф G_c включен в множество $\bar{Z}_{m'}$ в силу выполнения неравенства $f_0(G_c) \geq r_{m'}$. Аналогично случаю а) получаем $r_m \leq r_{m'} \leq f_0(G_c) \leq f_0(G_k) \leq f_0(G_l)$. Следовательно, $G_l \in \bar{Z}_m$.

в) Граф G_c включен в множество $\bar{Z}_{m'}$ на шаге 5), т.е. существует граф $G_a \in Z_{m'}$ такой, что $G_a \rightarrow G_c$. Поскольку отношение $\rightarrow$ согласовано с (Z', W') , то найдется потомок G_a' вершины G_a такой, что $G_a' \rightarrow G_k$. Из транзитивности отношения $\rightarrow$ получаем $G_a' \rightarrow G_l$.

Если $G_a' \in Z$, то $G_a' \in Z_m$, так как отношение $\rightarrow$ согласовано с функцией $\psi(G_h)$. Следовательно, граф G_l должен быть включен во множество $\bar{Z}_m$ на шаге 5).

Пусть $G_a' \notin Z$, т.е. некоторый предшественник G_b вершины G_a' содержится во множестве $\bar{Z}_m$. Тогда повторяем приведенные рассуждения, заменяя графы G_k и G_c на графы G_a' и G_b соответственно.

Поскольку отношение доминирования является отношением строгого порядка и множество Z' конечно, то, повторяя эти рассуждения конечное число раз, приходим к заключению, что граф G_l должен быть включен во множество $\bar{Z}_m$.

Следовательно, $Z \setminus \bar{Z} \subseteq \bar{Z}$.

Пусть $\tilde{Z} = \{G_k \in Z' \mid f_0(G_k) = f^0(G_k)\}$, тогда, очевидно, $|Z \setminus \tilde{Z}| \leq |\tilde{Z} \setminus \tilde{Z}|$.

Если отношение доминирования на множестве Z' не задано, то, полагая $\tilde{Z} = Z'$, получаем оценки $|\hat{Z} \cap Z'| \leq |Z \setminus \tilde{Z}| \leq |Z' \setminus \tilde{Z}|$.

3.12. Пример. Рассмотрим задачу построения оптимального по быстродействию расписания обслуживания множества требований $N = \{1, 2, \dots, n\}$ приборами A и B при условии, что каждое требование сначала обслуживается прибором A , затем прибором B (см. § 1 гл. 1). Покажем, как можно за $O(n^2)$ элементарных действий решить эту задачу по описанной схеме ветвей и границ.

На рис. 3.1,а приведен граф $G = (Q, U, V)$ для $n = 4$. Операция по обслуживанию требования $i \in N$ прибором A имеет тот же номер i , а операция по обслуживанию этого требования прибором B имеет номер $i + n$. Фиктивная операция $2n + 1 = 9$ обозначает завершение обслуживания всех требований. Каждой дуге $(i, j) \in U$ приписан вес a_{ij} , равный длительности t_i операции $i \in Q$. Каждому ребру $[i, j] \in V$ сопоставлена пара весов $a_{ij} = t_i$ и $a_{ji} = t_j$ (веса ребер на рисунке не указаны).

Для организации процесса ветвления будем использовать процедуру, описанную в п. 3.8, причем одновременно с упорядочением операции i будем упорядочивать и операцию $i + n$ (поскольку в § 1 гл. 1 показано, что существует оптимальное по быстродействию расписание, при котором требования множества N обслуживаются прибором A в той же последовательности, что и прибором B). Из условия (1.3) гл. 1 следует, что отношение доминирования можно задать следующим образом: $G_k \rightarrow G_l$ тогда и только тогда, когда выполняются условия:

- а) ранги вершин G_k и G_l в дереве (Z, W) равны;
- б) $G_k = G(\pi, i, n + i)$ и $G_l = G(\pi, j, n + j)$, $i \neq j$;
- в) либо $\min\{t_i, t_{n+j}\} < \min\{t_j, t_{n+i}\}$, либо $\min\{t_i, t_{n+j}\} = \min\{t_j, t_{n+i}\}$ и $i < j$.

Очевидно, что среди вершин одного и того же ранга дерева (Z, W) существует единственная минимальная вершина. Следовательно, если отношение $\rightarrow$ согласовано с функцией $\psi(G_h)$, то выполняется неравенство $|Z \setminus \tilde{Z}| \leq n - 1$, и в вычислении оценок $f_0(G_k)$ и $f^0(G_k)$, $G_k \in Z$, нет необходимости.

На рис. 3.1,б приведено дерево (Z, W) , полученное при решении этой задачи для графа G , представленного на рис. 3.1,а. Если граф G_l получен из G_k в результате упорядочения требования $i \in N$, то дуге $(G_k, G_l) \in W$ приписан параметр i .

На рис. 3.1,в представлен граф $G_9 \in \hat{P}(G)$. Допустимое относительно этого графа активное расписание является оптимальным по быстродействию.

3.13. Отметим, что если процесс вычислений будет прекращен, когда $\tilde{Z}_m \setminus \bar{Z}_m \neq \emptyset$, то нет гарантии, что наилучшее из полученных расписаний (допустимое относительно рекордного графа) является оптимальным. В этом случае можно вычислить (апостериорные) погрешности наилучшего из полученных к данному шагу (рекордного) расписания.

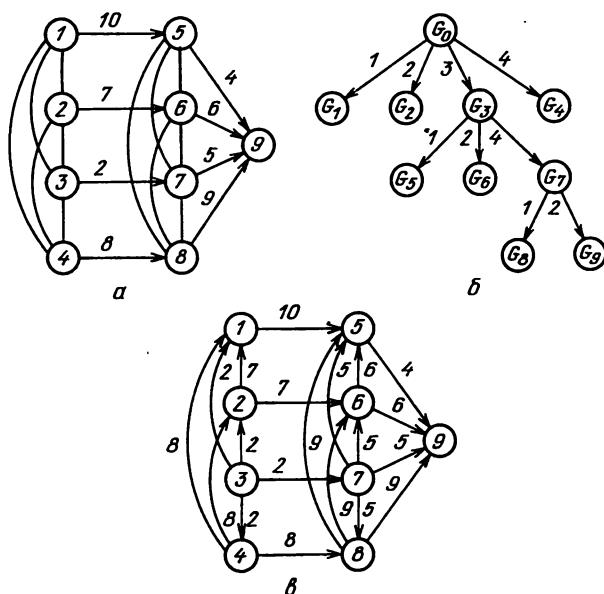


Рис. 3.1

Пусть $f_m = \min \{f_0(G_k) \mid G_k \in \tilde{Z}_m \setminus \bar{Z}_m\}$. Очевидно, абсолютная погрешность рекордного расписания не превосходит значения $r_m - f_m$, а относительная погрешность — значения $(r_m - f_m)/f_m$.

Существенно более эффективными являются приближенные методы, в которых вычислительный процесс с самого начала ориентирован на получение расписаний с заданной (*априорной*) абсолютной или относительной погрешностью.

Так, для получения расписания с абсолютной погрешностью ϵ вместо соотношения (3.1) на шаге 4) описанной в п. 3.2 схемы достаточно использовать более слабое условие $r_m \leq f_0(G_1) + \epsilon$. Аналогично для получения расписания с относительной погрешностью ϵ^0 вместо (3.1) достаточно использовать соотношение $r_m \leq f_0(G_1) + \epsilon^0 f_0(G_1)$. Использование указанных соотношений, как правило, обеспечивает значительное сокращение процесса ветвления.

Эффективность метода ветвей и границ существенно зависит от того, насколько полно учитывается специфика решаемой задачи, а также получаемая в процессе решения задачи информация. Определенное распространение получили так называемые *гибридные алгоритмы*, в которых на разных итерациях используются различные алгоритмы вычисления оценок $f_0(G_k)$ и $f^0(G_k)$, различные функции $\psi(G_k)$, различные способы ветвления и т.д. В результате удастся лучше приспособить вычислительный процесс к особенностям рассматриваемого на данной итерации графа G_k и построенного дерева (Z_m, W_m) .

§ 4. Оптимизация обслуживающей системы

В данном параграфе рассматривается задача оптимального выбора необходимого количества приборов, *распределения* заданного множества операций по выбранным приборам и *составления расписания* их выполнения.

4.1. Рассмотрим следующую обслуживающую систему. Множество $M = \{1, 2, \dots, M\}$ приборов разбито на m попарно непересекающихся непустых подмножеств $M_1, M_2, \dots, M_m$. Каждое множество M_k состоит из M_k одинаковых приборов. Множество $Q = \{1, 2, \dots, q\}$ операций также разбито на m попарно непересекающихся непустых подмножеств $Q_1, Q_2, \dots, Q_m$. Для выполнения операций $i \in Q_k$ может быть использован любой из приборов множества M_k . Каждая операция $i \in Q$ выполняется без прерываний.

Известны длительности выполнения операций, длительности пуско-наладочных, переналадочных и транспортных работ. На множестве операций может быть задано отношение строгого порядка, регламентирующее возможную последовательность их выполнения. Могут быть указаны возможности совмещения процессов выполнения операций во времени и т.п. Предполагается, что все условия и ограничения представимы в виде обобщенной временной модели (Q, U) (см. § 1 этой главы).

В задаче требуется:

1) определить число $r_k, r_k \leq M_k, k = 1, \dots, m$, приборов множества M_k , используемых в процессе выполнения операций;

2) распределить операции множества $Q_k, k = 1, \dots, m$, по этим приборам (т.е. разбить множество Q_k на r_k попарно непересекающихся непустых подмножеств $Q_k^1, Q_k^2, \dots, Q_k^{r_k}$);

3) построить расписание выполнения операций (т.е. для каждой операции $i \in Q$ указать момент ее начала $t_i^0 \geq 0$ или окончания $\bar{t}_i$). При этом значение заданной неубывающей по каждому из аргументов функции $\Phi(\bar{t}_1, \bar{t}_2, \dots, \bar{t}_q; r_1, r_2, \dots, r_m)$ должно быть минимальным.

4.2. Введем в рассмотрение взвешенный смешанный граф $G = (Q, U, V)$, где (Q, U) — указанная выше обобщенная временная модель, а ребро $[i, j]$ принадлежит V тогда и только тогда, когда операции i и j принадлежат одному и тому же множеству $Q_k, 1 \leq k \leq m$.

Если операции i и j принадлежат множеству Q_k и назначены на один и тот же прибор множества M_k , то в зависимости от порядка их выполнения должно иметь место одно из соотношений $t_j^0 - t_i^0 \geq a_{ij}$ или $t_i^0 - t_j^0 \geq a_{ji}$, где a_{ij} и a_{ji} — заданные неотрицательные действительные числа (веса, приписанные ребру $[i, j]$).

Если операции i и j принадлежат множеству Q_k и назначены на разные приборы множества M_k , то не требуется выполнения указанных соотношений.

В первом случае ребро $[i, j] \in V$ заменяется дугой (i, j) веса a_{ij} или дугой (j, i) веса a_{ji} . Во втором случае это ребро удаляется. Удаление ребра $[i, j]$ интерпретируется как назначение операций i и j на разные приборы множества M_k .

4.3. Если $M_k = 1$, $k = 1, \dots, m$, то очевидно, что $r_k = 1$ и $Q_k^1 = Q_k$, т.е. все операции множества Q_k выполняются одним и тем же прибором. Такая задача рассматривалась в предыдущих параграфах этой главы.

Для ее решения достаточно построить множество $\hat{P}(G)$ ориентированных графов и выбрать из него граф (Q, U') , для которого значение функции $\Phi(\bar{t}_1(Q, U'), \bar{t}_2(Q, U'), \dots, \bar{t}_q(Q, U'); 1, 1, \dots, 1)$ минимально. Величина $\bar{t}_i(Q, U')$ означает *наиболее ранний срок* завершения операции $i \in Q$ относительно графа (Q, U') . Напомним, что графы множества $\hat{P}(G)$ не содержат контуров положительного веса и получаются из графа G в результате замены каждого ребра $[i, j] \in V$ дугой (i, j) веса a_{ij} или дугой (j, i) веса a_{ji} и последующего удаления всех кратных дуг за исключением дуг наибольшего веса.

4.4. Пусть хотя бы для одного k , $1 \leq k \leq m$, значение $M_k > 1$.

Зададим число $r_k \leq M_k$ используемых приборов множества $\mathcal{M}_k$, и разделим операции множества Q_k по r_k приборам: $Q_k = \bigcup_{l=1}^{r_k} Q_k^l$, где Q_k^l , $l = 1, \dots, r_k$, — непустые попарно непересекающиеся подмножества множества Q_k , $1 \leq k \leq m$. В результате из смешанного графа G получим смешанный граф $G' = (Q, U, V')$, $V' \subseteq V$, в котором $[i, j] \in V'$ тогда и только тогда, когда операции i и j принадлежат одному и тому же множеству Q_k^l , $1 \leq l \leq r_k$, $1 \leq k \leq m$.

Каждый (ориентированный) граф из множества $\hat{P}(G')$ однозначно определяет активное расписание, допустимое относительно графа G' , и наоборот. Наилучшее среди допустимых относительно графа G' расписание определяет граф $(Q, U') \in \hat{P}(G')$, для которого минимально значение функции $\Phi(\bar{t}_1(Q, U'), \bar{t}_2(Q, U'), \dots, \bar{t}_q(Q, U'); r_1, r_2, \dots, r_m)$.

Очевидно, что для решения задачи достаточно:

а) построить все смешанные графы $G^1, G^2, \dots, G^\lambda$, соответствующие различным наборам чисел $r_k \leq M_k$, $k = 1, \dots, m$, используемых приборов и различным распределениям операций по приборам;

б) построить множество ориентированных графов $\hat{P}(G^\nu)$ для каждого смешанного графа G^ν , $\nu = 1, \dots, \lambda$;

в) выбрать из множества $\hat{R}(G) = \bigcup_{\nu=1}^{\lambda} \hat{P}(G^\nu)$ граф (Q, U') , которому соответствует наименьшее значение целевой функции.

4.5. Для сокращения трудоемкости решения задачи целесообразно совместить процессы *распределения* операций по приборам с *упорядочением* операций, назначенных на один и тот же прибор.

Выбирая последовательно ребра графа G и применяя к каждому из них (для определенности к ребру $[i, j]$) одно из трех возможных преобразований (удаление ребра, замену его дугой (i, j) веса a_{ij} либо дугой (j, i) веса a_{ji}), получаем последовательность $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_{|V|})$ преобразований графа G . Здесь α_l указывает, какое именно ребро выбрано на l -ом шаге и какое из преобразований к нему применено.

Ориентированный граф, полученный из графа G в результате последовательности преобразований α и последующего удаления всех кратных

дуг за исключением дуг наибольшего веса, обозначим через $G[\alpha] = (Q, U[\alpha])$.

Последовательность α преобразований графа G будем называть *допустимой*, если $G[\alpha] \in \hat{R}(G)$.

Нетрудно видеть, что $G[\alpha] \in \hat{R}(G)$ тогда и только тогда, когда он не содержит контуров положительного веса, для каждого $k = 1, \dots, m$ число $r_k[\alpha]$ компонент связности порожденного подграфа $(Q_k, U_k[\alpha])$ графа $(Q, U[\alpha] \setminus U)$ не превосходит M_k и эти компоненты связности являются полными ориентированными графами (каждая пара вершин такого графа соединена дугой).

Каждый граф $G[\alpha]$ из множества $\hat{R}(G)$ однозначно определяет число $r_k[\alpha]$ приборов множества $\mathcal{M}_k$, $1 \leq k \leq m$, единственное (с точностью до нумерации приборов) распределение операций множества Q_k по $r_k[\alpha]$ приборам, $1 \leq k \leq m$, и единственное активное расписание выполнения операций соответствующими приборами. При этом расписании $\bar{t}_i = \bar{t}_i(Q, U[\alpha])$, $i \in Q$. Таким образом, задача сводится к поиску допустимой последовательности α^* преобразований графа G , при которой будет получено минимальное значение целевой функции.

4.6. При конструктивном подходе к поиску α^* желательно располагать условиями, при которых заданная частичная последовательность $\alpha' = (\alpha_1, \alpha_2, \dots, \alpha_l)$, $l < |V|$, может быть достроена до полной допустимой последовательности $\alpha = (\alpha', \alpha_{l+1}, \dots, \alpha_{|V|})$.

Сопоставим последовательности α' и числу k , $1 \leq k \leq m$, вспомогательный граф $H_k[\alpha'] = (Q_k[\alpha'], V_k[\alpha'])$. Этот граф строится следующим образом. Если α_1 представляет собой удаление ребра $[i, j]$, $i, j \in Q_k$, из графа G , то граф $H_k[\alpha_1]$ получается из графа $H_k = (Q_k, \phi)$ в результате добавления ребра $[i, j]$. Если α_1 представляет собой замену ребра $[i, j]$ одной из дуг (i, j) или (j, i) , то граф $H_k[\alpha_1]$ получается из графа H_k в результате отождествления вершин i и j в одну вершину. Если α_1 — преобразование ребра $[i, j]$, $i, j \in Q_c$, $k \neq c$, то $H_k[\alpha_1] = H_k$.

Пусть построен граф $H_k[\alpha'']$, $\alpha'' = (\alpha_1, \alpha_2, \dots, \alpha_{p-1})$, $p \leq l$; α_p представляет собой преобразование ребра $[i, j]$, $i, j \in Q_k$, в графе G ; a и b — вершины графа $H_k[\alpha'']$, в которые отождествлены соответственно вершины i и j либо $a = i$ и (или) $b = j$. Если α_p — удаление ребра $[i, j]$, то граф $H_k[(\alpha'', \alpha_p)]$ получается из графа $H_k[\alpha'']$ в результате добавления ребра $[a, b]$. Аналогично, если α_p — замена ребра $[i, j]$ одной из дуг (i, j) или (j, i) , то граф $H_k[(\alpha'', \alpha_p)]$ получается из графа $H_k[\alpha'']$ в результате отождествления вершин a и b . Если α_p — преобразование ребра $[i, j]$, $i, j \in Q_c$, $k \neq c$, то $H_k[(\alpha'', \alpha_p)] = H_k[\alpha'']$.

Нетрудно видеть, что если α — допустимая последовательность преобразований графа G , то графы $H_k[\alpha]$, $k = 1, \dots, m$, являются полными, не содержат петель и число $|Q_k[\alpha]|$ их вершин не превосходит M_k .

Обозначим через $\chi(H_k[\alpha'])$ *хроматическое число* графа $H_k[\alpha']$, т.е. наименьшее число цветов, которые можно приписать вершинам этого графа так, что никакие две смежные вершины не получают одинакового цвета. Пусть граф $G[\alpha'] = (Q, U[\alpha'], V[\alpha'])$ получен в результате последователь-

ности α' преобразований графа G и последующего удаления всех кратных дуг за исключением дуг наибольшего веса. Покажем, что для того чтобы $\hat{R}(G[\alpha']) \neq \phi$, необходимо, чтобы

а) граф $(Q, U[\alpha'])$ не содержит контуров положительного веса;

б) для каждого $k = 1, \dots, t$ граф $H_k[\alpha']$ не содержит петлю;

в) выполнялись соотношения $\chi(H_k[\alpha']) \leq M_k, k = 1, \dots, t$.

Действительно, для всех $k = 1, \dots, t$ имеем $|Q_k[\alpha]| \geq \chi(H_k[\alpha'])$, и если граф $H_k[\alpha']$ содержит петлю, то и граф $H_k[\alpha]$ содержит петлю. Если граф $(Q, U[\alpha'])$ содержит контур положительного веса, то этот же контур содержится и в графе $(Q, U[\alpha])$.

Напомним, что ребро $[i, j] \in V$ называется конфликтным, если для него выполняется соотношение (3.2) (см. п.3.3 этой главы). Очевидно, что для того чтобы $\hat{R}(G[\alpha']) \neq \phi$, достаточно, чтобы выполнялись перечисленные выше условия и граф $G[\alpha']$ не содержал конфликтных ребер.

4.7. Для сокращения перебора графов множества $\hat{R}(G)$ можно воспользоваться методом ветвей и границ.

Пусть $[i, j]$ — некоторое конфликтное ребро графа G . Через α_1^1, α_2^1 и α_3^1 обозначим соответственно удаление из графа G ребра $[i, j]$, замену этого ребра дугой (i, j) веса a_{ij} и дугой (j, i) веса a_{ji} . На первой итерации получаем разбиение множества $\hat{R}(G)$ на подмножества $\hat{R}(G[\alpha_1^1])$, $\hat{R}(G[\alpha_2^1])$ и $\hat{R}(G[\alpha_3^1])$. Затем одно из полученных множеств $\hat{R}(G[\alpha_h^1])$, $1 \leq h \leq 3$, в свою очередь, разбивается на подмножества вида $\hat{R}(G[\alpha_h^1, \alpha_2^1])$, $\hat{R}(G[\alpha_h^1, \alpha_3^1])$ и $\hat{R}(G[\alpha_2^1, \alpha_3^1])$, где α_2^1, α_3^1 и α_2^2 — соответствующие преобразования некоторого конфликтного ребра $[a, b]$ графа $G[\alpha_h^1]$, и т.д.

Пусть на очередной итерации рассматривается граф $G[\alpha'] = (Q, U[\alpha'], V[\alpha'])$.

Если для графа $G[\alpha']$ не выполняется хотя бы одно из условий а), б) или в) предыдущего пункта, то этот граф исключается из дальнейшего рассмотрения, поскольку $\hat{R}(G[\alpha']) = \phi$.

Если же указанные условия выполняются и граф $G[\alpha']$ содержит хотя бы одно конфликтное ребро, то можно вычислить оценку снизу значения целевой функции. Для этого можно воспользоваться соотношением

$$\begin{aligned} & \Phi(\bar{t}_1(Q, U[\alpha]), \dots, \bar{t}_q(Q, U[\alpha]); |Q_1[\alpha]|, \dots, |Q_m[\alpha]|) \geq \\ & \geq \Phi(\bar{t}_1(Q, U[\alpha']), \dots, \bar{t}_q(Q, U[\alpha']); \chi(H_1[\alpha']), \dots, \chi(H_m[\alpha'])). \end{aligned} \quad (4.1)$$

Справедливость этого соотношения для любой допустимой последовательности $\alpha = (\alpha', \alpha_{i+1}, \dots, \alpha_{|V|})$ следует из неравенств $\bar{t}_i(Q, U[\alpha]) \geq \bar{t}_i(Q, U[\alpha']), i = 1, \dots, q, |Q_k[\alpha]| \geq \chi(H_k[\alpha']), k = 1, \dots, m$, и неубывания целевой функции по каждому из аргументов.

Если условия а), б) и в) выполняются для графа $G[\alpha']$ и в нем нет конфликтных ребер, то не вызывает затруднений построение последовательности α , для которой соотношение (4.1) выполняется как равенство.

Для этого достаточно заменить все ребра $[i, j]$ множества $V[\alpha']$ дугами (i, j) так, чтобы выполнялись соотношения $\bar{t}_i(Q, U[\alpha']) \leq \bar{t}_j(Q, U[\alpha'])$.

4.8. Пример. В гибкой производственной системе, состоящей из станков с ЧПУ (приборов) трех типов ($m = 3$), необходимо обработать три детали (обслужить три требования). Множества станков $\mathcal{M}_1 = \{1, 2, 3\}$, $\mathcal{M}_2 = \{4, 5\}$ и $\mathcal{M}_3 = \{6\}$. Обработка первой детали состоит в последовательном выполнении множества операций $\{1, 2, 3\}$, второй – множества операций $\{4, 5\}$ и третьей – множества операций $\{6, 7, 8\}$.

Множества операций $Q_1 = \{1, 4, 6\}$, $Q_2 = \{2, 5, 8\}$ и $Q_3 = \{3, 7\}$. Фиктивная операция 9 означает завершение обработки всех деталей. Зада- ны длительности выполнения операций: $t_1 = 3$, $t_2 = 2$, $t_3 = t_4 = 7$, $t_5 = 5$, $t_6 = 10$, $t_7 = 8$, $t_8 = 1$, $t_9 = 0$.

Целевая функция $\Phi(\bar{t}_1, \bar{t}_2, \dots, \bar{t}_9; r_1, r_2, r_3) = \bar{t}_9 + r_1 + 4r_2 + 3r_3$. Она отражает стремление минимизировать общее время обработки деталей с учетом длительности пуско-наладочных работ для каждого из используемых станков.

На рис. 4.1, а представлен смешанный граф $G = (Q, U, V)$ для данной обслуживающей системы. Вес a_{ij} дуги $(i, j) \in U$ равен t_i . Каждому ребру $[i, j] \in V$ сопоставлена пара весов $a_{ij} = t_i$ и $a_{ji} = t_j$. Веса ребер на рисунке не указаны.

Для решения задачи воспользуемся методом ветвей и границ, схема которого описана в предыдущем параграфе.

Пусть на очередной итерации рассматривается граф $G_l = (Q, U_l, V_l)$. В качестве ребра, определяющего разбиение множества $\hat{R}(G_l)$ на подмножества, из множества конфликтных ребер графа G_l выбирается ребро $[i, j]$,

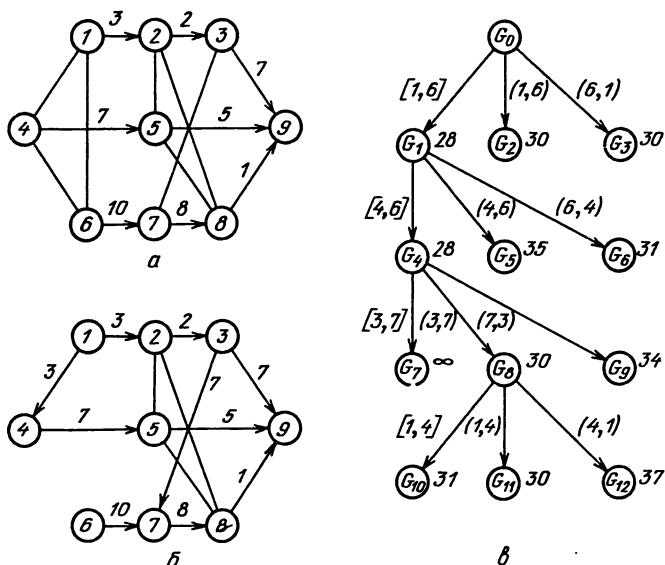


Рис. 4.1

для которого величина $\min \{h_{ij}, h_{ji}\}$ (см. п. 3.7 этой главы) принимает наибольшее значение.

Для оценки целевой функции используется соотношение (4.1). В качестве стратегии метода используется поиск по наилучшей оценке (см. п. 3.4 предыдущего параграфа).

На рис. 4.1, в представлено дерево (Z, W) , полученное при решении данной задачи. Граф $G_0 = G$. Каждой дуге (G_k, G_l) из множества W приписано либо ребро $[i, j]$, либо дуга (i, j) или (j, i) , в зависимости от того, получен ли граф G_l из графа G_k в результате удаления ребра $[i, j]$ либо замены этого ребра дугой (i, j) или (j, i) . Каждому графу $G_l \in Z$ при $l \neq 7$ приписана оценка целевой функции. Графу G_7 приписан символ ∞ , поскольку $\chi(H_3[\alpha']) > M_3$ и, следовательно, $\hat{R}(G_7) = \phi$. Здесь $G[\alpha'] = G_7$.

Граф G_{11} , представленный на рис. 4.1, б, не содержит конфликтных ребер и определяет решение задачи: $r_1 = 2, r_2 = r_3 = 1, Q_1^1 = \{1, 4\}, Q_2^1 = \{6\}, Q_3^1 = \{2, 5, 8\}, Q_4^1 = \{3, 7\}, t_1^0 = 0, t_2^0 = 3, t_3^0 = 5, t_4^0 = 3, t_5^0 = 10, t_6^0 = 0, t_7^0 = 12, t_8^0 = 20, t_9^0 = 21$. Оптимальное значение целевой функции равно $21 + 2 + 4 + 3 = 30$.

§ 5. Устойчивость оптимальных расписаний

В заключение рассмотрим некоторые вопросы, связанные с устойчивостью оптимальных по быстродействию расписаний относительно длительностей выполняемых операций.

5.1. Пусть $G = (Q, U, V)$ — конечный смешанный граф без петель, кратных дуг и кратных ребер, $Q = \{1, 2, \dots, q\}$ — множество вершин (операций), U — множество дуг и V — непустое множество ребер. Каждой дуге $(i, j) \in U$ сопоставлен вес a_{ij} , равный длительности $t_i \geq 0$ операции $i \in Q$. Каждому ребру $[i, j] \in V$ приписана пара неотрицательных весов $a_{ij} = t_i$ и $a_{ji} = t_j$.

Как показано в § 1 этой главы, существует взаимно однозначное соответствие между множеством всех активных расписаний, допустимых относительно графа G , и множеством $P(G) = \{G_1, G_2, \dots, G_p\}$ всех бесконтурных графов, порождаемых смешанным графом G в результате замены каждого ребра $[i, j] \in V$ одной из дуг (i, j) веса a_{ij} или (j, i) веса a_{ji} . Будем предполагать, что граф (Q, U) не содержит контуров, поскольку в противном случае $P(G) = \phi$. Графу $G_k = (Q, U_k) \in P(G)$ соответствует активное расписание s_k , при котором момент $t_i^0(s_k)$ начала операции $i \in Q$ равен наиболее раннему сроку начала операции i относительно этого графа.

Положим $t = (t_1, t_2, \dots, t_q)$. Через $l^t(\mu)$ обозначим вес пути μ в графе $G_k \in P(G)$, а через $\{\mu\}$ — множество вершин (операций), через которые проходит путь μ , без последней вершины этого пути. Путь μ будем называть *доминирующим*, если в графе G_k не существует пути ν , для которого $\{\mu\} \subset \{\nu\}$. Путь наибольшего веса в графе принято называть *критическим*.

Пусть H и H_k — множества всех доминирующих путей графов (Q, U) и $G_k \in P(G)$ соответственно. Очевидно, $H \subseteq H_k$ для всех $k = 1, \dots, p$ и хотя бы один критический путь в графе G_k является доминирующим. Оптимальное по быстродействию расписание s_h определяется графом

$G_h \in P(G)$ с наименьшим весом критического пути:

$$\max_{\mu \in H_h} l^t(\mu) = \min_{1 \leq k \leq p} \max_{\nu \in H_k} l^t(\nu). \quad (5.1)$$

5.2. Обозначим через $S(t)$ множество всех номеров h графов $G_h \in P(G)$ с наименьшим весом критического пути. Если $h \in S(t)$, то граф G_h будем называть *оптимальным*. Исследуем вопрос о том, насколько можно изменять длительности операций t (независимо одна от другой), чтобы граф G_h , $h \in S(t)$, оставался оптимальным.

Пусть R_q — множество неотрицательных действительных q -мерных векторов с чебышевской метрикой и $r(t, t') = \max_{i \in Q} |t_i - t'_i|$ — расстояние от

вектора t до вектора $t' = (t'_1, t'_2, \dots, t'_q)$. Замкнутый шар $\bar{\theta}_\gamma(t)$ радиуса γ с центром в t будем называть *шаром устойчивости графа* G_h , $h \in S(t)$, если для любого вектора $t' \in \bar{\theta}_\gamma(t) \cap R_q$ выполняется $h \in S(t')$.

Наибольшее значение радиуса γ шаров устойчивости $\bar{\theta}_\gamma(t)$ графа G_h будем называть *радиусом устойчивости графа* G_h и обозначать через $\gamma_h(t)$.

5.3. Покажем, что для того чтобы $\gamma_h(t) = \infty$, необходимо и достаточно, чтобы для любого пути $\mu \in H_h \setminus H$ и любого графа $G_k \in P(G)$ существовал путь $\nu \in H_k$ такой, что $\{\mu\} \subseteq \{\nu\}$.

Пусть $\gamma_h(t) = \infty$, а путь $\mu' \in H_h \setminus H$ и граф $G_{k'} \in P(G)$ таковы, что ни для какого пути $\nu \in H_{k'}$, соотношение $\{\mu\} \subseteq \{\nu\}$ не выполняется.

Выберем число ϵ , для которого $0 < \epsilon \leq \max_{i \in Q} t_i$, и положим $t' = (t'_1, t'_2, \dots, t'_q)$, где

$$t'_i = \begin{cases} \epsilon, & \text{если } i \in \{\mu'\}, \\ 0, & \text{если } i \notin \{\mu'\}. \end{cases}$$

Для любого пути $\nu \in H_{k'}$ имеет место равенство $l^{t'}(\nu) = \epsilon |\{\mu'\} \cap \{\nu\}|$. Поскольку включение $\{\mu'\} \subseteq \{\nu\}$ не выполняется, то $l^{t'}(\nu) < l^{t'}(\mu')$. Следовательно,

$$\max_{\nu \in H_{k'}} l^{t'}(\nu) < l^{t'}(\mu') = \max_{\mu \in H_h} l^{t'}(\mu)$$

и $h \notin S(t)$. Получили противоречие: $\gamma_h(t) < \max_{i \in Q} t_i < \infty$.

Докажем достаточность утверждения. Для любого сколь угодно большого $\epsilon > 0$ возьмем произвольный вектор $t' \in \bar{\theta}_\epsilon(t) \cap R_q$. Пусть

$$\max_{\mu \in H_h} l^{t'}(\mu) = l^{t'}(\mu_0).$$

Если $\mu_0 \in H$, то при любом $k = 1, \dots, p$

$$l^{t'}(\mu_0) \leq \max_{\nu \in H_k} l^{t'}(\nu).$$

Если $\mu_0 \in H_h \setminus H$, то по условию утверждения для любого $k = 1, \dots, p$ существует путь $\nu_0 \in H_k$ такой, что $\{\mu_0\} \subseteq \{\nu_0\}$ и, следовательно,

$$l^{t'}(\mu_0) \leq l^{t'}(\nu_0) \leq \max_{\nu \in H_k} l^{t'}(\nu).$$

Таким образом, в обоих случаях $h \in S(t')$.

Как следствие доказанного утверждения получаем оценку: если $\gamma_h(t) < \infty$, то $\gamma_h(t) < \max_{i \in Q} t_i$.

5.4. Пусть $\gamma_h(t) < \infty$. Выведем формулу для вычисления радиуса устойчивости.

Поскольку шар устойчивости $\bar{\theta}_\gamma(t)$ графа G_h является замкнутым, то

$$\gamma_h(t) = \inf \{r(t, t') \mid t' \in R_q, h \notin S(t')\}.$$

Учитывая (5.1), получаем

$$\gamma_h(t) = \inf \{r(t, t') \mid t' \in R_q, \max_{\mu \in H_h} l^{t'}(\mu) > \min_{k \neq h} \max_{\nu \in H_k} l^{t'}(\nu)\}.$$

Определим ближайший к t вектор $t' \in R_q$, для которого

$$\max_{\mu \in H_h} l^{t'}(\mu) > \max_{\nu \in H_k} l^{t'}(\nu). \quad (5.2)$$

Отметим, что множество H_h в неравенстве (5.2) можно заменить на $H_h \setminus H_k$.

Нетрудно видеть, что для выполнения неравенства

$$l^{t'}(\mu) > l^{t'}(\nu) \quad (5.3)$$

в качестве t' достаточно взять вектор $t'(r)$, полученный из t в результате уменьшения длительностей операций из множества $\{\nu\} \setminus \{\mu\}$ и увеличения длительностей операций из множества $\{\mu\} \setminus \{\nu\}$ на величину

$$r > r_{\nu\mu}^0 = \frac{l^t(\nu) - l^t(\mu)}{|\{\mu\} \cup \{\nu\}| - |\{\mu\} \cap \{\nu\}|}.$$

Очевидно, что для любого q -мерного вектора t' , для которого $r(t, t') \leq r_{\nu\mu}^0$, выполняется соотношение $l^{t'}(\mu) \leq l^{t'}(\nu)$.

При уменьшении компонент вектора t некоторые из них могут стать отрицательными, т.е. возможно $t'(r) \notin R_q$.

Пусть $t_{\nu\mu}^0 = 0$ и $(t_{\nu\mu}^1, t_{\nu\mu}^2, \dots, t_{\nu\mu}^{\omega_{\nu\mu}})$ — неубывающая последовательность длительностей операций множества $\{\nu\} \setminus \{\mu\}$. Вычислим величину

$$r_{\nu\mu} = \max_{0 < \beta \leq \omega_{\nu\mu}} \frac{l^t(\nu) - l^t(\mu) - \sum_{\alpha=0}^{\beta} t_{\nu\mu}^{\alpha}}{|\{\mu\} \cup \{\nu\}| - |\{\mu\} \cap \{\nu\}| - \beta} \quad (5.4)$$

и через β^* обозначим то значение β , при котором правая часть равенства (5.4) достигает наибольшего значения.

Очевидно, неравенство (5.3) будет выполняться, если в качестве t' взять вектор $t(r) \in R_q$, полученный из t в результате увеличения компонент t_i , $i \in \{\mu\} \setminus \{\nu\}$, на величину $r > r_{\nu\mu}$, уменьшения на ту же величину компонент $\{t_{\nu\mu}^{\beta^*+1}, t_{\nu\mu}^{\beta^*+2}, \dots, t_{\nu\mu}^{\omega_{\nu\mu}}\}$ и замены компонент $\{t_{\nu\mu}^1, t_{\nu\mu}^2, \dots, t_{\nu\mu}^{\beta^*}\}$ нулями. Очевидно, для любого вектора $t' \in R_q$, для которого $r(t, t') \leq r_{\nu\mu}$, справедливо соотношение $l^{t'}(\mu) \leq l^{t'}(\nu)$.

Таким образом, для любого вектора $t' \in R_q$ из условия (5.3) следует неравенство $r(t, t') > r_{\nu\mu}$, из условия (5.2) следует неравенство

$$r(t, t') > \max_{\nu \in H_k} \min_{\mu \in H_h \setminus H_k} r_{\nu\mu}$$

и, наконец, из условия

$$\max_{\mu \in H_h} l^{t'}(\mu) > \min_{k \neq h} \max_{\nu \in H_k} l^{t'}(\nu)$$

следует неравенство

$$r(t, t') > \min_{k \neq h} \max_{\nu \in H_k} \min_{\mu \in H_h \setminus H_k} r_{\nu\mu}.$$

В результате получаем оценку снизу радиуса устойчивости $\gamma_h(t)$.

Отметим, что знаменатель в соотношении (5.4) может быть равен нулю только при условии $\{\mu\} \subseteq \{\nu\}$. Поскольку $\gamma_h(t) < \infty$, то в силу п. 5.3 существуют граф $G_k \in P(G)$ и путь $\mu' \in H_h \setminus H$ такие, что для любого пути $\nu \in H_k$, множество $\{\nu\}$ не содержит множества $\{\mu'\}$. Следовательно, величина $r = \min_{k \neq h} \max_{\nu \in H_k} \min_{\mu \in H_h \setminus H_k} r_{\nu\mu}$ конечна.

Покажем, что для любого сколь угодно малого числа $\epsilon > 0$ существует вектор $t^\epsilon \in \bar{\theta}_{r+\epsilon}(t) \cap R_q$ такой, что $h \notin S(t^\epsilon)$.

Пусть $r = \max_{\nu \in H_{k^0}} r_{\nu\mu^0}$ и вектор t^0 получен из t в результате увеличе-

ния компонент t_i , $i \in \{\mu^0\}$, на величину r и замены компонент t_j , $j \notin \{\mu^0\}$, на $\max\{0, t_j - r\}$.

Согласно приведенным рассуждениям $h \in S(t^0)$, $k^0 \in S(t^0)$, $l^{t^0}(\mu^0) = \max_{\mu \in H_h} l^{t^0}(\mu)$ и для любого пути $\nu \in H_{k^0}$ множество $\{\nu\}$ не содержит множества $\{\mu^0\}$.

Через t^ϵ обозначим вектор, полученный из t^0 в результате увеличения на ϵ длительностей операций $\{\mu^0\}$. Нетрудно видеть, что для любого $\epsilon > 0$ имеет место неравенство

$$l^{t^\epsilon}(\mu^0) > \max_{\nu \in H_{k^0}} l^{t^\epsilon}(\nu)$$

и, следовательно, $h \notin S(t^\epsilon)$.

Таким образом, радиус устойчивости может быть вычислен по следующей формуле:

$$\gamma_h(t) = \min_{k \neq h} \max_{\nu \in H_k} \min_{\mu \in H_h \setminus H_k} \max_{0 \leq \beta \leq \omega_{\nu\mu}} \frac{l^t(\nu) - l^t(\mu) - \sum_{\alpha=0}^{\beta} t_{\nu\mu}^\alpha}{|\{\mu\} \cup \{\nu\}| - |\{\mu\} \cap \{\nu\}| - \beta} \quad (5.5)$$

5.5 Пример. На гибком автоматизированном участке, в состав которого входят станок A и станок B , необходимо обработать две детали. Первая деталь вначале обрабатывается на станке A (операция 2), а затем на станке B (операция 3). Вторая деталь вначале обрабатывается на станке B (операция 5), а затем на станке A (операция 6).

На рис. 5.1 представлен смешанный граф $G = (Q, U, V)$ для рассматриваемой обслуживающей системы. Операции 1 и 4 означают поступление на участок первой и второй детали соответственно. Фиктивная операция 7 означает завершение обработки обеих деталей. Вес a_{ij} дуги $(i, j) \in U$ равен длительности t_i операции i . Каждому ребру $[i, j] \in V$ соответствует пара весов $a_{ij} = t_i$ и $a_{ji} = t_j$. Веса ребер на рисунке не указаны. Вектор длительностей операций $t = (40, 50, 40, 30, 70, 60, 0)$.

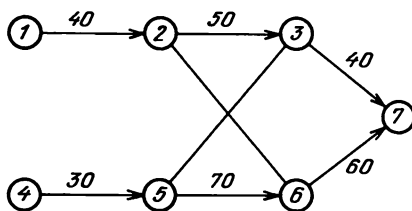


Рис. 5.1

Требуется построить оптимальное по быстродействию расписание s_h обработки двух деталей и определить радиус устойчивости оптимального графа G_h .

Очевидно, $P(G) = \{G_1, G_2, G_3\}$, где $G_1 = (Q, U \cup \{(2, 6), (5, 3)\})$, $G_2 = (Q, U \cup \{(2, 6), (3, 5)\})$, $G_3 = (Q, U \cup \{(5, 3), (6, 2)\})$. Построим множества доминирующих путей $H_1 = \{\mu_1, \mu_2, \mu_3, \mu_4\}$, $H_2 = \{\mu_2, \mu_5\}$, $H_3 = \{\mu_1, \mu_6\}$, где $\mu_1 = (1, 2, 3, 7)$, $\mu_2 = (4, 5, 6, 7)$, $\mu_3 = (1, 2, 6, 7)$, $\mu_4 = (4, 5, 3,$

7), $\mu_5 = (1, 2, 3, 5, 6, 7)$, $\mu_6 = (4, 5, 6, 2, 3, 7)$, и вычислим их веса: $l^t(\mu_1) = 130$, $l^t(\mu_2) = 160$, $l^t(\mu_3) = 150$, $l^t(\mu_4) = 140$, $l^t(\mu_5) = 260$ и $l^t(\mu_6) = 250$.

Таким образом, оптимальными являются граф G_1 и расписание s_1 : $t_1^0(s_1) = t_4^0(s_1) = 0$, $t_2^0(s_1) = 40$, $t_3^0(s_1) = 100$, $t_5^0(s_1) = 30$, $t_6^0(s_1) = 100$, $t_7^0(s_1) = 160$. По формуле (5.5) получаем $\gamma_1(t) = 25$.

5.6. Через H_h^t обозначим множество всех доминирующих критических путей графа $G_h \in P(G)$.

Нетрудно убедиться в том, что существует число $\epsilon > 0$ такое, что ни один путь из множества $H_h \setminus H_h^t$ не станет критическим, если вектор t заменить вектором $t' \in \bar{\theta}_\epsilon(t)$.

Действительно, в качестве ϵ достаточно взять любое положительное число, которое меньше

$$\epsilon_h = \min_{\nu \in H_h \setminus H_h^t} \frac{l_h^t - l^t(\nu)}{2m_h},$$

где $l_h^t = \max_{\mu \in H_h} l^t(\mu)$ и $m_h = \max_{\mu \in H_h} |\{\mu\}|$.

5.7. Покажем, что для того чтобы выполнялось неравенство $\gamma_h(t) > 0$, $h \in S(t)$, необходимо и достаточно, чтобы для любого пути $\mu \in H_h^t \setminus H$ и любого $k \in S(t)$ существовал путь $\nu \in H_k^t$ такой, что $\{\mu\} \subseteq \{\nu\}$.

Пусть $\gamma_h(t) = \epsilon > 0$, а путь $\mu' \in H_h^t$ и граф $G_{k'}$, $k' \in S(t)$, таковы, что ни для какого пути $\nu \in H_{k'}^t$ не выполняется включение $\{\mu'\} \subseteq \{\nu\}$.

Через t' обозначим вектор, полученный из t в результате увеличения длительностей операций $i \in \{\mu'\}$ на величину $\epsilon^0 = \min\{\epsilon_h, \epsilon_{k'}, \epsilon\}$. Очевидно, $t' \in \bar{\theta}_{\epsilon^0}(t) \cap R_q$ и согласно п. 5.6

$$\begin{aligned} \max_{\nu \in H_{k'}} l^{t'}(\nu) &= \max_{\nu \in H_{k'}^t} l^{t'}(\nu) = l^{t'}(\nu) + \\ &+ \epsilon^0 |\{\mu'\} \cap \{\nu\}| = l^{t'}(\mu') + \epsilon^0 |\{\mu'\} \cap \{\nu\}|. \end{aligned}$$

Поскольку включение $\{\mu'\} \subseteq \{\nu\}$ не выполняется, то $|\{\mu'\} \cap \{\nu\}| < |\{\mu'\}|$, $\max_{\nu \in H_{k'}} l^{t'}(\nu) < l^{t'}(\mu')$ и, следовательно, $h \notin S(t')$. Получили противоречие с предположением о том, что $\gamma_h(t) = \epsilon \geq \epsilon^0$.

Докажем достаточность утверждения. Пусть $0 < \epsilon < \min \left\{ \epsilon_h, \frac{l - l_h^t}{2m} \right\}$,

где $l = \min_{k \notin S(t)} l_k^t$ и $m = \max_{1 \leq k \leq p} m_k$.

Для любого вектора $t' \in \bar{\theta}_\epsilon(t) \cap R_q$ согласно п. 5.6 имеет место равенство $\max_{\mu \in H_h} l^{t'}(\mu) = \max_{\mu \in H_h^t} l^{t'}(\mu)$. Поскольку для любого пути $\mu \in H_h^t \setminus H$ и любого $k \in S(t)$ существует путь $\nu \in H_k^t$ такой, что $\{\mu\} \subseteq \{\nu\}$, то

$$\max_{\mu \in H_h^t} l^{t'}(\mu) \leq \max_{\nu \in H_k^t} l^{t'}(\nu).$$

Следовательно,

$$\max_{\mu \in H_h} l^{t'}(\mu) \leq \max_{\nu \in H_k} l^{t'}(\nu).$$

Поскольку $\epsilon < (l - l_h^t)/2m$, то из условия $k \notin S(t)$ следует, что $k \notin S(t')$. Таким образом, $h \in S(t')$ и $\gamma_h(t) > \epsilon > 0$.

Из доказанного утверждения получаем следствие: если $S(t) = \{h\}$, то $\gamma_h(t) > 0$. Аналогично, если $h \in S(t)$ и $H_h^t \subseteq H$, то $\gamma_h(t) > 0$.

5.8. П р и м е р ы. На рис. 5.2, а представлен смешанный граф G для описанной в п. 5.5 обслуживающей системы, когда вектор длительностей операций $t = (60, 50, 40, 0, 40, 20, 0)$. Нетрудно видеть, что $S(t) = \{1, 3\}$ и согласно утверждению из п. 5.7 $\gamma_1(t) > 0$ и $\gamma_3(t) = 0$. Точное значение $\gamma_1(t) = 10$ можно получить по формуле (5.5).

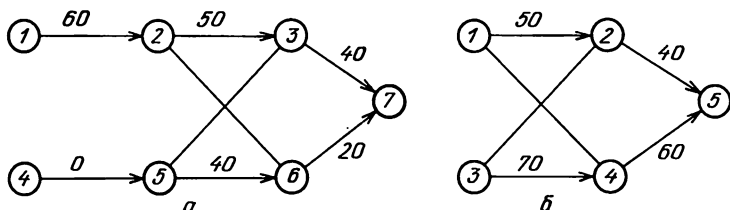


Рис. 5.2

На рис. 5.2, б представлен смешанный граф G с вектором длительностей операций $t = (50, 40, 70, 60, 0)$ при условии одновременного поступления двух деталей в обслуживающую систему. В этом случае оптимальными являются граф $G_1 = (Q, U \cup \{(1, 4), (3, 2)\})$ и расписание $s_1: t_1^0(s_1) = t_3^0(s_1) = 0, t_2^0(s_1) = t_4^0(s_1) = 70, t_5^0(s_1) = 130$. Согласно утверждению из п. 5.3 получаем $\gamma_1(t) = \infty$.

5.9. Наряду с шаром и радиусом устойчивости фиксированного оптимального графа определенный интерес представляют аналогичные понятия относительно всего множества оптимальных графов.

Открытый шар $\theta_\gamma(t)$ радиуса γ с центром в t будем называть *шаром устойчивости* t , если для любого $t' \in \theta_\gamma(t) \cap R_q$ имеет место включение $S(t') \subseteq S(t)$. Верхнюю грань радиуса γ шаров устойчивости t будем называть *радиусом устойчивости* t и обозначать через $\gamma(t)$.

Из данного определения следует, что $\gamma(t) = \infty$ для любого вектора $t \in R_q^0 = \{t \in R_q | S(t) = \{1, 2, \dots, p\}\}$. Множество R_q^0 не пусто, поскольку заведомо содержит нулевой q -мерный вектор 0 . Следовательно, для любого $t \in R_q \setminus R_q^0$ выполняется оценка

$$\gamma(t) \leq r(t, 0) = \max \{t_i | i \in Q\}.$$

Таким образом, для того чтобы $\gamma(t) = \infty$, необходимо и достаточно, чтобы $t \in R_q^0$.

Покажем, что для любого $t \in R_q$ значение $\gamma(t) > 0$. Пусть $t \in R_q \setminus R_q^0$. Выберем число ϵ , для которого $0 < \epsilon < (l - l_h^t)/2m, h \in S(t)$. Очевидно, что для любого $t' \in \theta_\epsilon(t) \cap R_q$ выполняется включение $S(t') \subseteq S(t)$. Следовательно, $\gamma(t) \geq \epsilon > 0$.

Так как ϵ можно выбрать сколь угодно близким к величине $(l - l_h^t)/2m$, то справедлива оценка

$$\gamma(t) \geq (l - l_h^t) / 2m.$$

Поскольку шар устойчивости t является открытым, то $\gamma(t) = \min\{r(t, t') \mid t' \in R_q, k \notin S(t), k \in S(t')\}$ и в силу (5.1) для вычисления $\gamma(t)$ достаточно решить следующую задачу нелинейного программирования:

$$\max_{1 \leq i \leq q} |x_i - t_i| \rightarrow \min, \quad (5.6)$$

$$\min_{1 \leq h \leq p} \max_{\mu \in H_h} l^x(\mu) \geq \min_{k \notin S(t)} \max_{\nu \in H_k} l^x(\nu), \quad (5.7)$$

$$x_i \geq 0, \quad i = 1, \dots, q. \quad (5.8)$$

Очевидно, если $x^* = (x_1^*, x_2^*, \dots, x_q^*)$ — решение задачи (5.6)–(5.8), то $\gamma(t) = r(t, x^*)$.

5.10. Множество всех векторов $t \in R_q$, для которых $h \in S(t)$, будем называть *областью устойчивости графа G_h* и обозначать через T_h . Учитывая (5.1), получаем

$$T_h = \{t \in R_q \mid \max_{\mu \in H_h \setminus H} l^t(\mu) \leq \min_{k \neq h} \max_{\nu \in H_k} l^t(\nu)\}. \quad (5.9)$$

Покажем, что множества T_h , $h = 1, \dots, p$, обладают следующими свойствами:

1) T_h — замкнутый конус, содержащий вектор 0;

$$2) R_q = \bigcup_{h=1}^p T_h;$$

$$3) R_q^0 = \bigcap_{h=1}^p T_h.$$

Действительно, из условия $t \in T_h$ следует, что $\lambda t \in T_h$ для всех $\lambda \geq 0$. Согласно соотношению (5.9) конус T_h является замкнутым.

Справедливость свойства 2) следует из того, что для любого вектора $t \in R_q$ множество $S(t)$ не пусто, поскольку по предположению (см. п. 5.1) граф (Q, U) не содержит контуров.

Справедливость свойства 3) следует непосредственно из определений множеств R_q^0 и T_h .

Отметим, что все полученные в этом параграфе результаты (за исключением верхних оценок радиуса устойчивости) легко переносятся на случай, когда изменяются только длительности некоторого подмножества Q' множества операций Q . Для этого достаточно в приведенных утверждениях вместо множеств $\{\mu\}$ и $\{\nu\}$ рассматривать множества $\{\mu\} \cap Q'$ и $\{\nu\} \cap Q'$ соответственно.

§ 6. Библиографическая справка

Модели обслуживающих систем в виде смешанных графов $G = (Q, U, V)$ предложены В.В. Шкурбой [401], Л.П. Матюшковым и В.С. Танаевым [199] и исследовались в работах [125, 126, 128, 129, 291, 294, 296, 298, 303, 304, 307–309, 342, 352]. Случай $U = \emptyset$ рассматривался в [47, 119].

В зарубежной литературе вместо смешанного графа G используется так называемый дизъюнктивный граф $(Q, U \cup W)$, в котором каждому ребру $[i, j] \in V$ соответствует пара симметричных дуг $(i, j) \in W$ и $(j, i) \in W$. Термин "дизъюнктивный граф" непосредственно связан с операцией выбора одной из двух возможных дуг из каждой такой пары.

Модели обслуживающих систем в виде дизъюнктивных графов предложены Б. Руа и Б. Суссманом [762] и развиты Е. Балашем [449–451], И. Набешимой [706–708], С. Ашауром и Р. Паркером [437, 438], Дж. Чарлтоном и С. Детом [490, 491], Ю. Грабовским [559–561], Б. Лагевеком, Я. Ленстрой и А. Ринной Каном [637, 638, 651, 754]. Эти вопросы рассматривались и в работах [420, 434, 436, 477, 487, 488, 509, 525, 526, 538, 565, 603, 713, 715, 745, 760, 783, 791, 812].

6.1. Представление обслуживающих систем в виде смешанных мультиграфов (см. § 1 этой главы) предложено Ю.Н. Сотсковым и В.С. Танаевым. При построении расписания, допустимого относительно графа (Q, U) (см. п. 1.7), используется алгоритм Р. Флойда [527] и С. Уоршелла [828], а в случае бесконтурного графа (Q, U) (см. п. 1.4) — алгоритм Е. Дейкстры [506].

Условия существования расписания, допустимого относительно смешанного графа, описаны в [291, 517, 791]. Ю.Н. Сотсковым доказана NP -трудность задачи проверки существования расписания, допустимого относительно графа G (см. п. 1.10).

В [2, 234, 261] рассматривались возможности использования обобщенных временных моделей для описания типичных для календарного планирования ситуаций.

6.2. В основу § 4 положены работы Ю.Н. Сотскова [294, 303, 304]. Условия допустимости последовательности преобразований смешанного графа описаны в [303, 304].

Другие формы сетевого представления обслуживающих систем с последовательными и параллельными приборами предложены Е. Балашем [452] и Ю. Грабовским [561]. См. также [538, 696].

6.3. Методы построения активных расписаний (в [147] такие расписания называются квазикompактными) предложены Б. Гиффлером [65, 539], В. Гере [537], Л.П. Матюшковым и В.С. Танаевым [199], Ю.Н. Сотсковым и В.С. Танаевым [295–297, 307, 308], С.А. Канцедалом и О.Н. Малых [127, 129, 131].

В работе [540] показано, что множество компактных (а следовательно, и активных) расписаний содержит хотя бы одно оптимальное при любой целевой функции, не убывающей относительно моментов завершения обслуживания требований. Множества незадерживающих [147] и структурно уплотненных расписаний [131] могут не содержать оптимального расписания.

6.4. Описанные в § 2 алгоритмы построения множества $P(G, H)$ и нахождения его мощности (см. пп. 2.2–2.4) предложены Ю.Н. Сотсковым и В.С. Танаевым [295, 297, 299, 307, 308]. Задача построения множества $P(G)$ (см. п. 2.5) рассматривалась Н.В. Ламбиным и В.С. Танаевым [163], а при $U = \phi$ — Р. Стенли [785]. Случай полного смешанного графа G исследовался В.С. Танаевым [339], С.С. Кислицыным [136], К. Окамурой и Х. Ямашинной [717], Т. Кобаяши [624] и другими авторами. Влияние локальных преобразований графа G на величину $\rho(G)$ (см. п. 2.6, 2.7) исследовалось в [308].

Доказательство NP -полноты задачи проверки выполнимости неравенства $|P(G, H)| > 0$ (см. п. 2.1) проведено Ю.Н. Сотсковым [299]. Е. Аркин и Х. Пападимитриу [426] доказали NP -трудность в сильном смысле зада-

чи проверки существования в G контура отрицательного веса и задачи нахождения пути наименьшего веса в смешанном графе, не содержащем контуров отрицательного веса. В [733] установлена NP -трудность в сильном смысле так называемой задачи о китайском почтальоне на смешанном графе. При $U = \phi$ или $V = \phi$ перечисленные в этом абзаце задачи полиномиально разрешимы [654, 638].

Задача раскраски вершин смешанного графа исследовалась Г.М. Левиным, Ю.Н. Сотсковым и В.С. Танаевым [171, 172, 309]. Хроматический многочлен смешанного графа описан в [309]. Вопросы программной реализации алгоритмов раскраски вершин смешанного графа рассматривались в [137].

6.5. Для решения NP -трудных задач разработано большое количество разнообразных методов направленного перебора вариантов, которые широко используются и в теории расписаний.

Общий формализм методов оптимизации, основанных на идее последовательного конструирования, анализа и отсеивания вариантов, разработан В.С. Михалевичем, Ю.М. Ермольевым, В.В. Шкурбой, Н.З. Шором и др. [213–217]. Детальное описание метода динамического программирования содержится в монографии Р. Беллмана [25]. Значительное развитие конструктивный подход получил в работах Н.Н. Моисеева и его учеников [219–223]. Ряд общих схем решения задач дискретной оптимизации предложен в работах Ю.И. Журавлева [100–102], В.П. Черенина [383], В.Р. Хачатурова [379], В.А. Емеличева и В.И. Комлика [115], И.В. Сергиенко [282], Ю.Ю. Финкельштейна [372], Г.М. Левина и В.С. Танаева [173, 340].

Формализации и теоретическому обоснованию метода ветвей и границ посвящены работы И.В. Романовского [258], А. Лэнд и А. Дойг [640], Т. Ибараки [601–603], Х. Кисэ [623], В. Коглера и К. Стайглица [625], Л. Миттена [692], В. Руа [761], Д. Танга и К. Вонга [814], К. Бейкера [549], А. Джеффриона и Р. Марстена [536], Ю.С. Афонина [15] и ряд других. Обширную библиографию по этим вопросам содержат обзорные работы А.А. Корбута, И.Х. Сигала и Ю.Ю. Финкельштейна [148], Е. Балаша и М. Гиньяр [454], Е. Лоулера и Д. Вуда [649]. (См. также [605].)

Описанные в п. 3.10, 3.11 оценки сложности метода ветвей и границ предложены Т. Ибараки [602].

6.6. Значительное количество работ посвящено использованию метода ветвей и границ при построении оптимальных по быстродействию расписаний. Различные схемы организации процесса ветвления (разбиения множества допустимых расписаний на подмножества) рассматривались Е. Балашем [450], Г. Бруксом и С. Уайтом [478], Л. Немети [714], Ю.Н. Сотсковым [291], И. Набешимой [706, 707], С. Ашауром и С. Хиремасом [435], Дж. Чарлтоном и С. Детом [491], М. Флорианом, П. Трепантом и Г. Мак-Махоном [526].

В работах [54, 126, 193, 199, 291–293, 300, 420, 426, 436, 450, 490, 559–561, 706–708, 713, 783, 791] оптимальное расписание ищется в классе активных расписаний, а в работах [125, 477, 478, 525, 526, 686, 714] – в классе компактных расписаний.

Для вычисления оценки снизу величины $\bar{t}_{\max}(s^*)$ обычно решается та или иная релаксированная задача. П. Братли, М. Флориан и П. Робайар [477] предложили использовать для этой цели задачу минимизации мак-

симального временного смещения обслуживания частично упорядоченного множества N требований одним прибором, если для каждого требования $i \in N$ задан момент d_i его поступления и директивный срок D_i . Эта задача является NP -трудной в сильном смысле [657]. Обзор методов ее решения приведен в [637, 686]. Б. Лагевер, Я. Ленстра и А. Ринной Кан [638] показали, что почти все известные в литературе оценки снизу величины $\bar{t}_{\max}(s^*)$ могут быть получены в результате решения указанной задачи или ее частных случаев. Так, в работах [291, 434–437, 491, 768] значения $d_i = D_i = 0$ для всех $i \in N$; в работах [291, 436, 478, 526] значения $D_i = 0, i \in N$; в [767] значения $d_i = 0, i \in N$; в работах [291, 300, 434, 436, 437, 490, 491, 571, 706, 714, 768, 791] предполагалось, что прибор может обслуживать все требования множества N одновременно, а в работах [291, 477, 686] — что множество требований N неупорядочено. Последняя из перечисленных релаксированных задач является NP -трудной в сильном смысле [351, 657], для остальных известны полиномиальные алгоритмы решения [351, 603, 686]. В [291] для получения оценок величины $\bar{t}_{\max}(s^*)$ предложено использовать методы оптимизации функций на перестановках. Эти методы развиты в работах [210, 264–268, 326, 329, 343].

При некоторых дополнительных ограничениях удастся получить более точные оценки величины $\bar{t}_{\max}(s^*)$. В работах [426, 477, 686] существенно использовалось предположение о целочисленности длительностей операций. В [376, 429, 434, 435, 437, 438, 478] предполагалось, что каждое требование обслуживается каждым прибором точно один раз.

6.7. Разнообразные варианты метода ветвей и границ разработаны для решения задачи минимизации общего времени обслуживания требований в случае одинаковых маршрутов, т.е. для систем поточного типа. Как правило, рассматривается ситуация, когда множество требований поступает в систему одновременно и все приборы обслуживают требования в одной и той же последовательности. В этом случае расписание однозначно определяется перестановкой требований. Принципиальная схема ветвей и границ для решения экстремальных задач на множестве перестановок предложена В. Коглером и К. Стайглицем [143, 625]. Большинство авторов рассматривают правило ветвления, согласно которому на основании некоторой перестановки σ , $\{\sigma\} \subseteq N$, строятся все перестановки вида (σ, i) , $i \in N \setminus \{\sigma\}$. Нижние оценки величины $\bar{t}_{\max}(\pi)$, где $\pi = (\sigma, \dots)$ для заданной перестановки σ , $\{\sigma\} \subseteq N$, предложены в [11, 361, 362, 384, 421, 431, 439, 466, 479, 574, 576, 580, 604, 667, 685, 704, 709] (см. также [45, 352, 446, 651, 754]). Вопросы вычисления глобальных нижних оценок $(\{\sigma\} = \emptyset)$ рассматривались в [10, 457, 458]. Б. Лагевер, Я. Ленстра, А. Ринной Кан [639, 651, 754] и В. Шварц [805] описали два различных подхода к генерированию нижних оценок общего времени обслуживания требований, позволяющих получать большинство ранее известных и некоторые новые нижние оценки. Метод ветвей и границ, основанный на ином правиле ветвления, предложен К. Потсом [742] (см. также [772]).

В целях сокращения объема вычислений метода ветвей и границ в ряде работ используются те или иные отношения доминирования, которые задаются условиями, достаточными для того, чтобы требование $i \in N$ пред-

уществовало в оптимальной перестановке требованию $k \in N, k \neq i$. Выявлению такого рода условий посвящены работы [448, 508, 578, 585, 684, 778, 779, 797, 798, 801]. Отметим, что условия из [448, 508, 778] не корректны (контрпримеры приведены в [685, 617, 797] соответственно). Кроме того, установлено, что остальные условия, по существу, эквивалентны [651, 754, 798, 801], и если они выполняются для некоторой пары требований $i \in N$ и $k \in N, i \neq k$, то справедливы соотношения $t_{i1} \leq t_{iL}$ либо $t_{i, L-1} \geq t_{iM}, L = 2, \dots, M$. К. Бейкер [448] и Дж. Гупта [582] провели анализ вероятности выполнения указанных соотношений. В [590, 804] предприняты попытки задания отношений доминирования, когда указанные соотношения не имеют места и значение $M = 3$.

Другие методы сокращения объема вычислений метода ветвей и границ исследуются в [11, 444, 563, 726].

К числу методов решения данной задачи, использующих идеи последовательного анализа вариантов, следует отнести и метод лексикографического поиска, предложенный Дж. Гуптой [573] и получивший развитие в [494, 587].

Сведения о результатах наиболее крупных вычислительных экспериментов с различными схемами ветвей и границ приведены в [430, 447, 639, 742]. Там же содержатся некоторые рекомендации по выбору правил ветвления и вычисления оценок.

6.8. Для решения задач с критериями оптимальности, отличными от минимизации $\bar{t}_{\max}(s)$, в случае одинаковых маршрутов обслуживания требований предложены разнообразные схемы ветвей и границ. Задача минимизации $L_{\max}(s)$ рассматривалась У. Таусендом [818], если $d_i = 0, i \in N$, а также Ю. Грабовским [562], если $d \geq 0, i \in N$ (см. также [564, 566]). Задача минимизации линейных и экспоненциальных функций штрафа рассматривалась в [819], а задача минимизации максимального штрафа произвольного вида — в [820, 821] (см. также [416]). Различные схемы ветвей и границ для минимизации суммарного времени обслуживания требований предложены в [143, 443, 604, 626] для случая $M = 2$, в [456, 711] — для произвольного числа приборов M и в [503] — для произвольного M при условии обслуживания требований без задержек. К. Калра, С. Бенсел и П. Багга [615] предложили метод ветвей и границ для решения задачи минимизации суммарного взвешенного времени обслуживания $\sum_{i \in N} \omega_i \bar{t}_i$ при

условии, что все $\bar{t}_i \leq D_i$. В [143, 626, 711] обсуждаются возможности использования методов ветвей и границ в качестве ϵ -приближенного алгоритма. Отношения доминирования для задачи минимизации суммарного времени обслуживания описаны в [582].

6.9. Методы типа ветвей и границ применялись и для оптимального обслуживания требований в системах с различными маршрутами. Так, в [104, 105, 478, 716] рассматривалась задача минимизации суммарного запаздывания при обслуживании требований, в [522] — задача минимизации суммарного штрафа за обслуживание требований, в [150, 538] — задача минимизации стоимости использования приборов, в [818] — задача минимизации максимального запаздывания при обслуживании требований, в [572] — задача минимизации общей стоимости переналадок приборов. См. также [22, 29, 107, 155, 244, 252, 298, 396, 398, 410, 517, 666, 679, 786].

Задачи построения оптимальных расписаний при произвольных неубывающих по каждому из аргументов целевых функциях рассматривались И. Набешимой [706–708] и Ю.Н. Сотсковым [294, 302–304].

Б. Суссманом [791], Ю.Н. Сотсковым [298], Дж. Чарлтоном и С. Детом [490] рассматривалась задача построения оптимального по быстродействию расписания, допустимого относительно смешанного графа $G = (Q, U, V)$, где (Q, U) — обобщенная временная модель.

6.10. Использованию метода ветвей и границ для получения приближенных решений задач теории расписаний посвящены работы [245, 293, 706, 708]. Полиномиальные ϵ -приближенные алгоритмы предложены в [139–141, 181, 523].

В [298] предложен метод минимизации максимального штрафа за обслуживание требований, существенно использующий алгоритм выделения контуров смешанного графа (см. [254]) и алгоритм проверки выполнимости конъюнктивной нормальной формы (см. [107, 500, 518]). Использование линейного от числа дизъюнкций алгоритма решения задачи 2-выполнимости из [518] и алгоритма из [307] позволяет выделить некоторые полиномиально разрешимые случаи указанной задачи теории расписаний [301, 306].

Е. Балашем [453] исследовался многогранник расписаний, допустимых относительно смешанного графа.

В.Б. Алюшкевичем и Ю.Н. Сотсковым [6, 305] рассматривались вопросы устойчивости множества оптимальных по быстродействию расписаний, допустимых относительно смешанного графа. При этом использовался подход, предложенный В.К. Леонтьевым и Э.Н. Гордеевым [78, 79, 182, 183]. Необходимые и достаточные условия, при выполнении которых радиус устойчивости оптимального графа равен нулю (см. п. 5.7 гл. 4) и бесконечности (п. 5.3), получены Ю.Н. Сотсковым. Формула для вычисления радиуса устойчивости оптимального графа (п. 5.4) получена В.Б. Алюшкевичем и Ю.Н. Сотсковым. Исследование области устойчивости оптимального графа (п. 5.10) и радиуса устойчивости множества оптимальных графов (п. 5.9) проведено Ю.Н. Сотсковым.

Вопросы устойчивости оптимальных расписаний при критериях оптимальности, отличных от $\bar{t}_{\max}$, также рассматривались в работах В.Б. Алюшкевича и Ю.Н. Сотскова.

6.11. Ситуации, в которых для выполнения отдельной операции требуется несколько приборов, рассматривались А.И. Куксой, Ю.П. Лаптыным, Б.Е. Поляченко [159–162], Я. Блажевичем [468, 469] и другими авторами [150, 245, 499, 505, 530, 531, 611, 650, 666, 696, 707, 708, 786, 787]. Обзоры результатов по решению таких задач содержатся в монографии В.С. Михалевича и А.И. Куксы [215], а также в работах М. Гонсалеса [548], Я. Блажевича, Я. Ленстры и А. Ринной Кана [470].

Ряд других обобщений задач теории расписаний рассматривался в работах [29, 74, 80, 122, 124, 151, 206, 218, 239, 253, 337, 338, 344, 364, 366, 389, 391, 396, 398, 407, 408, 410, 463, 476, 575, 577]. Многокритериальные задачи теории расписаний исследовались в [31, 109, 140, 193, 357–359, 474].

6.12. Много публикаций посвящено прикладным аспектам теории расписаний, в частности рассмотрению вопросов календарного планирования крупномасштабных проектов [63, 64, 240], химического производства

[60, 124, 405, 406], строительного производства [112, 115, 261, 292], вычислительных центров [22, 23, 76, 77, 104, 105, 262, 548], гибких производственных систем [103, 142, 196, 235, 243, 244, 353, 385, 387]. См. также [1, 21, 52, 142, 211, 212, 231, 241, 242, 269, 280, 287, 288, 346, 404, 575, 577, 724, 773, 831].

Описания программных реализаций методов решения отдельных задач теории расписаний приведены в [31, 142, 144, 169, 285, 310, 387]. В [635, 636] сообщается о программе, позволяющей вести обработку и накопление результатов по исследованию сложности задач теории расписаний.

В [348, 349] описан пакет программ РУПОР для решения широкого круга задач построения оптимальных расписаний обслуживания требований в одностадийных и многостадийных системах. Функциональные модули пакета написаны на алгоритмическом языке ФОРТРАН-IV, управляющая часть — на ассемблере. Пакет программ может эксплуатироваться в интерактивном режиме, имеет развитые информационно-справочные и обучающие средства.

1. Автоматизированное проектирование и производство в машиностроении / Под ред. Ю.М. Соломенцева и В.Г. Митрофанова. — М.: Машиностроение, 1986. — 256 с.
2. Адельсон-Вельский Г.М. О некоторых вопросах сетевого планирования // Исследования по дискрет. мат. — М.: Наука, 1973. — С. 105–134.
3. Адельсон-Вельский Г.М., Диниц Е.А., Карзанов А.В. Потокковые алгоритмы. — М.: Наука, 1975. — 120 с.
4. Айзенштат В.С. Совмещение примитивных циклических процессов // Докл. АН БССР. — 1963. — Т. 7, № 3. — С. 148–151.
5. Айзенштат В.С. Многооператорные циклические процессы // Докл. АН БССР. — 1963. — Т. 7, № 4. — С. 224–227.
6. Аллюшкевич В.Б., Сотсков Ю.Н. Исследование устойчивости оптимальных по быстродействию расписаний. — Препринт/ИТК АН БССР. — Минск, 1987. — № 9. — 20 с.
7. Ангуладзе М.Х. Об использовании метода статистического поиска с применением правил предпочтения в задачах составления расписаний // Сообщ. АН ГССР. — 1972. — Т. 65, № 2. — С. 417–420.
8. Андронов А.М., Хижняк А.П. Математические методы планирования и управления производственно-хозяйственной деятельностью предприятий гражданской авиации. — М.: Транспорт, 1977. — 215 с.
9. Аронович А.Б. О выборе оптимальных комбинаций локальных правил календарного планирования // Экон. и мат. методы. — 1970. — Т. 6, № 4. — С. 548–557.
10. Артамонов И.М. Алгоритмы определения последовательности обработки деталей, имеющих одинаковый технологический маршрут // Мат. исследования. — Кишинев, 1979. — Вып. 52. — С. 3–8.
11. Артамонов И.М. Об одном алгоритме решения задачи Беллмана — Джонсона // Модели и алгоритмы решения задач планирования и управления. — Кишинев, 1982. — С. 3–10.
12. Асратян А.С. О существовании реберной раскраски специального вида в двудольном графе // Сб. по мат. кибернетике. — М.: ВЦ АН СССР, 1977. — Вып. 2. — С. 45–57.
13. Асратян А.С. О построении реберной раскраски специального вида в двудольном графе и ее применении в задачах теории расписаний // Вестник МГУ. Сер. XV, выч. мат. и киберн. — 1978. — № 1. — С. 74–81.
14. Афанасьев В.А., Карелин В.В. Определение возможного минимума продолжительности выполнения комплекса работ // Кибернетика. — 1974. — № 1. — С. 89–90.
15. Афонин Ю.С. Блочный метод ветвей и границ // Автомат. и телемех. — 1986. — № 8. — С. 98–108.
16. Бабушкин А.И., Башта А.Л., Белов И.С. Построение календарного плана для задачи встречных маршрутов // Кибернетика. — 1977. — № 4. — С. 130–135.
17. Бабушкин А.И., Башта А.Л., Белов И.С. Построение календарных планов для многомаршрутной задачи трех станков // Автомат. и телемех. — 1976. — № 7. — С. 154–158.

18. *Бабушкин А.И., Башта А.Л., Белов И.С., Душин Б.И.* Минимизация цикла работы поточной линии // Автомат. и телемех. — 1975. — № 6. — С. 161–167.
19. *Бабушкин А.И., Белов И.С.* Оптимальные комбинации приоритетных правил составления расписаний // Автомат. и телемех. — 1986. — № 5. — С. 82–90.
20. *Баркан С.А.* Об общей модели двухстадийного производства // Изв. АН БССР. Сер. физ.-техн. наук. — 1969. — № 2. — С. 42–47.
21. *Баркан С.А., Танаев В.С.* О построении расписаний учебных занятий // Изв. АН БССР. Сер. физ.-мат. наук. — 1970. — № 1. — С. 76–82.
22. *Барский А.Б.* Две задачи оптимизации использования неоднородных вычислительных систем // Изв. АН СССР. Техн. кибернетика. — 1971. — № 4. — С. 119–125.
23. *Барский А.Б.* Планирование параллельных вычислительных процессов. — М.: Машиностроение, 1980. — 192 с.
24. *Белецкий С.А., Данильченко А.М., Панишев А.В.* Перестановочный прием в задаче трех станков // Кибернетика. — 1985. — № 2. — С. 106–109.
25. *Беллман Р.* Динамическое программирование. — М.: ИЛ, 1960. — 400 с.
26. *Белов И.С., Столин Я.Н.* Алгоритм в одномаршрутной задаче календарного планирования // Математическая экономика и функциональный анализ. — М.: Наука, 1974. — С. 248–257.
27. *Блох А.Ш., Танаев В.С.* Многооператорные процессы // Изв. АН БССР. Сер. физ.-мат. наук. — 1966 — № 2. — С. 5–11.
28. *Болун И.Ф.* К задаче упорядочения работ в многофазных системах обслуживания // Модели и алгоритмы АСУ. — Кишинев, 1986. — С. 29–58.
29. *Большаков В.А., Уздемир А.П., Шмелев В.В.* Задачи планирования дискретного (штучного) производства и численные методы их решения. III // Автомат. и телемех. — 1976. — № 1. — С. 146–156.
30. *Бондаренко О.А.* Эвристические подходы к решению задач календарного планирования // Автоматиз. системы упр. предприятиями. Труды семинара. — 1968. — Вып. 1. — С. 47–65.
31. *Борзенко В.И., Трахтенгерц Э.А., Шершанов В.М.* Аппроксимационная многокритериальная диспетчеризация в вычислительных системах // Автомат. и телемех. — 1986. — № 9. — С. 152–162.
32. *Бородич С.А.* О трудноразрешимости задач минимизации числа запаздывающих работ в двухстадийной системе обслуживания с нефиксированными маршрутами // Сложность и методы решения задач оптимизации. — Минск: ИТК АН БССР, 1984. — С. 4–8.
33. *Бородич С.А.* Об одной задаче минимизации максимального временного смещения в двухстадийной системе обслуживания с нефиксированными маршрутами // Изв. АН БССР. Сер. физ.-мат. наук. — 1985. — № 5. — С. 109.
34. *Бородич С.А.* К вопросу сложности построения оптимальных по быстродействию расписаний для одного класса обслуживающих систем с операторами переноса // Изв. АН БССР. Сер. физ.-мат. наук. — 1986. — № 4. — С. 116–117.
35. *Бородич С.А.* О параметрической задаче для двухстадийной обслуживающей системы с нефиксированными маршрутами // Применение мат. методов и вычисл. техники при решении народно-хозяйственных задач: Тез. докл. респ. науч.-техн. совещ. — Минск, 1986. — С. 14–15.
36. *Бородич С.А., Струсович В.А.* Обслуживание упорядоченного множества требований в системе с нефиксированными маршрутами. — Препринт/Ин-т математики АН БССР. — Минск, 1986. — № 7 (243). — 24 с.
37. *Бородич С.А., Струсович В.А.* Расписания обслуживания частично упорядоченного множества требований для одного класса детерминированных систем // Изв. АН БССР. Сер. физ.-мат. наук. — 1986. — № 3. — С. 19–22.
38. *Бородич С.А., Тузиков А.В.* О сложности построения оптимальных по быстродействию расписаний для некоторых двухстадийных систем обслуживания с нефиксированными маршрутами // Методы, алгоритмы и программы решения экстремальных задач. — Минск: ИТК АН БССР, 1985. — С. 76–85.
39. *Бродский И.Л.* Оптимальная реакция на единичное возмущение расписания // Экон. и матем. методы. — 1976, № 6. — С. 1206–1210.
40. *Букина В.Н., Беккер Р.* О решении задачи Джонсона в пространстве перестановок // Исследование операций и АСУ. — Киев: КГУ, 1977. — Вып. 10. — С. 116–122.

41. Бурдюк В.Я. О задаче m станков ($m > 2$) // Кибернетика. — 1969. — № 3. — С. 74–76.
42. Бурдюк В.Я. Стохастическая задача двух станков // Кибернетика. — 1969. — № 5. — С. 110–114.
43. Бурдюк В.Я. О выборе последовательности загрузки станков // Экон. и мат. методы. — 1970. — Т. 6, № 1. — С. 112–116.
44. Бурдюк В.Я., Рева В.Н. Об одном методе оптимизации функционала от перестановок при наличии ограничений // Кибернетика. — 1980. — № 1. — С. 99–103.
45. Бурдюк В.Я., Шкурба В.В. Теория расписаний. Задачи и методы решений. 1. // Кибернетика. — 1971. — № 1. — С. 89–102.
46. Бурдюк Т.А. Оптимальное упорядочение работ для станков равной производительности // Изв. АН СССР. Техн. кибернетика. — 1972. — № 1. — С. 32–36.
47. Бурков В.Н. Распределение ресурсов как задача оптимального быстрого действия // Автомат. и телемех. — 1966. — Т. 27, вып. 7. — С. 119–129.
48. Бурков В.Н., Ланда В.Д., Ловецкий С.Е., Тейман А.И., Чернышев В.Н. Сетевые модели и задачи управления. — М.: Сов. радио. — 1967. — 144 с.
49. Бурков В.Н., Ловецкий С.Е. Методы решения экстремальных комбинаторных задач (обзор) // Изв. АН СССР. Техн. кибернетика. — 1968. — № 4. — С. 82–93.
50. Бурков В.Н., Ловецкий С.Е. Методы решения экстремальных задач комбинаторного типа (обзор) // Автомат. и телемех. — 1968. — № 11. — С. 68–93.
51. Василенко С.И. Об одной задаче теории расписаний // ЖВМ и МФ. — 1983. — Т. 23, № 3. — С. 713–718.
52. Васильев Г.А. Алгоритм нахождения допустимого решения в задачах календарного планирования горных работ // Вопросы кибернетики. Ташкент: АН УзССР, 1978. — Вып. 100. — С. 13–21.
53. Визинг В.Г. Хроматический класс мультиграфа // Кибернетика. — 1965. — № 3. — С. 29–39.
54. Вишняков В.А., Герман О.В. Планирование мультипрограммной работы в параллельных вычислительных системах // Изв. АН СССР. Техн. кибернетика. — 1984. — № 1. — С. 206–213.
55. Владимиров В.Н., Егоров С.В. Декомпозиционный метод решения задачи Беллмана — Джонсона с возмущениями // Изв. АН СССР. Техн. кибернетика. — 1982. — № 2. — С. 86–91.
56. Власюк В.А. Задачи оптимального расписания при параллельно-последовательном соединении машин // Изв. АН СССР. Техн. кибернетика. — 1966. — № 4. — С. 67–80.
57. Власюк В.А. Оптимальное расписание обработки на трех последовательных машинах // Изв. АН СССР. Техн. кибернетика. — 1967. — № 4. — С. 79–84.
58. Волошин В.В. Об одной задаче производственного планирования // Труды Ленингр. политехн. ин-та им. М.И. Калинина. — 1967. — Вып. 291. — С. 43–46.
59. Выгнан П.К., Кудин Б.И. Алгоритм построения календарного графика работы оборудования на участках с разнонаправленными маршрутами обработки // Применение мат. в экон. — Л.: ЛГУ, 1969. — С. 68–78.
60. Ганеев В.В., Соболев О.С., Софиев А.Э., Шлепаков П.А. Задача расчета оптимального плана-графика производства полиэтилена // Оперативное планирование и управление производством. — М.: Энергоатомиздат, 1985. — С. 23–27.
61. Гимади Э.Х., Глебов Н.И., Перепелица В.А. Исследования по теории расписаний // Управляемые системы. — 1974. — Вып. 12. — С. 3–10.
62. Гимади Э.Х., Глебов Н.И., Перепелица В.А. Алгоритмы с оценками для задач дискретной оптимизации // Проблемы кибернетики. — Вып. 31. — М.: Наука, 1975. — С. 35–42.
63. Гимади Э.Х., Пузынина Н.М. Задача календарного планирования крупномасштабного проекта в условиях ограниченных ресурсов: опыт построения математического обеспечения // Управляемые системы. — 1983. — Вып. 23. — С. 24–32.
64. Гимади Э.Х., Пузынина Н.М., Севастьянов С.В. О некоторых экстремальных задачах реализации крупных проектов типа БАМ // Экон. и мат. методы. — 1979. — Т. 15, вып. 5. — С. 1017–1020.
65. Гиффлер Б. Алгебры календарного планирования и их применение при формулировании моделей общих систем // Календарное планирование. — М.: Прогресс, 1966. — С. 62–83.

66. Гиффлер Б., Томпсон Д., Ван-Нессе В. Опыт вычислений с применением линейного алгоритма и алгоритма Монте-Карло для решения задач календарного планирования на производстве // Календарное планирование. — М.: Прогресс, 1966. — С. 42–61.
67. Глебов Н.И. Алгоритм составления расписания для двух работ // Управляемые системы. — 1968. — Вып. 1. — С. 14–20.
68. Глебов Н.И. О верхней оценке длины оптимального расписания для двух работ // Пробл. кибернетики. — М.: Наука, 1968. — Вып. 20. — С. 225–229.
69. Глебов Н.И. Некоторые случаи сводимости m -станочной задачи Джонсона к задаче с двумя станками // Управляемые системы. — 1978. — Вып. 17. — С. 46–51.
70. Глебов Н.И., Перепелица В.А. О нижней и верхней оценках для одной задачи теории расписаний // Исслед. по кибернетике. — М.: Сов. радио, 1970. — С. 11–17.
71. Глебский Ю.В., Шевченко В.Н. О составлении оптимального плана работ // Научн. труды Моск. инж.-экон. ин-та. — 1964. — Вып. 21. — С. 318–324.
72. Глебский Ю.В., Шевченко В.Н. О составлении оптимального расписания работы на n станках // Труды по вопр. примен. ЭВМ в нар. хозяйстве. — Горький. — 1964. — С. 31–34.
73. Голенко Д.И., Гофман Э.Ф., Тарнопольский Ю.Я., Столин Я.Н. Асимптотические точные оценки для одномаршрутных задач теории расписаний // Численные методы нелинейного программирования: Материалы I Всесоюзн. семинара. — Киев, 1976. — С. 180–186.
74. Голенко Д.И., Тарнопольский Ю.Я. Оптимизация календарных планов методами направленного поиска // Кибернетика. — 1970. — № 6. — С. 138–144.
75. Голенко Д.И., Фишбейн С.М. Некоторые методы решения одномаршрутных задач календарного планирования // Труды Ленингр. инж.-экон. ин-та им. П. Тольятти. — 1972. — Вып. 94. — С. 51–66.
76. Головкин Б.А. Классификация методов диспетчеризации работы многопроцессорных и многомашинных вычислительных систем // УС и М. — 1982. — № 3. — С. 3–11.
77. Головкин Б.А. Расчет характеристик и планирование параллельных вычислительных процессов. — М.: Радио и связь, 1983. — 272 с.
78. Гордеев Э.Н., Леонтьев В.К. Устойчивость в задачах на узкие места // ЖВМ и МФ. — 1980. — Т. 20, № 4. — С. 1071–1075.
79. Гордеев Э.Н., Леонтьев В.К. Траекторные параметрические задачи // ЖВМ и МФ. — 1984. — Т. 24, № 1. — С. 37–46.
80. Гордон В.С., Танаев В.С. О минимаксных задачах теории расписаний с одним прибором // Изв. АН БССР. Сер. физ.-мат. наук. — 1983. — № 3. — С. 3–9.
81. Гордон В.С., Шафранский Я.М. Об одном классе задач теории расписаний с частично упорядоченными требованиями // IV Всесоюзн. конф. по проблемам теоретической кибернетики: Тез. докл. — Новосибирск, 1977. — С. 101–103.
82. Гордон В.С., Шафранский Я.М. Оптимальное упорядочение при последовательно-параллельных ограничениях предшествования // Докл. АН БССР. — 1978. — Т. 22, № 3. — С. 244–247.
83. Гордон В.С., Шафранский Я.М. Об оптимальном упорядочении при последовательно-параллельных ограничениях предшествования // Изв. АН БССР. Сер. физ.-мат. наук. — 1978. — № 5. — С. 135.
84. Гофман Э.Ф., Кеслер С.Ш. Алгоритмизация решения задач календарного планирования // Труды Ленингр. инж.-экон. ин-та им. П. Тольятти. — 1976. — Вып. 120. — С. 12–33.
85. Григер В.А. О точном алгоритме составления оптимального расписания многооператорной поточной линии // Вопросы радиоэлектроники. Сер. ЭВТ. — 1983. — Вып. 12. — С. 58–63.
86. Григер В.А., Нуриев Н.К., Гинзбург М.М. Построение расписаний для многопроцессных автооператорных поточных линий гальванического производства // Вопросы радиоэлектроники. Сер. ЭВТ. — 1985. — Вып. 12. — С. 59–64.
87. Гринберг В.С., Севастьянов С.В. О величине константы Штейнница // Функцион. анализ и его прил. — 1980. — Т. 14, № 2. — С. 56–57.
88. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи. — М.: Мир, 1982. — 416 с.

89. Данильченко А.М., Левченко С.Н., Панишев А.В. Приближенный алгоритм решения задачи трех станков // Автомат. и телемех. — 1985. — № 7. — С. 133–139.
90. Данильченко А.М., Панишев А.В. О частных случаях задачи трех станков // Изв. АН СССР. Техн. кибернетика. — 1986. — № 4. — С. 197.
91. Демин В.К., Чеботарев А.С. Оптимальное обслуживание детерминированного потока требований. I, II // Изв. АН СССР. Техн. кибернетика. — 1976. — № 4. — С. 193–199; 1976. — № 5. — С. 65–70.
92. Джексон Д.Р. Очереди с динамическим правилом приоритета // Календарное планирование. — М.: Прогресс, 1966. — С. 357–377.
93. Доцатов В.В., Тогер А.В. О некоторых обобщениях одмаршрутной задачи календарного планирования // Машинная обработка информации. — Киев, 1970. — Вып. 29. — С. 92–98.
94. Драшлин Е.Х., Якимов Р.М. Об одном методе определения оптимальной последовательности непрерывного процесса обработки // Науч. труды. — Пермь: Перм. политехн. ин-т, 1966. — Вып. 21. — С. 55–59.
95. Драшлин Е.Х., Якимов Р.М. Анализ свойств оптимального варианта последовательности обработки изделий // Науч. труды. — Пермь: Перм. политехн. ин-т, 1966. — Вып. 21. — С. 59–61.
96. Душин Б.И. Замечание к алгоритму в одмаршрутной задаче Джонсона // Кибернетика. — 1980. — № 2. — С. 129–131.
97. Емеличев В.А., Комлик В.И. Метод построения последовательности планов для решения задач дискретной оптимизации. — М.: Наука, 1981. — 207 с.
98. Емеличев В.А., Супруненко Д.А., Танаев В.С. О работах белорусских математиков в области дискретной оптимизации // Изв. АН ССР. Техн. кибернетика. — 1982. — № 6. — С. 25–45.
99. Ефетова К.Ф., Подчасова Т.П., Португал В.М., Тринчук Б.Е. Планирование производства в условиях АСУ: Справочник. — Киев: Техніка, 1984. — 135 с.
100. Журавлев Ю.И. Оценка сложности алгоритмов построения минимальных дизъюнктивных нормальных форм для функций алгебры логики // Дискретный анализ. — Новосибирск, 1964. — Вып. 3. — С. 41–77.
101. Журавлев Ю.И. Оценка сложности локальных алгоритмов для некоторых экстремальных задач на конечных множествах // ДАН СССР. — 1964. — Т. 158, № 5. — С. 1018–1021.
102. Журавлев Ю.И. Локальные алгоритмы вычисления информации. I, II // Кибернетика. — 1965. — № 1. — С. 12–19; 1966. — № 2. — С. 1–11.
103. Жутая С.Н., Соколицы А.С. Распределение производственной программы по плановым периодам на участках, оснащенных станками с ЧПУ // Резервы повышения экономической эффективности машиностроительного производства. — Саратов: СПИ, 1982. — С. 84–87.
104. Зайченко Ю.П., Вишаль Д.М., Клитовченко О.А. Организации планирования работ на ВЦКП с учетом всех стадий технологического процесса // Вестник Киевского политехн. ин-та. Техн. кибернетика. — 1980. — Вып. 4. — С. 72–78.
105. Зайченко Ю.П., Клитовченко О.А. Исследование эвристических алгоритмов планирования работы ВЦ при многостадийном обслуживании // Вестник Киевского политехн. ин-та. Техн. кибернетика. — 1982. — Вып. 6. — С. 29–83.
106. Зак Ю.А. Некоторые свойства задач теории расписаний // Автомат. и телемех. — 1978. — № 1. — С. 123–132.
107. Заковский А.Д. Алгоритмы синтеза дискретных автоматов. — М.: Наука, 1971. — С. 213–268.
108. Зимин И.Н., Иванилов Ю.П. Решение задач сетевого планирования сведением их к задачам оптимального управления // ЖВМ и МФ. — 1971. — Т. 11, № 3. — С. 632–641.
109. Зиндер Я.А., Шкурба В.В. Эффективные итеративные алгоритмы теории расписаний // Кибернетика. — 1985. — № 1. — С. 72–75.
110. Золотухин В.Ф. Об исключении заведомо неоптимальных решений задачи Беллмана — Джонсона // Изв. АН СССР. Техн. кибернетика. — 1979. — № 1. — С. 44–51.
111. Зыков А.А. Теория конечных графов. — Новосибирск: Наука, 1969. — 544 с.
112. Ильницкий А.Л. Об одном подходе к решению задачи календарного планирования в строительстве // Исследование операций и АСУ. — Киев: КГУ, 1976. — Вып. 8. — С. 25–31.

113. *Ильницкий А.Л.* Двойственность в целочисленном программировании при решении задач календарного планирования // Исследование операций и АСУ. — Киев: КГУ, 1978. — Вып. 11. — С. 75–81.
114. *Ильницкий А.Л.* Оценка целевой функции одной дискретной задачи теории расписаний // Исследование операций и АСУ. — Киев: КГУ, 1979. — Вып. 13. — С. 54–58.
115. *Ильницкий А.Л.* Универсальность и условия существования решений одного класса задач теории расписаний // Исследование операций и АСУ. — Киев: КГУ, 1980. — Вып. 15. — С. 38–46.
116. *Инукаи Т.* Замечания к статье "Анализ коммутационной матрицы для системы ПБ/МДВУ" // ТИИЭР. — 1978. — Т. 66, № 12. — С. 94–95.
117. *Иоффе Э.Г.* Об одном алгоритме расчета календарных расписаний для производственного участка // Экон. и мат. методы. — 1971. — Т. 7, № 6. — С. 876–882.
118. *Иоффе Э.Г.* Алгоритм для определения всех оптимальных расписаний в двух-операционной задаче Джонсона // Автомат. и телемех. — 1973. — № 7. — С. 95–101.
119. *Ириков В.А.* Некоторые задачи упорядочения // Изв. АН СССР. Техн. кибернетика. — 1970. — № 4. — С. 38–42.
120. *Ито Я., Урано Е., Муратани Т., Ямагути М.* Анализ коммутационной матрицы для системы ПБ/МДВУ // ТИИЭР. — 1977. — Т. 65, № 3. — С. 152–162.
121. *Кадец М.И.* Об одном свойстве векторных ломаных в n -мерном пространстве // УМН. — 1953. — Т. 8, № 1. — С. 139–143.
122. *Казанцев Э.Н.* Математическая модель графика выпуска изделий, удовлетворяющего требованию комплектации сборки // Изв. АН БССР. Сер. физ.-мат. наук. — 1968. — № 1. — С. 54–60.
123. Календарное планирование / Сост. Мут Дж. Ф., Томпсон Дж.Л. — М.: Прогресс, 1966. — 466 с.
124. *Калужный Б.И., Солдатов В.С.* Модель диспетчеризации дискретных производственных процессов // Изв. АН СССР. Техн. кибернетика. — 1976. — № 6. — С. 21–26.
125. *Канцедал С.А.* Алгоритм сокращения поиска решений в задаче теории расписаний сетевой структуры // Автомат. и телемех. — 1982. — № 4. — С. 72–77.
126. *Канцедал С.А.* Декомпозиционный подход к решению задач теории расписаний большой размерности // Автомат. и телемех. — 1983. — № 10. — С. 144–151.
127. *Канцедал С.А., Малых О.Н.* О построении допустимых планов в общей задаче календарного планирования // Автомат. системы управл. и приборы автомат. — Харьков: Вища школа, 1978. — Вып. 45. — С. 81–86.
128. *Канцедал С.А., Малых О.Н.* Математические модели календарного планирования производства и их использование. Сообщение. I, II // Автомат. системы управл. и приборы автомат. — Харьков: Вища школа, 1978. — Вып. 47. — С. 29–35; С. 36–41.
129. *Канцедал С.А., Малых О.Н.* Классы расписаний и алгоритмы их построения // Автомат. системы управл. и приборы автомат. — Харьков: Вища школа, 1979. — Вып. 51. — С. 69–75.
130. *Канцедал С.А., Малых О.Н.* Случайный поиск в задачах теории расписаний // Автомат. системы управл. и приборы автомат. — Харьков: Вища школа, 1979. — Вып. 51. — С. 75–81.
131. *Канцедал С.А., Малых О.Н.* О классах расписаний // Кибернетика. — 1981. — № 6. — С. 66–74.
132. *Карзанов А.В.* Нахождение максимального потока в сети методом предпотоков // ДАН СССР. — 1974. — Т. 215, № 1. — С. 49–52.
133. *Карп Р.М.* Сводимость комбинаторных проблем // Кибернет. сб. Новая серия. — М.: Мир, 1975. — Вып. 12. — С. 16–38.
134. *Катаев В.А.* Оптимальная очередь обработки деталей на произвольном количестве станков (теория расписаний) // Науч. труды НИИ управляющих машин и систем. — Пермь. — 1967. — Вып. 1. — С. 57–68.
135. *Киладзе Ю.В., Кварацхелия О.И.* Модель локальной оптимизации для одномаршрутной задачи календарного планирования // Труды Тбилисского гос. ун-та. — 1980. — Вып. 212. — С. 178–191.
136. *Кислицын С.С.* Конечные частично упорядоченные множества и соответствующие им множества перестановок // Мат. заметки. — 1968. — Т. 4, вып. 5. — С. 511–518.

137. *Климова Л.И., Сотсков Ю.Н.* Раскраска вершин смешанного графа // Алгоритмы и программы решения задач оптимизации. — Минск: ИТК АН БССР, 1983. — С. 90–95.
138. *Ковалев М.Я.* Эффективный ϵ -приближенный алгоритм для задачи максимизации быстродействия в двухстадийной системе из параллельных приборов // Изв. АН БССР. Сер. физ.-мат. наук. — 1985. — № 3. — С. 118.
139. *Ковалев М.Я.* Приближенное решение задачи минимизации суммарного запаздывания требований // Изв. АН БССР. Сер. физ.-мат. наук. — 1985. — № 5. — С. 110.
140. *Ковалев М.Я., Тузииков А.В.* Построение ϵ -аппроксимации множества Парето некоторых двухкритериальных задач // Мат. вопросы автоматизации проектирования и испытаний. — Минск: ИТК АН БССР, 1986. — С. 126–130.
141. *Ковалев М.Я., Шафранский Я.М.* Построение ϵ -приближенных алгоритмов оптимизации функций на последовательно конструируемых множествах // ЖВМ и МФ. — 1986. — № 7. — С. 1006–1018.
142. *Козан Я.А., Лищинский Л.Ю., Парадизов Н.В.* Модели обслуживания станков в гибких производственных системах // Автомат. и телемех. — 1986. — № 6. — С. 147–155.
143. *Коглер В., Штиглиц К.* Перечислительные и итеративные алгоритмы // Теория расписаний и вычислительные машины. — М.: Наука, 1984. — С. 249–317.
144. *Козлик Г.А., Карась В.М., Кириллов И.А.* Пакет прикладных программ для решения основной задачи теории расписаний // УС и М. — 1979. — № 2. — С. 40–43.
145. *Конвей Р., Джонсон Б., Максвелл У.* Экспериментальные исследования распределения работ в соответствии с фиксированными правилами очередности их выполнения // Применение статистических методов в производстве. — М. — 1963. — С. 212–232.
146. *Конвей Р.У., Максвелл У.Л.* Календарное планирование в условиях сети очередей с дисциплиной по кратчайшей операции // Календарное планирование. — М.: Прогресс, 1966. — С. 321–347.
147. *Конвей Р.В., Максвелл В.Л., Миллер Л.В.* Теория расписаний. — М.: Наука, 1975. — 360 с.
148. *Корбут А.А., Сигал И.Х., Финкельштейн Ю.Ю.* Метод ветвей и границ // Math. Operationsforsch. Statist., Ser. Optimization. — 1977. — V. 8, № 2. — P. 253–280.
149. *Корбут А.А., Финкельштейн Ю.Ю.* Дискретное программирование. — М.: Наука, 1969. — 368 с.
150. *Корнилова Л.Е.* Оптимальное распределение ресурса в сетевом планировании методом оптимального потока // Кибернетика. — 1970. — № 6. — С. 145–148.
151. *Косарев Н.Г., Уздемир А.П.* Динамическая задача планирования научных исследований и разработок и метод ее решения // Автомат. и телемех. — 1977. — № 1. — С. 62–73.
152. *Косолап А.И.* Применение нелинейного программирования к задачам теории расписаний // Численные методы нелинейного программирования: Материалы I Всесоюз. семинара. — Киев, 1976. — С. 145–149.
153. *Костевич Л.С., Лапко А.А.* Теория игр. Исследование операций. — Минск: Вышэйшая школа, 1982. — 232 с.
154. *Кочнев Л.В.* Оптимизация системы приоритета в задачах расписания // Кибернетика и ВУЗ. — Томск: Томск. гос. ун-т, 1970. — Вып. 3. — С. 114–122.
155. *Крейниш М.И., Никонов В.А., Савицкая Т.Н.* О решении одной задачи загрузки оборудования // Вычисл. методы и мат. обеспечение ЭВМ. — Казань, 1979. — С. 76–83.
156. *Кристофидес Н.* Теория графов. Алгоритмический подход. — М.: Мир, 1978. — 432 с.
157. *Кузин Б.И., Дуболазов В.А.* Организация и оперативно-календарное планирование машиностроительного производства в АСУП. — Л.: ЛГУ, 1978. — 238 с.
158. *Кук С.А.* Сложность процедур вывода теорем // Кибернет. сб. Новая серия. М.: Мир, 1975. — Вып. 12. — С. 5–15.
159. *Кука А.И.* К использованию теории двойственности для решения сетевых задач теории расписаний // ЖВМ и МФ. — 1982. — Т. 22, № 5. — С. 1074–1079.
160. *Кука А.И., Лаптин Ю.П.* Динамическое программирование в сетевой задаче теории расписаний // Кибернетика. — 1978. — № 1. — С. 111–113.

161. *Кукса А.И., Лаптин Ю.П.* Использование динамического программирования при двойственном подходе к решению задачи календарного планирования // Изв. АН СССР. Техн. кибернетика. — 1981. — № 4. — С. 79–85.
162. *Кукса А.И., Поляченко Б.Е.* Алгоритм ветвей и границ для решения двух сетевых задач календарного планирования // Кибернетика. — 1984. — № 1. — С. 73–76.
163. *Ламбин Н.В., Танаев В.С.* О бесконтурной ориентации смешанных графов // Докл. АН БССР. — 1970. — Т. 14, № 9. — С. 780–781.
164. *Лапко А.А.* Об оптимизации приоритетных систем обслуживания // Изв. АН БССР. Сер. физ.-мат. наук. — 1977. — № 2. — С. 117–120.
165. *Лапко А.А.* Обслуживание детерминированных потоков требований в заданные сроки // Изв. АН БССР. Сер. физ.-мат. наук. — 1980. — № 5. — С. 130.
166. *Лапко А.А., Поварич М.П.* Диспетчеризация потоков требований с переменными параметрами в приоритетных системах массового обслуживания // Теория и методы оптимизации в проектировании. — Минск: ИТК АН БССР, 1978. — Вып 1. — С. 98–104.
167. *Латыпов И.Ш.* Об алгоритме управления одним классом технологических линий // Исследование операций и аналитическое проектирование в технике. — Казань, 1979. — С. 47–51.
168. *Латыпов И.Ш.* О сложности одной задачи теории расписаний // Системы программного обеспечения решения задач оптимального планирования: Тез. докл. IX Всесоюз. симпозиума. — М., 1986. — С. 88–89.
169. *Лебедева Т.Т., Михалевич В.С., Рошин В.А., Сергиенко И.В., Стукало А.С., Трубин В.А., Шор Н.З.* Структура, состав и назначение пакета программ ДИСПРО для решения задач дискретной оптимизации // Изв. АН СССР. Техн. кибернетика. — 1983. — № 1. — С. 193–196.
170. *Левин В.И.* К задаче трех станков // Изв. АН СССР. Техн. кибернетика. — 1982. — № 1. — С. 76–85.
171. *Левин Г.М.* К оценке хроматических характеристик смешанного графа // Изв. АН БССР. Сер. физ.-мат. наук. — 1980. — № 2. — С. 37–41.
172. *Левин Г.М., Сотсков Ю.Н., Танаев В.С.* О раскраске вершин смешанного графа // Методы и программы решения оптимизационных задач на графах и сетях, ч. 1: Тез. докл. III Всесоюз. совещ. — Новосибирск, 1982. — С. 87–90.
173. *Левин Г.М., Танаев В.С.* Декомпозиционные методы оптимизации проектных решений. — Минск: Наука и техника, 1978. — 240 с.
174. *Левин Л.А.* Универсальные задачи перебора // Проблемы передачи информации. — 1973. — Т. 9, № 3. — С. 115–116.
175. *Левин М.Ш.* Об эффективном решении некоторых задач теории расписаний на сетях // Кибернетика. — 1980. — № 1. — С. 131–135.
176. *Левин М.Ш., Левнер Е.В.* Об эффективном решении задач Беллмана — Джонсона на сети, имеющей вид дерева // Автомат. и телемех. — 1978. — № 10. — С. 110–116.
177. *Левнер Е.В.* Оптимальное планирование обработки объектов на ряде машин // Автомат. и телемех. — 1969. — № 17. — С. 94–100.
178. *Левнер Е.В.* Сетевой подход к задачам теории расписаний // Исследования по дискретн. мат. — М.: Наука, 1973. — С. 135–150.
179. *Левнер Е.В.* Сетевой подход к задачам теории расписаний // Современные проблемы теории управления. — М.: ИПУ. — 1973. — С. 43–54.
180. *Левнер Е.В.* Теория расписаний в экономических системах (некоторые математические вопросы). — Препринт / ЦЭМИ АН СССР. — М., 1977. — 54 с.
181. *Левнер Е.В., Генс Г.В.* Анализ вычислительной сложности приближенных алгоритмов для некоторых дискретных оптимизационных задач // Мат. методы решения экон. задач. — 1980. — № 9. — С. 97–106.
182. *Леонтьев В.К.* Устойчивость задачи коммивояжера // ЖВМ и МФ. — 1975. — Т. 15, № 5. — С. 1298–1309.
183. *Леонтьев В.К.* Устойчивость в комбинаторных задачах выбора // ДАН СССР. — 1976. — Т. 228, № 1. — С. 23–25.
184. *Леонтьев В.К.* Дискретные экстремальные задачи // Итоги науки и техники. Теория вероятностей. Математическая статистика. Теоретическая кибернетика. — 1979. — Т. 16. — С. 39–101.
185. *Лепешинский Н.А.* К вопросу упорядочения обработки деталей // Изв. АН БССР. Сер. физ.-мат. наук. — 1966. — № 2. — С. 31–35.

186. *Лепешинский Н.А.* Об одной задаче теории расписаний // Изв. АН БССР. Сер. физ.-мат. наук. — 1966. — № 3. — С. 90–96.
187. *Лепешинский Н.А.* О классификации $2 \times n$ задач Джонсона — Беллмана // Изв. АН БССР. Сер. физ.-мат. наук. — 1967. — № 1. — С. 18–21.
188. *Лепешинский Н.А.* Одна задача теории расписаний и максимум суммы линейных форм на множестве подстановок // Изв. АН БССР. Сер. физ.-мат. наук. — 1967. — № 3. — С. 51–54.
189. *Лескин А.А.* Алгебраические модели гибких производственных систем. — Л.: Наука, 1986. — 153 с.
190. *Лившиц Э.М., Рублинецкий В.И.* О сравнительной сложности некоторых задач дискретной оптимизации // Вычислит. мат. и вычислит. техн. — 1972. — Вып. 3. — С. 78–85.
191. *Литл Дж., Мурти К., Суини Д., Кэрел К.* Алгоритм для решения задачи о коммивояжере // Экон. и мат. методы. — 1965. — Т. 4, вып. 1. — С. 94–107.
192. *Ломинадзе Н.Н.* О некоторых возможностях построения и использования эвристических правил предпочтения // Труды Проблемной лаб. автомат. и вычислит. техн. Техн. кибернетика. — Груз. политехн. ин-т, 1974. — № 4. — С. 43–46.
193. *Ломинадзе Н.Н.* О решении задачи упорядочения и составления расписаний при нескольких показателях качества // Труды Проблемной лаб. автомат. и вычислит. техн. Техн. кибернетика. — Груз. политехн. ин-т, 1974. — № 4. — С. 37–42.
194. *Лурье А.Л.* О некоторых задачах календарного планирования // Пробл. кибернетики. — М.: Наука, 1962. — Вып. 7. — С. 201–208.
195. *Лурье А.Л.* О математических методах решения задач на оптимум при планировании социалистического хозяйства. — М.: Наука, 1964. — 323 с.
196. *Макаров И.М., Рахманкулов В.З., Назаретов В.М., Блинов С.А., Михайлов А.М., Манько С.В., Ахрем А.А.* Робототехника и гибкие автоматизированные производства. Управление робототехническими системами и гибкими автоматизированными производствами. — М.: Высшая школа, 1986. — 160 с.
197. *Марков Л.А., Первозванский А.А.* Правило узких мест в задаче Джонсона // Кибернетика. — 1974. — № 5. — С. 142–144.
198. *Маршалл А., Олкин И.* Неравенства: Теория мажоризации и ее приложения. — М.: Мир, 1983. — 576 с.
199. *Матюшков Л.П., Танаев В.С.* Программные генераторы допустимых расписаний. I, II // Вычислит. техн. в машиностроении. — Минск: ИТК АН БССР, 1967. — С. 35–48; 1968. — С. 12–28.
200. *Мельников О.И.* К задаче Беллмана — Джонсона // Изв. АН БССР. Сер. физ.-мат. наук. — 1975. — № 4. — С. 43–47.
201. *Мельников О.И.* Свойства оптимальных расписаний задачи Беллмана — Джонсона для двух станков // Изв. АН БССР. Сер. физ.-мат. наук. — 1976. — № 3. — С. 19–25.
202. *Мельников О.И.* Решение некоторых задач Беллмана — Джонсона с (0,1)-матрицами // Изв. АН БССР. Сер. физ.-мат. наук. — 1976. — № 4. — С. 11–16.
203. *Мельников О.И.* Устойчивость оптимального расписания задачи Беллмана — Джонсона // Изв. АН БССР. Сер. физ.-мат. наук. — 1976. — № 6. — С. 99–101.
204. *Мельников О.И.* О $2 \times n$ -задаче Беллмана — Джонсона, для которой все расписания оптимальны // Изв. АН БССР. Сер. физ.-мат. наук. — 1984. — № 2. — С. 31–33.
205. *Мельников О.И., Шафранский Я.М.* Минимизация средней длительности обслуживания в детерминированных системах с двумя приборами // Изв. АН БССР. Сер. физ.-мат. наук. — 1971. — № 3. — С. 124.
206. *Мельников О.И., Шафранский Я.М.* Параметрическая задача теории расписаний // Кибернетика. — 1979. — № 3. — С. 53–57.
207. *Метельский А.С.* О некоторых периодических процессах // Докл. АН БССР. — 1963. — Т. 7, № 9. — С. 584–587.
208. *Метельский А.С.* К системе периодических процессов // Докл. АН БССР. — 1963. — Т. 7, № 11. — С. 724–728.
209. *Метельский А.С.* Периодические процессы // Докл. АН БССР. — 1965. — Т. 9, № 12. — С. 788–790.
210. *Мельников Н.Н.* Об экстремальных значениях линейной формы на некоторых множествах подстановок // Изв. АН БССР. Сер. физ.-мат. наук. — 1972. — № 5. — С. 5–10.

211. *Мироносецкий Н.Б.* Экономико-математические методы календарного планирования. — Новосибирск : Наука, 1974. — 140 с.
212. *Мироносецкий Н.Б., Андерсон А.Р.* Управление подготовкой производства. — Новосибирск: Наука, 1976. — 160 с.
213. *Михалевич В.С.* Последовательные алгоритмы оптимизации и их применение. I, II // Кибернетика. — 1965. — № 1. — С. 45–56; 1965. — № 2. — С. 85–89.
214. *Михалевич В.С., Волков В.Л., Волошин А.Ф., Поздняков Ю.М.* Алгоритмы последовательного анализа и отсеивания вариантов в задачах дискретной оптимизации // Кибернетика. — 1980. — № 3. — С. 76–85.
215. *Михалевич В.С., Кукса А.И.* Методы последовательной оптимизации в дискретных сетевых задачах оптимального распределения ресурсов. — М.: Наука, 1983. — 217 с.
216. *Михалевич В.С., Шкурба В.В.* Последовательные схемы оптимизации в задачах упорядочения выполнения работ // Кибернетика. — 1966. — № 2. — С. 34–40.
217. *Михалевич В.С., Шор Н.З.* Численное решение многовариантных задач по методу последовательного анализа вариантов // Науч.-метод. материалы экон.-мат. семинара. — М.: ВЦ АН СССР. — 1962. — Вып. 1. — С. 15–42.
218. *Мищенко А.В., Сушков Б.Г.* Задача оптимального распределения ресурсов на сетевой модели при линейных ограничениях на время выполнения работ // ЖВМ и МФ. — 1980. — Т. 20, № 5. — С. 1324–1328.
219. *Моисеев Н.Н.* Численные методы оптимального управления. — М.: ВЦ МГУ, 1968.
220. *Моисеев Н.Н.* Численные методы в теории оптимальных систем. — М.: Наука, 1971. — 424 с.
221. *Моисеев Н.Н.* Элементы теории оптимальных систем. — М.: Наука, 1975. — 526 с.
222. *Моисеев Н.Н.* Математика ставит эксперимент. — М.: Наука, 1979. — 223 с.
223. *Моисеев Н.Н., Иванилов Ю.П., Столяров Е.М.* Методы оптимизации. — М.: Наука, 1978. — 351 с.
224. *Муравьев В.И., Романовский И.В.* Использование метода переменного базиса для решения непрерывного аналога задачи Джонсона // Исследование операций и статист. моделирование. — Л., 1974. — Вып. 2. — С. 27–37.
225. *Носов В.А., Сачков В.Н., Тараканов В.Е.* Комбинаторный анализ. Матричные проблемы, теория выбора // Итоги науки и техники. Теория вероятностей. Математическая статистика. Теоретическая кибернетика. — 1981. — Т. 18. — С. 53–93.
226. *Носов В.А., Сачков В.Н., Тараканов В.Е.* Комбинаторный анализ (неотрицательные матрицы, алгоритмические проблемы) // Итоги науки и техники. Теория вероятностей. Математическая статистика. Теоретическая кибернетика. — 1983. — Т. 21. — С. 120–178.
227. *Нуриев Н.К., Гудин Н.В., Кайдыров Р.А., Журавлев Б.Л.* Организация оптимальной работы автооператорных гальванических линий // Казань: Казанский химико-технолог. ин-т им. С.М. Кирова, 1986. — 80 с.
228. *Оганесян Г.А.* Некоторые свойства и решение частных случаев задачи теории расписаний // Вопросы совершенствования планирования и управления. — Душанбе: НИИЭМП Госплана ТаджССР, 1978. — Вып. 14. — С. 133–145.
229. *Оганесян Г.А.* Эффективный алгоритм решения частного случая задачи коммивояжера // Экономико-математическое моделирование в системах управления. — Душанбе: НИИЭМП Госплана ТаджССР, 1980. — Вып. 16. — С. 123–125.
230. *Озерной В.М., Рябов Л.П.* Эвристический метод оптимизации последовательности выполнения операций // Автомат. и телемех. — 1967. — № 12. — С. 169–172.
231. Организация, планирование и управление предприятием электронной промышленности / Под ред. П.М. Стукалова. — М.: Высшая школа, 1980. — 351 с.
232. *Оре О.* Теория графов. — М.: Наука, 1980. — 336 с.
233. *Осколкова С.Е., Осколков И.О.* Применение некоторых эвристических методов к решению задач календарного планирования (обзор) // Автомат. и телемех. — 1968. — № 2. — С. 177–184.
234. Основные положения по разработке и применению систем сетевого планирования и управления. — М.: Экономика, 1974. — 216 с.
235. *Офицеров Д.В., Свириг Ю.П.* Метод решения задачи оперативно-календарного планирования автоматизированного производства // Изв. АН БССР. Сер. физ.-техн. наук. — 1985. — № 3. — С. 78–83.

236. Панишев А.В. Метод дихотомии в задаче трех станков // Изв. АН СССР. Техн. кибернетика. — 1986. — № 4. — С. 181–186.
237. Панишев А.В., Данильченко А.М. Вычисление границ в задаче трех станков // Автомат. системы управл. и приборы автомат. — Харьков: Вища школа, 1984. — Вып. 69. — С. 62–66.
238. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. — М.: Мир, 1985. — 510 с.
239. Перепелица В.А. Об одной задаче теории расписаний // Кибернетика. — 1966. — № 5. — С. 75–78.
240. Перепелица В.А., Севастьянов С.В. Об одной задаче теории расписаний на сети // Управляемые системы. — Новосибирск, 1976. — Вып. 15. — С. 48–67.
241. Петров В.А. Планирование поточно-группового производства. — Л.: Машиностроение, 1966. — 192 с.
242. Петров В.А. Групповое производство и автоматизированное оперативное управление. — Л.: Машиностроение, 1975. — 312 с.
243. Петров В.А., Маслеников А.Н., Осипов Л.А. Планирование гибких производственных систем. — Л.: Машиностроение, 1985. — 182 с.
244. Пилипенко В.А., Канарский В.Ф., Воловин Л.Н. Система краткосрочного планирования работы механообрабатывающего участка в условиях АСУ // УС и М. — 1986. — № 3. — С. 108–110.
245. Плещинский Л.С. ϵ -подход к оптимизации сетевой модели // Экон. и мат. методы. — 1976. — Т. 12, № 6. — С. 1212–1215.
246. Подчасова Т.П. Об оценках и выборе правил предпочтения в задачах календарного планирования // Автомат. системы упр. предприятиями: Тр. семинара. — 1968. — Вып. 1. — С. 5–46.
247. Подчасова Т.П., Португал В.М., Татаров В.А., Шкурба В.В. Эвристические методы календарного планирования. — Киев: Техніка, 1980. — 140 с.
248. Португал В.М. Исследование асимптотического поведения задачи Джонсона // Кибернетика. — 1971. — № 1. — С. 105–107.
249. Португал В.М. Асимптотическое поведение решений некоторых задач теории расписаний // Кибернетика. — 1976. — № 2. — С. 132–134.
250. Португал В.М. Асимптотическое поведение решений задач теории расписаний по различным критериям // Кибернетика. — 1976. — № 5. — С. 97–99.
251. Португал В.М., Семенов А.И. Модели планирования на предприятии. — М.: Наука, 1978. — 270 с.
252. Потиебовски Г. Приближенные алгоритмы решения общей задачи теории расписаний // Прикл. пробл. больших систем управления. — София, 1980. — С. 115–124.
253. Ревчук И.Н. Условия осуществимости полного обслуживания в системе с M приборами, имеющими окна доступности // Изв. АН БССР. Сер. физ.-мат. наук. — 1977. — № 4. — С. 123.
254. Ревчук Т.Н., Сотсков Ю.Н. Выделение контуров смешанного графа // Методы и программы решения экстремальных задач. — Минск: ИТК АН БССР, 1981. — С. 46–52.
255. Резниченко В.Н. Решение одной задачи теории расписаний эвристическим методом // Алгоритмизация производственных процессов: Тр. семинара. — Киев, 1968. — Вып. 1. — С. 91–110.
256. Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. — М.: Мир, 1980. — 476 с.
257. Романовский И.В. Асимптотика рекуррентных соотношений динамического программирования и оптимальное стационарное управление // ДАН СССР. — 1964. — Т. 157, № 6. — С. 1303–1306.
258. Романовский И.В. Численные методы дискретного программирования // Тр. 4-й зимней школы по мат. программированию и смежн. вопр. — М., 1972. — Вып. 5. — С. 76–128.
259. Рубинов А.Р. Теоремы о магистральных в обобщенной задаче коммивояжера и их применение к задачам раскроя и теории расписаний // ДАН СССР. — 1981. — Т. 259, № 3. — С. 542–546.
260. Рубинов А.Р. О сводимости обобщенной задачи коммивояжера к задаче коммивояжера меньшей размерности // ДАН СССР. — 1982. — Т. 264, № 5. — С. 1087–1090.

261. Рыбальский В.И. Кибернетика в строительстве. — Киев: Будівельник, 1975. — 232 с.
262. Рыцов И.К. О понятии расписаний для многопроцессорных систем // Развитие теории многопроцессорных систем. — Киев, 1984. — С. 63–68.
263. Сальвадор М. Календарное планирование и упорядочение работ // Исследование операций. — М.: Мир, 1981. — Т. 2. — С. 232–263.
264. Санникова А.К. Об оптимизации функций на перестановках с ограничением на группирование элементов // Изв. АН БССР. Сер. физ.-мат. наук. — 1982. — № 1. — С. 116–117.
265. Санникова А.К., Танаев В.С. Об одном классе экстремальных задач на перестановках // Докл. АН БССР. — 1979. — Т. 23, № 9. — С. 784–786.
266. Сарванов В.И. Приближенное решение задачи минимизации линейной формы на множестве циклических подстановок // Изв. АН БССР. Сер. физ.-мат. наук. — 1977. — № 6. — С. 5–10.
267. Сарванов В.И. К оптимизации на подстановках // Изв. АН БССР. Сер. физ.-мат. наук. — 1979. — № 4. — С. 9–11.
268. Сарванов В.И. О сложности минимизации линейной формы на множестве циклических подстановок // ДАН СССР. — 1980. — Т. 253, № 3. — С. 533–535.
269. Сафроненко В.А. Математическое и электронное моделирование задачи оптимального календарного планирования. — Минск: Наука и техника, 1972. — 182 с.
270. Сачков В.Н. Комбинаторные методы дискретной математики. — М.: Наука, 1977. — 320 с.
271. Севастьянов С.В. Об асимптотическом подходе к некоторым задачам теории расписаний // III Всесоюз. конф. по проблемам теорет. кибернетики: Тез. докл. — Новосибирск, 1974. — С. 67–69.
272. Севастьянов С.В. Об асимптотическом подходе к некоторым задачам теории расписаний // Управляемые системы. — 1975. — Вып. 14. — С. 40–51.
273. Севастьянов С.В. О приближенном решении некоторых задач теории расписаний // Дискретный анализ. — 1978. — Вып. 32. — С. 66–75.
274. Севастьянов С.В. О приближенном решении задачи календарного распределения // Управляемые системы. — 1980. — Вып. 20. — С. 49–63.
275. Севастьянов С.В. Приближенные алгоритмы в задачах Джонсона и суммирования векторов // Управляемые системы. — 1980. — Вып. 20. — С. 64–73.
276. Севастьянов С.В. Некоторые обобщения задачи Джонсона // Управляемые системы. — 1981. — Вып. 21. — С. 45–61.
277. Севастьянов С.В. Алгоритмы с оценками для задач Джонсона и Акерса — Фридмана в случае трех станков // Управляемые системы. — 1982. — Вып. 22. — С. 51–57.
278. Севастьянов С.В. Эффективное построение расписаний, близких к оптимальным, для случаев произвольных и альтернативных маршрутов деталей // ДАН СССР. — 1984. — Т. 276, № 1. — С. 46–49.
279. Семенов И.Г. Алгоритм упорядочения работ в конвейерной системе по критерию минимума времени прохождения работ и неизменном порядке их выполнения на всех машинах // Автоматизация управления промышленными объектами. — Киев, 1980. — С. 85–98.
280. Семенов А.И., Португал В.М. Задачи теории расписаний в календарном планировании мелкосерийного производства. — М.: Наука, 1972. — 183 с.
281. Сервах В.В. О задаче Акерса — Фридмана // Управляемые системы. — 1983. — Вып. 23. — С. 70–81.
282. Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации. — Киев: Наукова думка, 1985. — 384 с.
283. Скворцов В.В., Нурiev Н.К. Постановка и подходы к решению многоэкстремальной задачи оптимизации работы поточной линии в изменяющихся условиях // Теория оптимальных решений. — Вильнюс, 1978. — Вып. 5. — С. 55–59.
284. Скрыдлов Н.В., Шафранский В.В. Применение правил приоритета для составления расписания работ при ограничениях на ресурсы // Работы по техн. кибернетике. — М.: ВЦ АН СССР, 1967. — Вып. 1. — С. 67–79.
285. Смирнов А.Н. Математическая модель объемного календарного планирования технологически зависимых операций // Автомат и телемех. — 1985. — № 10. — С. 108–115.

286. *Смоляр Л.И.* Теория расписаний и управление. — М.: Знание, 1977. — 64 с.
287. *Смоляр Л.И.* Модели оперативного планирования в дискретном производстве. — М.: Наука, 1978. — 320 с.
288. *Смоляр Л.И.* Оперативно-календарное планирование (Модели и методы). — М.: Экономика, 1979. — 136 с.
289. *Солдатов В.С.* Оптимальное решение задач теории расписаний; работы допускают прерывания // Тр. XXII Научной конф. Моск. физ.-техн. ин-та. — Долгопрудный: МФТИ, 1976. — С. 125–128.
290. *Солих Р.* Задача календарного планирования для циклически повторяющегося производства // ЖВМ и МФ. — 1973. — Т. 13, № 2. — С. 326–342.
291. *Сотсков Ю.Н.* Оптимальное расписание множества работ, заданного смешанным графом // Изв. АН БССР. Сер. физ.-мат. наук. — 1977. — № 4. — С. 133.
292. *Сотсков Ю.Н.* Сетевое планирование и управление в строительстве с использованием смешанных графов // Системный подход в управлении строительством. — Минск: ИС и А Госстроя БССР, 1977. — С. 109–114.
293. *Сотсков Ю.Н.* Составление расписаний на смешанных графах ϵ -методом // Автоматизированное проектирование технических систем и процессов. — Минск: ИТК АН БССР, 1979. — С. 49–53.
294. *Сотсков Ю.Н.* Задача теории расписаний на смешанных графах // Теория и методы автоматизации проектирования. — Минск: ИТК АН БССР, 1980. — С. 19–22.
295. *Сотсков Ю.Н.* Определение числа орграфов, порождаемых смешанным графом // Алгоритмы и программы решения задач оптимизации. — Минск: ИТК АН БССР, 1980. — С. 36–39.
296. *Сотсков Ю.Н.* Перечисление расписаний, заданных смешанным графом // Изв. АН БССР. Сер. физ.-мат. наук. — 1980. — № 5. — С. 132–133.
297. *Сотсков Ю.Н.* Построение орграфов, порождаемых смешанным графом // Алгоритмы и программы решения задач оптимизации. — Минск: ИТК АН БССР, 1980. — С. 40–44.
298. *Сотсков Ю.Н.* Расписания на смешанных графах с ограниченным максимальным штрафом // Изв. АН БССР. Сер. физ.-мат. наук. — 1980. — № 2. — С. 37–41.
299. *Сотсков Ю.Н.* Об ориентировании ребер смешанного графа // Изв. АН БССР. Сер. физ.-мат. наук. — 1981. — № 5. — С. 22–24.
300. *Сотсков Ю.Н.* Составление оптимального по быстродействию расписания выполнения взаимосвязанных работ последовательными приборами // Методы и программы решения экстремальных задач. — Минск: ИТК АН БССР, 1981. — С. 28–34.
301. *Сотсков Ю.Н.* О методе решения общей задачи теории расписаний // I Всесоюз. совещание по статист. и дискр. анализу нечисловой информ., экспертным оценкам и дискр. оптимизации: Тез. докл.: — М., 1981. — С. 218–219.
302. *Сотсков Ю.Н.* Оптимальное обслуживание двух требований при регулярном критерии // Автоматизация процессов проектирования. — Минск: ИТК АН БССР, 1985. — С. 86–95.
303. *Сотсков Ю.Н.* Сетевые модели в теории расписаний // Оптимизация, принятие решений, микропроцессорные системы. — София: ИТКР БАН, 1985. — С. 157–162.
304. *Сотсков Ю.Н.* Сетевой подход к решению задач теории расписаний // Методы, алгоритмы и программы решения экстремальных задач. — Минск: ИТК АН БССР, 1985. — С. 52–62.
305. *Сотсков Ю.Н., Алюшкевич В.Б.* Устойчивость оптимальной ориентации ребер смешанного графа // Докл. АН БССР. — 1988. — Т. 32, № 4. — С. 108–111.
306. *Сотсков Ю.Н., Барановская С.М.* Минимизация максимального штрафа за обслуживание требований последовательными приборами // Сложность и методы решения задач оптимизации. — Минск: ИТК АН БССР, 1984. — С. 37–47.
307. *Сотсков Ю.Н., Танаев В.С.* О перечислении ориентированных бесконтурных графов, порождаемых смешанным графом // Изв. АН БССР. Сер. физ.-мат. наук. — 1974. — № 2. — С. 16–21.
308. *Сотсков Ю.Н., Танаев В.С.* Об одном подходе к перечислению ориентированных бесконтурных графов, порождаемых смешанным графом // Изв. АН БССР. Сер. физ.-мат. наук. — 1976. — № 5. — С. 99–102.
309. *Сотсков Ю.Н., Танаев В.С.* Хроматический многочлен смешанного графа // Изв. АН БССР. Сер. физ.-мат. наук. — 1976. — № 6. — С. 20–23.

310. Сотсков Ю.Н., Черняк Ж.А. Определение параметров сетевого графика // Алгоритмы и программы решения экстремальных задач и смежные вопросы. — Минск: ИТК АН БССР, 1982. — С. 42—53.
311. Столин Я.Н. Решение одомаршрутной задачи Джонсона для трех машин с погрешностью 4 max t_{ij} // Численные методы нелинейного программирования: Тез. II Всесоюз. семинара. — Харьков, 1976. — С. 287—289.
312. Стори А.Е., Вагнер Х.М. Опыт применения целочисленного программирования при расчете календарного плана для предприятий единичного и мелкосерийного производства // Календарное планирование. — М.: Прогресс, 1966. — С. 241—256.
313. Струсович В.А. Некоторые эффективно разрешимые классы задачи Беллмана — Джонсона // Математические методы и их приложения в больших экономических и технических системах: Тез. докл. Всесоюз. школы-семинара. — М., 1980. — С. 177—180.
314. Струсович В.А. К задаче Беллмана — Джонсона // Докл. АН БССР. — 1981. — Т. 25, № 6. — С. 518—521.
315. Струсович В.А. Задачи конвейерного типа с регулярными матрицами. — Минск, 1981. — 39 с. — Деп. в ВИНТИ 4.08.81, № 3857—81.
316. Струсович В.А. Эффективно разрешимые классы задачи Беллмана — Джонсона и ее модификаций. — Минск, 1981. — 43 с. — Деп. в ВИНТИ 4.08.81, № 3858—81.
317. Струсович В.А. О возможности приближенного решения одной задачи теории расписаний // I Всесоюз. совещ. по статист. и дискр. анализу нечисловой информации, экспертным оценкам и дискр. оптимизации: Тез. докл. — М., 1981. — С. 222—223.
318. Струсович В.А. Класс регулярных задач Беллмана — Джонсона // Изв. АН БССР. Сер. физ.-мат. наук. — 1982. — № 1. — С. 28—33.
319. Струсович В.А. Об одной задаче теории расписаний с древовидными ограничениями предшествования // Методы и программы решения оптимизационных задач на графах и сетях, ч. 1.: Тез. докл. III Всесоюз. совещ. — Новосибирск, 1984. — С. 221—223.
320. Струсович В.А. Сложность решения некоторых задач теории расписаний с заданными отношениями предшествования // Сложность и методы решения задач оптимизации. — Минск: ИТК АН БССР, 1984. — С. 9—20.
321. Струсович В.А. Задача трех станков: достаточное условие сводимости в выпуклой форме // Вестн. БГУ им. В.И. Ленина. Сер. 1, физ., мат., мех. — 1985. — № 1. — С. 36—39.
322. Струсович В.А. Доминирующие приборы в детерминированных системах // Применение матем. методов и вычисл. техники при решении народнохозяйственных задач: Тез. докл. респ. научн.-техн. совещ. — Минск, 1986. — С. 151—152.
323. Струсович В.А. О возможности построения оптимальных по быстродействию расписаний для многостадийной системы с нефиксированными маршрутами прохождения стадий // Изв. АН БССР. Сер. физ.-мат. наук. — 1986. — № 6. — С. 43—48.
324. Струсович В.А. Минимизация суммарного времени обслуживания для системы с нефиксированными маршрутами // Вестн. БГУ им. В.И. Ленина. Сер. 1, физ., мат., мех. — 1988. — № 1. — С. 44—46.
325. Струсович В.А., Довнар Ю.С. Модифицированный алгоритм решения одного класса задач Беллмана — Джонсона // ЭВМ — учебному процессу, науке и технике. — Минск: Вышэйшая школа, 1985. — С. 58—62.
326. Супруненко Д.А. О значениях линейной формы на множестве подстановок // Кибернетика. — 1968. — № 2. — С. 59—63.
327. Супруненко Д.А., Айзенштат В.С., Лепешинский Н.А. Экстремальные значения функций на множествах подстановок // I Всесоюз. конф. по исслед. операций: Тез. докл. — Минск, 1977. — С. 61—64.
328. Супруненко Д.А., Айзенштат В.С., Метельский А.С. Многоэтапный процесс превращения // Докл. АН БССР. — 1962. — Т. 6, № 9. — С. 541—544.
329. Супруненко Д.А., Метельский Н.Н. Задача о назначениях и минимизация суммы линейных форм на симметрической группе // Кибернетика. — 1973. — № 3. — С. 64—68.
330. Супруненко Д.А., Тышкевич Р.И. Динамические отображения и один класс детерминированных машин // Кибернетика. — 1965. — № 2. — С. 9—17.

331. Сытник В.Ф. О математической структуре задачи календарного планирования // Исследования операций и АСУ. — Киев: КГУ, 1975. — Вып. 5. — С. 14–23.
332. Сытник В.Ф. Некоторые результаты исследования задачи Джонсона // Исследования операций и АСУ. — Киев: КГУ, 1976. — Вып. 8. — С. 17–25.
333. Сытник В.Ф., Кузьменко Л.И. Некоторые модификации алгоритма решения календарной задачи по методу Монте-Карло // Исследование операций и АСУ. — Киев: КГУ, 1975. — Вып. 6. — С. 45–51.
334. Сытник В.Ф., Кузьменко Л.И. Симулятивная модель календарного планирования и ее реализации на ЭВМ // Исследование операций и АСУ. — Киев: КГУ, 1976. — Вып. 7. — С. 39–48.
335. Танаев В.С. К задаче составления расписания работы поточной линии с одним автооператором // Инж. физ. ж. — 1964. — Т. 7, № 3. — С. 111–114.
336. Танаев В.С. Об одной задаче теории расписаний // Изв. АН БССР. Сер. физ.-мат. наук. — 1964. — № 4. — С. 128–131.
337. Танаев В.С. Теория расписаний. — М.: Знание. — 1988. — 40 с.
338. Танаев В.С. Некоторые оптимизируемые функции одностадийного производства // Докл. АН БССР. — 1965. — Т. 9, № 1. — 11–14.
339. Танаев В.С. О числе перестановок n частично упорядоченных элементов // Докл. АН БССР. — 1967. — Т. 11, № 3. — С. 208.
340. Танаев В.С. Метод решения задач дискретного программирования // Экон. и мат. методы. — 1968. — Т. 4, вып. 5. — С. 776–782.
341. Танаев В.С. Прерывания в детерминированных системах обслуживания с параллельными идентичными приборами // Изв. АН БССР. Сер. физ.-мат. наук. — 1973. — № 6. — С. 44–48.
342. Танаев В.С. Смешанные (дизъюнктивные) графы в теории расписаний // Большие системы информации и управления. — София: ИТКР БАН, 1975. — С. 181.
343. Танаев В.С. Об оптимизации функций, рекуррентно заданных на множествах перестановок // Изв. АН БССР. Сер. физ.-мат. наук. — 1977. — № 3. — С. 27–30.
344. Танаев В.С. Об оптимальном разбиении конечного множества на подмножества // Докл. АН БССР. — 1979. — Т. 23, № 1. — С. 26–28.
345. Танаев В.С. Современное состояние и основные тенденции развития теории расписаний // Оптимизация, принятие решений, микропроцессорные системы. — София: ИТКР БАН, 1985. — С. 148–151.
346. Танаев В.С. Некоторые вопросы оптимизации проектных решений в САПР // Автоматизация процессов проектирования. — Минск: ИТК АН БССР, 1985. — С. 73–85.
347. Танаев В.С. Декомпозиция и агрегирование в задачах математического программирования. — Минск: Наука и техника, 1987. — 184 с.
348. Танаев В.С., Гордон В.С., Сотсков Ю.Н., Янова О.В., Шафранский Я.М., Горох О.В., Барановская С.М. Пакет программ решения задач оптимального упорядочения (ППП РУПОР). — Минск: ИТК АН БССР, 1986. — 66 с.
349. Танаев В.С., Гордон В.С., Сотсков Ю.Н., Янова О.В., Шафранский Я.М., Горох О.В., Барановская С.М. Пакет программ решения задач оптимального упорядочения (ППП РУПОР). Описание программ. — Минск: ИТК АН БССР, 1987. — 86 с.
350. Танаев В.С., Гордон В.С., Шафранский Я.М. Теория расписаний. Одностадийные системы. — М: Наука, 1984. — 381 с.
351. Танаев В.С., Шкурба В.В. Введение в теорию расписаний. — М.: Наука, 1975. — 256 с.
352. Теория расписаний и вычислительные машины // Под ред. Э.Г. Коффмана. — М.: Наука, 1984. — 336 с.
353. Тимковский В.Г. Приближенный полиномиальный алгоритм для общей задачи теории расписаний // Мат. методы оптимизации и их приложения в больших экон. и техн. системах: Тез. докл. Всесоюз. школы-семинара. — М.: 1980. — С. 181–184.
354. Тимковский В.Г. К сложности составления расписания произвольной системы // Изв. АН СССР. Техн. кибернетика. — 1985. — № 3. — С. 102–109.
355. Тимковский В.Г. Полиномиальное решение задачи Ленстры — Ринной Кана о составлении расписания работ на двух машинах // Кибернетика. — 1985. — № 2. — С. 109–111.

356. *Тимковский В.Г.* Приближенное решение задачи составления расписания циклической системы // Экон. и мат. методы. – 1986. – Т. 22, № 1. – С. 171–174.
357. *Тузигов А.В.* Минимизация максимального штрафа и решение некоторых двухкритериальных задач теории расписаний // Докл. АН БССР. – 1983. – Т. 27, № 10. – С 907–910.
358. *Тузигов А.В.* О двухкритериальной задаче теории расписаний с учетом изменения длительностей обслуживания // ЖВМ и МФ. – 1984. – № 10. – С. 1585–1590.
359. *Тузигов А.В.* О двухкритериальных задачах теории расписаний // Оптимизация. принятие решений, микропроцессорные системы. – София: ИТКР БАН, 1985. – С. 163–168.
360. *Тузигов А.В.* Оптимальное обслуживание двух требований в системе с ограниченными ресурсами // Методы, алгоритмы и программы решения экстремальных задач. – Минск: ИТК АН БССР, 1985. – С. 43–51.
361. *Тютюкин В.К.* Нахождение оптимальных планов в многооперационных процессах обработки изделий методом "ветвей и границ" // Применение математики в экономике. – Л., 1970. – Вып. 6. – С. 29–45.
362. *Тютюкин В.К.* Об оптимизации календарной длительности обработки изделий при одинаковой последовательности запуска на всех станках // Оптимальное планирование. – 1970. – Вып. 16. – С. 89–97.
363. *Тютюкин В.К.* Оптимальное двухоперационное расписание при параллельно-последовательном движении производства // Применение математики в экономике. – Л., 1976. – Вып. 11. – С. 60–69.
364. *Уздемир А.П.* Задачи планирования дискретного (штучного) производства и численные методы их решения I, II // Автомат. и телемех. – 1975. – № 9. – 115–122; 1975. – № 10. – С. 90–104.
365. *Уздемир А.П., Большаков В.А.* Система планирования дискретного производства. – М.: ВНИИСИ. – 1983. – 32 с.
366. *Уздемир А.П., Шмелев В.В.* Целочисленные динамические задачи экономического планирования с сетевыми ограничениями. I, II // Автомат. и телемех. – 1978. – № 7. – С. 106–115; 1978. – № 9. – С. 110–120.
367. *Фейгин Л.И.* Управление и прогнозирование в задачах теории расписаний при неполной информации // ДАН СССР. – 1971. – Т. 200, № 6. – С. 1298–1301.
368. *Фейгин Л.И.* Общая задача теории расписаний при неполной информации // Автомат. и телемех. – 1972. – № 3. – С. 110–116.
369. *Фейгин Л.И.* Векторная оптимизация в задачах теории расписаний при неполной информации // Изв. АН СССР. Техн. кибернетика. – 1976. – № 5. – С. 3–9.
370. *Фейгин Л.И.* Статистическая оценка оптимизации в общей задаче теории расписания // Изв. АН СССР. Техн. кибернетика. – 1982. – № 2. – С. 221–223.
371. *Фейгин Л.И.* Задачи теории расписаний при нечетких длительностях операций // ДАН СССР. – 1983. – Т. 272, № 4. – С. 812–815.
372. *Финкельштейн Ю.Ю.* Приближенные методы и прикладные задачи дискретного программирования. – М.: Наука, 1976. – 264 с.
373. *Фишер Г., Томпсон Г.Л.* Комбинации локальных правил календарного планирования применительно к самонастраивающимся на вероятностной основе программам для ЭВМ // Календарное планирование. – М.: Прогресс, 1966. – С. 260–290.
374. *Форд Л., Фалкерсон Д.* Потоки в сетях. – М.: Мир, 1966. – 276 с.
375. *Франчук В.И.* Некоторые алгоритмы решения задач календарного планирования // Мат. методы решения экон. задач. – 1974. – Вып. 5. – С. 109–113.
376. *Халафян А.А.* Разномаршрутная задача теории расписаний // Ученые – науке и народному хозяйству. – Краснодар, 1983. – С. 276–280.
377. *Харари Ф.* Теория графов. – М.: Мир, 1973. – 300 с.
378. *Харди Г.Г., Литтлвуд Дж.Е., Полиа Г.* Неравенства. – М.: ИЛ, 1948. – 456 с.
379. *Хачатуров В.Р.* Аппроксимационно-комбинаторный метод и некоторые его приложения // ЖВМ и МФ. – 1974. – Т. 14, № 6. – С. 1464–1487.
380. *Хачиян Л.Г.* Полиномиальный алгоритм в линейном программировании // ДАН СССР. – 1979. – Т. 244, № 5. – С. 1093–1096.
381. *Хенкин В.Э.* Об одном вопросе теории расписаний (2 × M задача упорядочения) // Кибернетика. – 1966. – № 6. – С. 67–71.

382. Чеботарь К.С. Об одной задаче оптимизации на множестве перестановок // Изв. АН МССР. Сер. физ.-мат. наук. — 1984. — № 1. — С. 51–53.
383. Черенин В.П. Решение некоторых комбинаторных задач оптимального планирования методом последовательных расчетов // Науч.-метод. материалы экон.-мат. семинара лаб. экон.-мат. методов АН СССР. — М.: ВЦ АН СССР. — 1962. — Вып. 2. — 44 с.
384. Чернова Г.В. Задача определения последовательности обработки деталей, имеющих одинаковые технологические маршруты // Применение математики в экономике. — Л. 1967. — Вып. 4. — С. 66–81.
385. Чудаков А.Д., Фалевич Б.Я. Автоматизированное оперативно-календарное планирование. — М.: Машиностроение, 1986. — 224 с.
386. Шапиро А.Д. Применение некоторых алгоритмов теории расписаний в календарном планировании индивидуального и мелкосерийного производства // Автоматизированные системы управления промышленными предприятиями. — Киев, 1975. — С. 21–34.
387. Шаров В.Г., Фалевич Б.Я., Зафранская Л.А. Алгоритмы моделирования пакетной передачи партий деталей на последующую обработку в задаче оперативно-календарного планирования // УС и М. — 1986. — № 5. — С. 123–125.
388. Шафранский В.В. Применение правил приоритета для оптимизации использования ограниченных ресурсов // Изв. АН СССР. Техн. кибернетика. — 1968. — № 5. — С. 62–74.
389. Шафранский В.В. Алгоритм расчета сетей по заданным правилам приоритета // Программный метод управления. — М.: ВЦ АН СССР. — 1971. — Вып. 1. — С. 59–94.
390. Шафранский В.В. Минимизация времени выполнения программы работ при заданных ограничениях на ресурсы // Программный метод управления. — М.: ВЦ АН СССР, 1973. — Вып. 2. — С. 63–99.
391. Шафранский В.В. Математические модели и методы планирования развития отраслей промышленности. — М.: Наука, 1984. — 224 с.
392. Шафранский Я.М. Оптимизация детерминированных систем обслуживания с древовидным частичным порядком // Изв. АН БССР. Сер. физ.-мат. наук. — 1978. — № 2. — С. 119.
393. Шафранский Я.М. Об оптимальном упорядочении в детерминированных системах с древовидным частичным порядком // Изв. АН БССР. Сер. физ.-мат. наук. — 1978. — № 2. — с. 120.
394. Шафранский Я.М. О задаче минимизации на множестве перестановок частично упорядоченных элементов. I, II // Изв. АН БССР. Сер. физ.-мат. наук. — 1980. — № 5. — С. 132; 1982. — № 1. — С. 113.
395. Шафранский Я.М. Об алгоритме отыскания минимума приоритето-порождающих функций на специальных множествах перестановок. I, II // Изв. АН БССР. Сер. физ.-мат. наук. — 1982. — № 3. — С. 38–42; 1983. — № 1. — С. 15–20.
396. Шахбазян К.В., Лебединская Н.Б. Минимизация суммарного штрафа в задаче параллельного упорядочения независимых работ // ДАН СССР. — 1977. — Т. 237, № 4. — С. 790–792.
397. Шевченко В.Н. Задача оптимального календарного планирования с ограничением на число рабочих // Изв. вузов. Радиофизика. — 1965. — Т. 8, № 3. — С. 635–637.
398. Шевченко В.Н. Задача о равномерном распределении простоев (несколько смен) // Экон. и мат. методы. — 1967. — Т. 3, вып. 4. — С. 619–623.
399. Шкурба В.В. Вычислительные схемы решения задач теории расписаний // Кибернетика. — 1965. — № 3. — С. 72–76.
400. Шкурба В.В. Применение метода статистических испытаний в задачах календарного планирования // Мат. методы в организации и экономике производства. — М.: 1966. — С. 209–223.
401. Шкурба В.В. О задачах упорядочения // Кибернетика. — 1967. — № 2. — С. 63–65.
402. Шкурба В.В. Задача трех станков. — М.: Наука, 1976. — 96 с.
403. Шкурба В.В., Белецкий С.А., Ефетова К.Ф. Планирование и управление в автоматизированном производстве. — Киев: Наукова думка, 1984. — 224 с.
404. Шкурба В.В., Подчасова Т.П., Пшичук А.Н., Тур Л.П. Задачи календарного планирования и методы их решения. — Киев: Наукова думка, 1966. — 155 с.

405. Шлепаков П.А., Зиглина А.Л. Алгоритмы и программы оперативно-календарного планирования производств нефтепереработки // Оперативное планирование и управление производством. — М.: Энергоатомиздат, 1985. — С. 27–31.
406. Шлепаков П.А., Зиглина А.Л., Соболев О.С. Пакет программ для календарного планирования производства // Механизация и автоматиз. производства. — 1984. — № 12. — С. 27–30.
407. Шмелев В.В. Метод составления расписания обработки одинаковых партий изделий // Автомат. и телемех. — 1973. — № 7. — С. 124–127.
408. Шмелев В.В. Динамическая задача межцехового планирования // Моделирование и управление в развивающихся системах. — М.: Наука, 1978. — С. 120–125.
409. Шурайц Ю.М. Минимизация времени ожидания комплексов работ при двухэтапном обслуживании // Автомат. и телемех. — 1977. — № 12. — С. 138–144.
410. Щербак А.Ф., Левин Г.Л., Волчек Б.А. Об одном классе задач оптимального временного распределения работ // Экон. и мат. методы. — 1974. — Т. 10, вып. 5. — С. 947–956.
411. Якимов Р.М. Об одном функциональном уравнении, описывающем оптимальную последовательность обработки изделий // Науч. труды. — Пермь: Перм. политехн. ин-т., 1960. — Вып. 21. — С. 62–71.
412. Achugbue J.O., Chin F.Y. Scheduling the open shop to minimize mean flow time// SIAM J. Comput. — 1982. — V.11, № 4. — P. 709–720.
413. Achugbue J.O., Chin F. Y. Complexity and solutions of some three-stage flow shop scheduling problems // Math. Oper. Res. — 1982. — V. 7, № 4. — P. 532–543.
414. Achuthan N.R. A special case of the $(n|m|F_{\max})$ problem// Opsearch. — 1977. — V.14, №2. — P. 71–87.
415. Achuthan N.R., Ghosh D.T. A note on heuristic rules for the $(n|3|F_{\max})$ problem// Opsearch. — 1980. — V. 17, №1. — P. 50–56.
416. Achuthan N.R., Grabowski J., Sidney J.B. Optimal flow-shop scheduling with earliness and tardiness penalties // Opsearch. — 1981. — V. 18. — P. 117–138.
417. Adam N.R., Surkis J. Priority update intervals and anomalies in dynamic ratio type job shop scheduling rules// Manag. Sci. — 1980. — V. 26, № 12. — P. 1227–1237.
418. Adiri I., Amit N. Route-depended open-shop scheduling // IIE Trans. — 1983. — V. 15, № 3. — P. 231–234.
419. Adiri I., Pohoryles D. Flowshop/no-idle or no-wait scheduling to minimize the sum of completion times// Nav. Res. Log. Quart. — 1978. — V.29, № 3. — P. 495–504.
420. Adrabinski A., Grabowski J. An algorithm for solving the machine sequencing problem// Zastosow. Matem. — 1977. — V. 15, № 4. — P. 536–540.
421. Aggarwal S.C. An example of Ignall and Schrage // Oper. Res. — 1973. — V. 21, № 3, — P. 862.
422. Akers S.B. A graphical approach to production scheduling problems// Oper. Res. — 1956. — V. 4, № 2. — P. 244–245.
423. Akers S.B., Friedman J. A non-numerical approach to production scheduling problems// Oper. Res. — 1955. — V. 3. — P. 429–442.
424. Albrycht J., Matloka M. On certain strict methods for solving sequential problems// Funct. et Approximatio. Comment. Math. — 1980. — V. 9. — P. 49–53.
425. Anderson E.J. A new continuous model for job-shop scheduling// Int. J. Syst. Sci.—1981.— V. 12, №12. — P. 1469–1475.
426. Arkin E.M., Papadimitriou C.H. On negative cycles in mixed graphs// Oper. Res. Lett.— 1985. — V. 4, №3. — P. 113–116.
427. Arora R.K., Rana S.P. Scheduling in a semi-ordered flow-shop without intermediate queues// AIIE Trans. — 1980. — V. 12, № 3. — P. 263–272.
428. Arthanari T.S., Mukhopadhyay A.C. A note on a paper by W. Szwarc// Nav. Res. Log. Quart. — 1971. — V. 18, № 1. — P. 135–138.
429. Ashour S. A decomposition approach for the machine scheduling problem// Int. J. Prod. Res. — 1967. — V. 6, № 2. — P. 109–122.
430. Ashour S. An experimental investigation and comparative evaluation of flow-shop scheduling techniques // Oper. Res. — 1970. — V. 18, № 3. — P. 541–549.
431. Ashour S. A branch-and-bound algorithm for flow shop scheduling problems// AIIE Trans. — 1970. — V. 2, № 2. — P. 172–176.
432. Ashour S. Sequencing theory// Lect. Notes Econ. Math. Syst. — 1972. — V. 69. — 133 p.

433. *Ashour S., Char A.R.* Computational experience on zero-one programming approach to various combinatorial problems// *J. Oper. Res. Soc. Japan.* — 1970. — V. 13, № 2. — P. 78–108.
434. *Ashour S., Chin K. -Y., Moore T.E.* An optimal schedule time of a job shop-like disjunctive graph// *Networks.* — 1973. — V. 3. — P. 333–349.
435. *Ashour S., Hiremath S.R.* A branch-and-bound approach to the job-shop scheduling problem// *Int. J. Prod. Res.* — 1973. — V. 11, № 1. — P. 47–58.
436. *Ashour S., Moore T.E., Chin K. -Y.* An implicit enumeration algorithm for nonpreemptive shop scheduling problem// *AIIE Trans.* — 1974. — V. 6, № 1. — P. 62–72.
437. *Ashour S., Parker R.G.* An out-of-kilter approach for machine sequencing problems// *Lect. Notes Econ. Math. Syst.* — 1973. — V. 86. — P. 206–223.
438. *Ashour S., Parker R.G.* A precedence graph algorithm for the shop scheduling problem// *Oper. Res. Quart.* — 1971. — V. 22, № 2. — P. 165–175.
439. *Ashour S., Quraishi N.* Investigation of various bounding procedures for production scheduling problems// *Int. J. Prod. Res.* — 1969. — V. 7, № 3. — P. 249–252.
440. *Axsäter S.* On scheduling in a semi-ordered flow shop without intermediate queues// *AIIE Trans.* — 1982. — V. 14, № 2. — P. 128–130.
441. *Bagga P.C.* Two machine scheduling problem// *Logist. Rev.* — 1967. — V. 3, № 1. — P. 27–37.
442. *Bagga P.C., Chakravarti N.K.* Optimal m -stage production schedule// *CORS J.* — 1968. — V. 6, № 2. — P. 71–78.
443. *Bagga P.C., Chakravarti N.K.* Minimizing waiting cost over two stage production schedule// *Z. angew. Math. Mech.* — 1969. — V. 49, № 5. — P. 299–301.
444. *Bagga P.C., Kalra K.P.* Node elimination in three-machine flowshop scheduling problem// *Indian J. Pure Appl. Math.* — 1982. — V. 13, № 8. — P. 871–878.
445. *Bagga P.C., Khurana K.* Transportation of job-blocks with separated setup times and removal times// *Cahiers C.E.R.O.* — 1984. — V. 26, № 3–4. — P. 195–200.
446. *Baker K.R.* Introduction to sequencing and scheduling. — New York: Wiley, 1974. — 305 p.
447. *Baker K.R.* A comparative study of flow-shop algorithms// *Oper. Res.* — 1975. — V. 23, № 1. — P. 62–73.
448. *Baker K.R.* An elimination method for the flow-shop problem// *Oper. Res.* — 1975. — V. 23, № 1. — P. 159–162.
449. *Balas E.* Discrete programming by the filter method// *Oper. Res.* — 1967. — V. 15, № 5. — P. 915–957.
450. *Balas E.* Machine-sequencing via disjunctive graphs: an implicit enumeration algorithm// *Oper. Res.* — 1969. — V. 17, № 6. — P. 941–957.
451. *Balas E.* Machine sequencing: disjunctive graphs and degree-constrained subgraphs// *Nav. Res. Log. Quart.* — 1970. — V. 17, № 1. — P. 1–9.
452. *Balas E.* Project scheduling with resource constraints// *Applications of Mathem. Programming.* — London: English University Press, 1971. — P. 187–200.
453. *Balas E.* On the facial structure of scheduling polyhedra// *Math. Progr. Study.* — 1985. — V. 24. — P. 179–218.
454. *Balas E., Guignard M.* Report of the session on branch and bound/implicit enumeration// *Ann. Discr. Math.* — 1979. — V. 5. — P. 185–191.
455. *Balas E., Landweer P.R.* Traffic assignment in communication satellites// *Oper. Res. Lett.* — 1983. — V. 2, № 4. — P. 141–147.
456. *Bansal S.P.* Minimizing the sum of completion times of n jobs over m machines in a flowshop. — A branch and bound approach// *AIIE Trans.* — 1977. — V. 9, № 3. — P. 306–315.
457. *Bansal S.P.* On lower bounds in permutation flowshop problems// *Int. J. Prod. Res.* — 1979. — V. 17, № 4. — P. 411–418.
458. *Bansal S.P., Kuan D.* Global lower bound in the flow-shop problem// *Opsearch.* — 1976. — V. 13, № 2. — P. 115–118.
459. *Baran-Jarosz B.* Metoda rozwiązywania pewnego zagadnienia sekwencyjnego// *Prz. statyst.* — 1973. — T. 20, № 2. — S. 181–192.
460. *Barany I.* A vector-sum theorem and its application to improving flow shop guarantees// *Math. Oper. Res.* — 1981. — V. 6, № 3. — P. 445–452.
461. *Barany I., Fiala T.* Többgépes ütemelési problémák közel optimalis megoldása// *Sigma.* — 1982. — V. 15, № 3. — P. 177–191.

462. *Barany I., Grinberg V.S.* A vector-sum theorem in two-dimensional space// *Period. Math. Hung.* — 1985. — V. 16, № 2. — P. 135–138.
463. *Barker J.R., McMahon G.B.* Scheduling in general job shop// *Manag. Sci.* — 1985. — V. 31, № 5. — P. 594–598.
464. *Bellman R.* Mathematical aspects of scheduling theory// *J. SIAM.* — 1956. — V. 4, № 3. — P. 168–205.
465. *Bellman R., Gross O.* Some combinatorial problems arising in the theory of multistage processes// *J. SIAM.* — 1954. — V. 2, № 3. — P. 175–183.
466. *Bestwick P.F., Hastings N.A.J.* A new bound for machine scheduling// *Oper. Res. Quart.* — 1976. — V. 27, № 2ii. — P. 479–487.
467. *Blackstone J.N., Phillips D.T., Hogg C.L.* A state-of-the-art survey of dispatching rules for manufacturing job shop operations// *Int. J. Prod. Res.* — 1982. — V. 20, № 1. — P. 27–45.
468. *Blazewicz J.* Deadline scheduling of tasks — a survey// *Found. Control Engin.* — 1977. — V. 1, № 4. — P. 203–216.
469. *Blazewicz J.* Solving the resource constrained deadline scheduling via reduction to the network flow problem// *Eur. J. Oper. Res.* — 1981. — V. 6. — P. 75–79.
470. *Blazewicz J., Lenstra J.K., Rinnooy Kan A.H.G.* Scheduling subject to resource constraints: classification and complexity// *Discr. Appl. Math.* — 1983. — V. 5. — P. 11–24.
471. *Blazewicz J., Weglarz J.* Złożoność obliczeniowa problemów szeregowania zadań na maszynach z ograniczeniami zasobowymi// *Archiv. Automat. Telemekh.* — 1984. — T. 29, № 1–2. — S. 3–33.
472. *Bongiovanni G., Coopersmith D., Wong C.K.* An optimal time slot assignment algorithm for an SS/TDMA system with variable number of transponders// *IEEE Trans. Commun.* — 1981. — V. COM-29, № 5. — P. 721–726.
473. *Bonney M.C., Gundry S.W.* Solutions to the constrained flowshop sequencing problem// *Oper. Res. Quart.* — 1976. — V. 27, № 41. — P. 869–883.
474. *Borowitz I., Ein-dor P.* Two issues in the evaluation of job shop schedules// *OMEGA.* — 1976. — V. 4, № 3. — P. 313–320.
475. *Bowman E.H.* The schedule-sequencing problem// *Oper. Res.* — 1959. — V. 7, № 5. — P. 621–624.
476. *Bozoki G., Richard J.-P.* A branch-and-bound algorithm for the continuous-process job-shop scheduling problem// *AIIE Trans.* — 1970. — V. 11, № 3. — P. 246–252.
477. *Bratley P., Florian M., Robillard P.* On sequencing with earliest starts and due dates with application to computing bounds for the $(n|m|G|F_{\max})$ problem// *Nav. Res. Log. Quart.* — 1973. — V. 20. — P. 57–67.
478. *Brooks G.H., White C.R.* An algorithm for finding optimal or near optimal solutions to the production scheduling problem// *J. Indust. Engng.* — 1965. — V. 16. — P. 34–40.
479. *Brown A.P.G., Lomnicki Z.A.* Some applications of the “branch-and-bound” algorithm to the machine scheduling problem// *Oper. Res. Quart.* — 1966. — V. 17, № 2. — P. 173–186.
480. *Brucker P.* NP-complete operation research problems and approximation algorithms// *Z. Oper. Res.* — 1979. — Bd. 23, № 3. — S. 73–94.
481. *Brucker P.* Minimizing maximum lateness in a two-machine unit-time job shop// *Computing.* — 1981. — V. 27. — P. 367–370.
482. *Brucker P.* A linear time algorithm to minimize maximum lateness for the two-machine unit-time, job-shop, scheduling problem// *Lect. Notes Control Inform. Sci.* — 1982. — V. 38. — P. 566–571.
483. *Burns F., Rooker J.* A special case of $3 \times n$ flow shop problem// *Nav. Res. Log. Quart.* — 1975. — V. 22, № 4. — P. 811–817.
484. *Burns F., Rooker J.* Johnson's three-machines flow-shop conjecture// *Oper. Res.* — 1976. — V. 24, № 3. — P. 578–580.
485. *Burns F., Rooker J.* Three-stage flow-shop with recessive second stage// *Oper. Res.* — 1978. — V. 26, № 1. — P. 207–208.
486. *Campbell H.G., Dudek R.A., Smith M.L.* A heuristic algorithm for the n job, m machine sequencing problem// *Manag. Sci.* — 1970. — V. 16, № 10. — P. B630–B637.
487. *Carlier J.* Disjonctions dans ordonnancements// *RAIRO. Rech. Oper.* — 1975. — V. 2, № 1–2. — P. 83–100.

488. *Carlier J.* Ordonnancements a contrained disjonctives// *RAIRO. Rech. Oper.* – 1978. – V. 12, №4. – P. 333–351.
489. *Chadrasekaran R.* Recognition of the Gilmore – Gomory travelling salesman problem// *Discr. Appl. Math.* – 1986. – V. 14, № 3. – P. 231–238.
490. *Charlton J.M., Death C.C.* A generalized machine scheduling algorithm// *Oper. Res. Quart.* – 1970. – V. 21, № 1. – P. 127–134.
491. *Charlton J.M., Death C.C.* A method of solution for general machine scheduling problems// *Oper. Res.* – 1970. – V. 18, № 4. – P. 689–707.
492. *Chin F.Y., Tsai L.-L.* On J -maximal and J -minimal flow-shop schedules// *J. Assoc. Comput. Mach.* – 1981. – V. 28, № 3. – P. 462–476.
493. *Cho Y., Sahni S.* Preemptive scheduling of independent jobs with release and due times on open, flow and job shops// *Oper. Res.* – 1981. – V. 29, № 3. – P. 511–522.
494. *Ciobanu G.* A general algorithm for solving flow shop scheduling// *Econ. comput. econ. cybern. stud. res.* – 1975. – № 3. – P. 207–212.
495. *Corwin B.D., Esogbue A.D.* Two machine flow shop scheduling problems with sequence dependence setup times: a dynamic programming approach// *Nav. Res. Log. Quart.* – 1974. – V. 21, № 3. – P. 515–524.
496. *Dannenbring D.G.* An evaluation of flow shop sequencing heuristics// *Manag. Sci.* – 1977. – V. 23, № 11. – P. 1174–1182.
497. *Dantzig G.B.* A machine-job scheduling model// *Manag. Sci.* – 1960. – V. 6, № 2. – P. 191–196.
498. *Das D., Dowsland W.B.* A model to examine the effect of loading conditions on the scheduling rules in job shop environment// *Int. J. Prod. Res.* – 1981. – V. 19, № 5. – P. 577–587.
499. *Davis E.W., Heidorn G.E.* An algorithm for optimal projects scheduling under multiple resource constraints// *Manag. Sci.* – 1971. – V. 17, № 12. – P. 803–816.
500. *Davis M., Putnam H.* A computing procedure for quantification theory// *J. Assoc. Comput. Mach.* – 1960. – P. 201–215.
501. *Day J.E., Hottenstein M.P.* Review of sequencing research// *Nav. Res. Log. Quart.* – 1970. – V. 17, № 1. – P. 11–39.
502. *Dekel E., Sahni S.* Parallel scheduling algorithms// *Oper. Res.* – 1983. – V. 31, № 1. – P. 24–49.
503. *Deman J.M., van, Baker K.R.* Minimizing mean flow time in the flow shop with no intermediate queues// *AIIE Trans.* – 1974. – V. 6, № 1. – P. 28–34.
504. *Deterministic and stochastic scheduling/ Eds Dempster M.A.H., Lenstra J.K., Rinnooy Kan A.H.G. – Reidel: Dordrecht. – 1982. – 419 p.*
505. *Dewess G.* Die Methode der markierten Verschiebung – ein neues Verfahren zur Optimierung ressourcenbeschränkter Netzplanabläufe// *Math. Operationsforsch. Statist. Ser. Optimization.* – 1983. – V. 14, № 2. – P. 217–235.
506. *Dijkstra E.W.* A note on two problems in connection with graphs// *Numer. Math.* – 1959. – № 1. – P. 269.
507. *Dudek R.A., Smith M.L., Panwalkar S.S.* Use of a case study in sequencing/ scheduling research// *OMEGA*. – 1974. – V. 2. – P. 253–261.
508. *Dudek R.A., Teuton O.F.* Development of M -stage decision rule for the scheduling n jobs through m machines// *Oper. Res.* – 1964. – V. 12, № 3. – P. 471–497.
509. *Durand A.* Une methode optimale de traitement des contraintes disjonctives dans les problems d'ordonnancement// *Revue Franc. Inform. Rech. Oper.* – 1967. – № 3. – P. 49–61.
510. *Dutta S.K., Cunningham A.A.* Sequencing two-machine flow-shops with finite intermediate storage// *Manag. Sci.* – 1975. – V. 21, № 9. – P. 989–996.
511. *Eilon S., Chowdhury I.G.* Conditions for optimal scheduling for the three-machine problem// *J. Inst. Math. Appl.* – 1976. – V. 17, № 1. – P. 1–4.
512. *Eilon S., Chowdhury I.G.* Due dates in job shop scheduling// *Int. J. Prod. Res.* – 1976. – V. 14, № 2. – P. 223–237.
513. *Ellman D., Townsend W.* A dynamic programming approach to the three-machine scheduling// *J. Inst. Math. Appl.* – 1974. – V. 13, № 2. – P. 161–167.
514. *Elmagiraby S.E.* The machine sequencing problem – review and extensions// *Nav. Res. Log. Quart.* – 1968. – V. 15, № 2. – P. 205–232.
515. *Elvers D.A.* The sensivity of the relative effectiveness of job shop dispatching rules with respect to various arrival distribution// *AIIE Trans.* – 1974. – V. 6, № 1. – P. 41–49.

516. *Elvers D.A., Taube L.R.* Deterministic/stochastic assumptions in job shops// *Eur. J. Oper. Res.* – 1983. – V. 14, № 1. – P. 89–94.
517. *Erschler J., Roubellat F., Vernhes J.P.* Finding some essential characteristics of the feasible solutions of a scheduling problem// *Oper. Res.* – 1976. – V. 24, № 4. – P. 774–783.
518. *Even S., Itai A., Shamir A.* On the complexity of timetable and multicommodity flow problems// *SIAM J. Comput.* – 1976. – V. 5. – P. 691–703.
519. *Fiala T.* Közelítő algoritmus a három gép problémára// *Alkalmazott. mat. lapok.* – 1977. – V. 3, № 3. – P. 389–398.
520. *Fiala T.* An algorithm for the open-shop problem// *Math. Oper. Res.* – 1983. – V. 8, № 1. – P. 100–109.
521. *Fiorini S.* A bibliographie survey of edge-colorings// *J. Graph. Theor.* – 1978. – V. 2. – P. 93–106.
522. *Fisher M.L.* Optimal solution of scheduling problems using Lagrange multipliers, part 1// *Oper. Res.* – 1973. – V. 21, № 5. – P. 1114–1127.
523. *Fisher M.L.* Worst-case analysis of heuristic algorithms// *Manag. Sci.* – 1980. – V. 26, № 1. – P. 1–17.
524. *Fisher M.L., Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.* Surrogate duality relaxation for job shop scheduling// *Discr. Appl. Math.* – 1983. – V. 5. – P. 65–75.
525. *Florian M., Tilquin C., Tilquin G.* An implicit enumeration algorithm for complex scheduling problems// *Int. J. Prod. Res.* – 1975. – V. 13, № 1. – P. 25–40.
526. *Florian M., Trepant P., McMahon G.* An implicit enumeration algorithm for the machine sequencing problem// *Manag. Sci. (B)*. – 1971. – V. 17, № 12. – P. 782–792.
527. *Floyd R.W.* Algorithm 97, shortest path// *Comm. ACM.* – 1962. – № 5. – P. 345.
528. *French S.* Sequencing and Scheduling: an introduction to the mathematics of the job-shop. – Horwood: Chichester, 1982. – 245 p.
529. *Gabow H.N., Kariv O.* Algorithms for edge coloring bipartite graphs and multigraphs// *SIAM J. Comput.* – 1982. – V. 11, № 1. – P. 117–129.
530. *Garey M.R., Graham R.L., Johnson D.S., Yao A.C.-C.* Resource constrained scheduling as generalized bin packing// *J. Combin. Theory (A)*. – 1976. – V. 21, № 3. – P. 257–298.
531. *Garey M.R., Johnson D.S.* Complexity results for multiprocessor scheduling under resource constraints// *SIAM J. Comput.* – 1975. – V. 4, № 4. – P. 397–411.
532. *Garey M.R., Johnson D.S.* "Strong" NP-completeness results: motivation, examples and implications// *J. Assoc. Comput. Mach.* – 1978. – V. 25, № 3. – P. 499–508.
533. *Garey M.R., Johnson D.S., Sethi R.* The complexity of flow-shop and jobshop scheduling// *Math. Oper. Res.* – 1976. – V. 1, № 2. – P. 117–129.
534. *Gelders L.F., Sambandam N.* Four simple heuristics for scheduling a flow-shop// *Int. J. Prod. Res.* – 1978. – V. 16, № 3. – P. 221–231.
535. *Gelders L.F., Wassenhove L.N.V.* Production planning: a review// *Eur. J. Oper. Res.* – 1981. – № 7. – P. 101–110.
536. *Geoffrion A.M., Marsten R.E.* Integer programming algorithm: a framework and state-of-the art survey// *Manag. Sci.* – 1972. – V. 18, № 9. – P. 465–491.
537. *Gere W.S.* Heuristics in job shop scheduling// *Manag. Sci.* – 1966. – V. 13, № 3. – P. 167–190.
538. *Gorenstein S.* An algorithm for project (job) sequencing with resource constraints// *Oper. Res.* – 1972. – V. 20, № 4. – P. 835–850.
539. *Giffler B.* Schedule algebra: a progress report// *Nav. Res. Log. Quart.* – 1968. – V. 15, № 2. – P. 255–280.
540. *Giffler B., Thompson G.L.* Algorithms for solving production-scheduling problems// *Oper. Res.* – 1960. – V. 8, № 4. – P. 487–503.
541. *Giglio R., Wagner H.* Approximate solutions to the three-machine scheduling problem// *Oper. Res.* – 1964. – V. 12, № 2. – P. 305–324.
542. *Gilmore P.C.* A solvable case of the traveling salesman problem// *Canad. Math. Bull.* – 1966. – V. 9, № 6. – P. 743–744.
543. *Gilmore P.C., Gomory R.E.* A solvable case of the traveling salesman problem// *Proc. Nat. Acad. Sci. USA.* – 1964. – V. 51, № 2. – P. 178–181.
544. *Gilmore P.C., Gomory R.E.* Sequencing a one-state variable machine: a solvable case of the traveling salesman problem // *Oper. Res.* – 1964. – V. 12, № 5. – P. 655–679.

545. *Gilmore P.C., Lawler E.L., Shmoys D.B.* Well-solved special cases//The traveling salesman problem. A guided tour of combinatorial optimization. — Chichester: Wiley, 1985.
546. *Godin V.B.* Interactive scheduling: historical survey and state of the art//AIIE Trans. — 1978. — V. 10, № 3. — P. 331–337.
547. *Gondran M., Minoux M.* Graphs and algorithms. — Chichester: Wiley, 1984. — 650 p.
548. *Gonzalez M.J.* Deterministic processor scheduling//Computing Surveys. — 1977. — V. 9, № 3. — P. 173–204.
549. *Gonzalez T.* A note on open shop preemptive schedules//IEEE Trans. Comput. — 1979. — V. 28, № 10. — P. 782–786.
550. *Gonzalez T.* Unit execution time shop problems//Math. Oper. Res. — 1982. — V. 7, № 1. — P. 57–66.
551. *Gonzalez T., Sahni S.* Open shop scheduling to minimize finish time//J. Assoc. Comput. Mach. — 1976. — V. 23, № 4. — P. 665–679.
552. *Gonzalez T., Sahni S.* Flowshop and jobshop schedules: complexity and approximation//Oper. Res. — 1978. — V. 26, № 1. — P. 36–52.
553. *Goodwin J.C., Elvers D.A., Goodwin J.S.* Overtime usage in a job shop environment//OMEGA. — 1978. — V. 6, № 6. — P. 494–500.
554. *Gopal I.S., Bongiovanni G., Bonuccelli M.A., Tang D.T., Wong C.K.* An optimal switching algorithm for multibeam satellite systems with variable band with beams//IEEE Trans. Commun. — 1982. — V. COM-30, № 11. — P. 2475–2481.
555. *Gopal I.S., Bonuccelli M.A., Wong C.K.* Scheduling in multibeam satellites interfering zones//IEEE Trans. Commun. — 1983. — V. COM-31, № 8. — P. 941–951.
556. *Gopal I.S., Wong C.K.* Minimizing the number of switchings in a SS/TDMA system//IEEE Trans. Commun. — 1985. — V. COM-33, № 6. — P. 497–501.
557. *Goyal S.K.* A note on a paper: On the flow-shop sequencing problem with no wait in process//Oper. Res. Quart. — 1973. — V. 24, № 1. — P. 130–133.
558. *Goyal S.K.* Job-shop sequencing problem with no wait in process//Int. J. Prod. Res. — 1975. — V. 13, № 2. — P. 197–206.
559. *Grabowski J.* A new formulation and solution of the sequencing problem: mathematical model//Zastosow. Matem. — 1976. — T. 15, № 4. — S. 325–343.
560. *Grabowski J.* A new formulation and solution of the sequencing problem: algorithm//Zastosow. Matem. — 1976. — T. 15, № 4. — S. 463–474.
561. *Grabowski J.* Formulation and solution of the sequencing problem with parallel machines//Zastosow. Matem. — 1978. — T. 16, № 2. — S. 215–292.
562. *Grabowski J.* On two-machine scheduling with release and due dates to minimize maximum lateness//Opsearch. — 1980. — V. 17, № 4. — P. 133–154.
563. *Grabowski J.* A new algorithm of solving the flow-shop problem//Operations research in progress. Theory and decision library. — Dordrecht-Boston-London, 1982. — V. 32. — P. 57–75.
564. *Grabowski J.* A note on an algorithm for solving the $n|2|F, r_i \geq 0|L_{\max}$ problem//Opsearch. — 1983. — V. 20, № 2. — P. 107–112.
565. *Grabowski J., Janiak A.* On job-shop scheduling with resources constraints//Lect. Notes Contr. Inf. Sci. — 1984. — V. 59. — P. 193–200.
566. *Grabowski J., Skubalska E., Smutnicki C.* On flow shop scheduling with release and due dates to minimize maximum lateness//J. Oper. Res. Soc. — 1983. — V. 34, № 7. — P. 615–620.
567. *Grabowski J., Syslo M.M.* On a machine sequencing problem (I)//Zastosow Matem. — 1973. — T. 13, № 3. — S. 339–345.
568. *Graham R.L.* The combinational mathematics of scheduling//Scientific American. — 1978. — V. 238. — P. 124–132.
569. *Graham R.L., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Optimization and approximation in deterministic sequencing and scheduling: a survey//Ann. Discrete Math. — 1979. — V. 5. — P. 287–326.
570. *Graves S.C.* A review of production scheduling//Oper. Res. — 1981. — V. 29, № 4. — P. 646–675.
571. *Greenberg H.H.* A branch-and-bound solution to the general scheduling problem//Oper. Res. — 1968. — V. 16, № 2. — P. 353–361.
572. *Gupta S.K.* n jobs and m machines, job-shop problems with sequence-dependent set-up times//Int. J. Prod. Res. — 1982. — V. 20, № 5. — P. 643–656.

573. Gupta J.N.D. A general algorithm for the $n \times M$ flow-shop scheduling problem//Int. J. Prod. Res. – 1969. – V. 7, № 3. – P. 241–247.
574. Gupta J.N.D. M -stage flow-shop scheduling by branch and bound//Opsearch. – 1970. – V. 7, № 1. – P. 37–43.
575. Gupta J.N.D. Economic aspects of production scheduling systems//J. Oper. Res. Soc. Japan. – 1971. – V. 13. – P. 169–193.
576. Gupta J.N.D. M -stage scheduling problem – a critical appraisal//Int. J. Prod. Res. – 1971. – V. 9, № 2. – P. 267–281.
577. Gupta J.N.D. The generalized n job, m machine scheduling problem//Opsearch. – 1971. – V. 8. – P. 173–185.
578. Gupta J.N.D. An improved combinatorial algorithm for the flow-shop scheduling problem//Oper. Res. – 1971. – V. 19, № 7. – P. 1753–1759.
579. Gupta J.N.D. A functional heuristic algorithm for the flowshop scheduling problem//Oper. Res. Quart. – 1971. – V. 22, № 1. – P. 39–47.
580. Gupta J.N.D. Optimal scheduling in a multistage flowshop//AIIE Trans. – 1972. – V. 4, № 3. – P. 238–243.
581. Gupta J.N.D. Heuristic algorithm for multistage flowshop scheduling problem//AIIE Trans. – 1972. – V. 4, № 1. – P. 11–18.
582. Gupta J.N.D. Analysis of combinatorial approach to flowshop scheduling problems//Oper. Res. Quart. – 1975. – V. 26, № 2ii. – P. 431–440.
583. Gupta J.N.D. Optimal schedules for special structures flowshops//Nav. Res. Log. Quart. – 1975. – V. 22, № 2. – P. 255–269.
584. Gupta J.N.D. A heuristic algorithm for the flowshop scheduling problem//Rev. Franc. Automat., Inform., Rech. Opér. – 1976. – V. 10, № 6. – P. 63–73.
585. Gupta J.N.D. Errata//Oper. Res. – 1976. – V. 24, № 4. – P. 796.
586. Gupta J.N.D. Optimal flowshop schedules with no intermediate storage space//Nav. Res. Log. Quart. – 1976. – V. 23, № 2. – P. 235–243.
587. Gupta J.N.D. An improved lexicographic search algorithm for the flowshop scheduling problem//Comput. Oper. Res. – 1979. – V. 6, № 2. – P. 117–120.
588. Gupta J.N.D., Maykut A.R. Flow-shop scheduling by heuristic decomposition//Int. J. Prod. Res. – 1973. – V. 11, № 2. – P. 102–105.
589. Gupta J.N.D., Maykut A.R. Heuristic algorithms for scheduling n jobs in a flowshop//J. Oper. Res. Soc. Japan. – 1973. – V. 16, № 3. – P. 131–150.
590. Gupta J.N.D., Reddi S.S. Improved dominance condition for the three-machine flow-shop scheduling problem//Oper. Res. – 1978. – V. 26, № 1. – P. 200–203.
591. Haider S.W., Moodie C.L., Buck J.R. An investigation of the advantages of using a man-computer interactive scheduling methodology for job shops//Int. J. Prod. Res. – 1981. – V. 19, № 4. – P. 381–392.
592. Hardgrave W.W., Nemhauser G. A geometric model and graphical algorithm for a sequencing problem//Oper. Res. – 1963. – V. 11, № 6. – P. 889–900.
593. Hariri A.M.A., Potts C.N. Algorithms for two-machine flow-shop sequencing with precedence constraints//Europ. J. Oper. Res. – 1984. – V. 17, № 2. – P. 238–248.
594. Hartwig A., Terno J. Transformation von Reihenfolgeproblemen auf Rundreiseprobleme//Dresden: Technische Universität Dresden, 1978. – 12 p.
595. Hefetz N., Adiri I. An efficient optimal algorithm for the two-machines, unit-time, job-shop, schedule-length, problem//Math. Oper. Res. – 1982. – V. 7, № 3. – P. 354–360.
596. Hefetz N., Adiri I. A note on the influence of missing operations in scheduling problems//Nav. Res. Log. Quart. – 1982. – V. 29, № 3. – P. 535–539.
597. Heller J. Some numerical experiments for $M \times J$ flow shop and its decision theoretical aspects//Oper. Res. – 1960. – V. 8, № 2.
598. Heller J., Logemann G. An algorithm for construction and evaluation of feasible scheduling//Manag. Sci. – 1962. – V. 8, № 2. – P. 168–183.
599. Hershaner J.C., Evert R.J. Search and simulation of a job shop sequencing rule//Manag. Sci. – 1975. – V. 21, № 7. – P. 833–843.
600. Holloway C.A., Nelson R.T. Job shop scheduling with due dates and overtime capability//Manag. Sci. – 1974. – V. 21, № 1. – P. 68–78.
601. Ibaraki T. Computational efficiency of approximate branch and bound algorithms//Math. Oper. Res. – 1976. – V. 1. – P. 287–298.

602. *Ibaraki T.* On the computational efficiency of branch-and-bound algorithms//J. Oper. Res. Soc. Japan. — 1977. — V. 20. — P. 16–35.
603. *Ibaraki T.* Branch-and-bound procedure and state — space representation of combinatorial optimization problems//Inform. and Contr. — 1978. — V. 36, № 1. — P. 1–27.
604. *Ignall E., Schrage L.* Application of the branch and bound technique to some flow-shop scheduling problems//Oper. Res. — 1965. — V. 13, № 3. — P. 400–412.
605. Integer programming and related areas. A classified bibliography//Lect. Notes Econ. Math. Syst. — 1976. — V. 128; 1978. — V. 160; 1982. — V. 197.
606. *Inukai T.* An efficient SS/TDMA time slot assignment algorithm//IEEE Trans. Commun. — 1979. — V. COM-27, № 10. — P. 1449–1455.
607. *Jackson J.R.* Job shop-like queueing systems//Manag. Sci. — 1963. — № 10. — P. 131–142.
608. *Jackson J.R.* An extension of Johnson's results on job lot scheduling//Nav. Res. Log. Quart. — 1956. — V. 3, № 3. — P. 201–203.
609. *Jaeschke G.* Das Reihenfolgeproblem für Erzeugnisse mit gleichem Ablaufplan//Elektron. Rechenanlag. — 1964. — Jg. 6, № 3. — S. 137–144.
610. *Jain S.K., Scott K.L., Vasold E.G.* Orderbook balancing using a combination of LP and heuristic techniques//Interfaces. — 1978. — V. 9, № 1. — P. 55–67.
611. *Janiak A., Grabowski J.* Job-shop problem with resource constraints//Large Scale Syst.: Theory and Appl. Proc. IFAC/IFORS Symp.— Oxford, 1984. — P. 475–480.
612. *Johnson S.M.* Optimal two and three stage production schedules with set up times included//Nav. Res. Log. Quart. — 1954. — V. 1, № 1. — P. 61–68. (Имеются переводы: Джонсон С.М. Оптимальные двух- и трехоперационные календарные планы производства с учетом подготовительно-заключительного времени//Календарное планирование. — М.: Прогресс, 1966. — С. 33–41; Джонсон С.М. Оптимальное расписание для двух- и трехступенчатых процессов с учетом времени наладки//Киберн. сб. — М.: Мир, 1965. — Вып. 1. — С. 78–86).
613. *Johnson S.M.* Discussion: sequencing n jobs on two machines with arbitrary time lags//Manag. Sci. — 1959. — V. 5, № 3. — P. 299–303.
614. *Jones C.H.* An economic evaluation of job shop dispatching rules//Manag. Sci. — 1973. — V. 20, № 3. — P. 293–301.
615. *Kalra K.R., Bansal S.P., Bagga P.C.* Minimization of weighted sum of completion times with due dates in permutation flowshop problems//J. Comb. Inform. Syst. Sci. — 1979. — V. 4, № 2. — P. 169–177.
616. *Karp R.M.* On the computational complexity of combinatorial problems//Networks. — 1975. — V. 5, № 1. — P. 45–68.
617. *Karuch W.* A counterexample to a proposed algorithm for optimal sequencing of jobs//Oper. Res. — 1965. — V. 13, № 2. — P. 323–325.
618. *Khurana K., Bagga P.C.* Ordered flowshop problem with setup considerations//Cahiers C.E.R.O. — 1984. — V. 26, № 1–2. — P. 65–76.
619. *Khurana K., Bagga P.C.* Minimizing the makespan in a 2-machine flowshop with time lags and setup conditions//Z. Oper. Res. Ser. A. — 1984. — V. 28, № 2. — P. 163–174.
620. *Khurana K., Bagga P.C.* Scheduling of job-block with dead line in $n \times 2$ flow shop problem with separated set-up times//Ind. J. Pure Appl. Math. — 1985. — V. 16, № 3. — P. 213–224.
621. *King J. P.* Scheduling and the problem of computational complexity//OMEGA. — 1979. — V. 7, № 3. — P. 233–240.
622. *King J. R., Spachis A.S.* Scheduling: bibliography and review//Int. J. Physical Distribution and Materials Manag.— 1980. — V. 10, № 3. — P. 100–132.
623. *Kise H.* Scheduling and branch-and-bound method//Syst. and Contr. — 1973. — V. 17, № 5. — P. 275–281.
624. *Kobayashi T.* A method for counting the number of feasible subsets of a partially ordered finite set//J. Oper. Res. Soc. Japan. — 1966. — V. 8, № 4. — P. 155–171.
625. *Kohler W.H., Steiglitz K.* Characterization and theoretical comparison of branch-and-bound algorithms for permutational problems//J. Assoc. Comput. Mach. — 1974. — V. 21, № 1. — P. 140–156.
626. *Kohler W.H., Steiglitz K.* Exact, approximate and guaranteed accuracy algorithms for the flow shop problem $n|2|F|F$ //J. Assoc. Comput. Mach. — 1975. — V. 22, № 1. — P. 106–114.

627. *Krarup J., Werra D.*, de. Chromatic optimization: limitations, objectives, uses, references//Eur. J. Oper. Res. — 1982. — V. 11, № 1. — P. 1–19.
628. *Krone M.* Heuristic programming applied to scheduling models//Proc. 5th Annu. Princeton Conf. Inform. Sci. and Syst. — Princeton, 1971. — P. 193–196.
629. *Krone M.J., Steiglitz K.* Heuristic programming solution of a flowshop-scheduling problem//Oper. Res. — 1974. — V. 22, № 3. — P. 629–637.
630. *Kurisu T.* Flow shop sequencing problem with time lags//Technol. Repts Osaka Univ. — 1973. — № 22. — P. 303–314.
631. *Kurisu T.* Two-machine sequencing problem with exponential processing times//Tech. Repts. Osaka Univ. — 1973. — № 23. — P. 1–8.
632. *Kurisu T.* Two-machine scheduling under required precedence among jobs//J. Oper. Res. Soc. Japan. — 1976. — V. 19, № 1. — P. 1–13.
633. *Kurisu T.* Two-machine scheduling under arbitrary precedence constraints//J. Oper. Res. Soc. Japan. — 1977. — V. 20, № 2. — P. 113–131.
634. *Kurisu T.* Three-machine scheduling problem with precedence constraints//J. Oper. Res. Soc. Japan. — 1977. — V. 20, № 3. — P. 231–242.
635. *Lageweg B.J., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Computer aided complexity classification of deterministic scheduling problems//Preprint/Mathematische Centrum. — Amsterdam, 1981. — Afdeling mathematische Besliskunde BW 138/81. — 20 p.
636. *Lageweg B.J., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Computer aided complexity classification of combinatorial problems//Comm. ACM. — 1982. — V. 25, № 11. — P. 817–822.
637. *Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.* Minimizing maximum lateness on one machine: computational experience and some applications//Statist. Neerlandia. — 1976. — V. 30. — P. 25–41.
638. *Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.* Job-shop scheduling by implicit enumeration//Manag. Sci. — 1977. — V. 24, № 4. — P. 441–450.
639. *Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.* A general bounding scheme for the permutation flow-shop problem//Oper. Res. — 1978. — V. 26, № 1. — P. 53–62.
640. *Land A.H., Doig A.G.* An automatic method of solving discrete programming problems//Econometrica. — 1960. — V. 28, № 3. — P. 497–520.
641. *Lanzener C.H., Himes R.S.* A linear programming solution to the general sequencing problem//CORS J. — 1970. — V. 8, № 2. — P. 129–134.
642. *Lasdon L.S., Terjung R.S.* An efficient algorithm for multi-item scheduling//Oper. Res. — 1971. — V. 19, № 4. — P. 946–969.
643. *Lawler E.L.* Recent results in the theory of machine scheduling//Math. Progr. State Art: 11th Int. Symp. — Bonn, 1982. — P. 202–234.
644. *Lawler E.L., Labetoulle J.* On preemptive scheduling on unrelated parallel processors by linear programming//J. Assoc. Comput. Mach. — 1978. — V. 25, № 4. — P. 617–619.
645. *Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Minimizing maximum lateness in a two-machine open shop//Math. Oper. Res. — 1981. — V. 6, № 1. — P. 153–158.
646. *Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Recent developments in deterministic sequencing and scheduling: a survey//Deterministic and Stochastic scheduling. — Reidel: Dordrecht, 1982. — P. 35–73.
647. *Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.* Erratum//Math. Oper. Res. — 1982. — V. 7. — P. 635.
648. *Lawler E.L., Luby M.G., Vazirani V.V.* Scheduling open shops with parallel machines//Oper. Res. Letters. — 1982. — V. 1. — P. 161–164.
649. *Lawler E.L., Wood D.E.* Branch-and-bound methods: a survey//Oper. Res. — 1966. — V. 14, № 4. — P. 699–719.
650. *Lefeman L.J.* Some graphtheoretical problems of network analysis under resource constraints//Oper. Res. Verfahren. — 1979. — V. 30. — P. 194–205.
651. *Lenstra J.K.* Sequencing by enumerative methods. — Amsterdam: Math. Center Tracts 69, 1977. — 202 p.
652. *Lenstra J.K., Rinnooy Kan A.H.G.* Computational complexity of discrete optimization problems//Ann. Disc. Math. — 1979. — V. 4. — P. 121–140.
653. *Lenstra J.K., Rinnooy Kan A.H.G.* Complexity results for scheduling chains on a single machine//Eur. J. Oper. Res. — 1980. — V. 4, № 4. — P. 270–275.

654. *Lenstra J.K., Rinnooy Kan A.H.G.* Complexity of vehicle routing and scheduling problems//Networks. — 1981. — V. 11. — P. 221–227.
655. *Lenstra J.K., Rinnooy Kan A.H.G.* Scheduling theory since 1981: an annotated bibliography: Preprint/Mathematische Centrum. — Amsterdam, 1983. — Afdeling mathematische Beliskunde BW 188/83.
656. *Lenstra J.K., Rinnooy Kan A.H.G.* New directions in scheduling theory//Oper. Res. Lett. — 1984. — V. 2, № 6. — P. 255–259.
657. *Lenstra J.K., Rinnooy Kan A.H.G., Brucker P.* Complexity of machine scheduling problems//Ann. Discr. Math. — 1977. — V. 1. — P. 343–362.
658. *Lenstra J.K., Rinnooy Kan A.H.G., Stougie L.* A framework for the probabilistic analysis of hierarchical planning system//Preprint: Mathematische Centrum. — Amsterdam, 1983. — Afdeling mathematische Beliskunde BW 180/83. — 20 p.
659. *Lenstra J.K., Rinnooy Kan A.H.G., Van Emde Boas P.* An appraisal of computational complexity for operations researchers//Eur. J. Oper. Res. — 1982. — № 11. — P. 201–210.
660. *Lev V., Adiri I.* V-shop scheduling//Eur. J. Oper. Res. — 1984. — V. 18, № 1. — P. 51–56.
661. *Lewandowski J.L., Liu J.W.S., Liu C.L.* SS/TDMA time slot assignment with restricted switching modes//IEEE Trans. Commun. — 1983. — V. COM-31, № 1. — P. 149–154.
662. *Liesegang G., Rüger M.* Letter//Oper. Res. Quart. — 1972. — V. 23, № 4. — P. 591.
663. *Liesegang G., Schirmer A.* Heuristische Verfahren zur Maschinenbelegungsplanung bei Reihenfertigung//Z. Oper. Res. — 1975. — B. 19. — S. 195–211.
664. *Littger K.W.* A new approach for the optimum solution of the M by J production scheduling problem//Int. J. Prod. Res. — 1976. — V. 14, № 1. — P. 77–83.
665. *Liu C.Y., Bulfinch R.L.* On the complexity of preemptive open-shop scheduling problems//Oper. Res. Lett. — 1986. — V. 4, № 2. — P. 71–74.
666. *Lofts N.R.* Multiple allocation of resources in a network — an optimal scheduling algorithm//INFOR. — 1974. — V. 12, № 1. — P. 25–38.
667. *Lomnicki Z.A.* A “branch-and-bound” algorithm for the exact solution of three-machine scheduling problem//Oper. Res. Quart. — 1965. — V. 16. — P. 89–100.
668. *Maggu P.L., Das G.* Equivalent jobs for job-blocks in job scheduling//Opsearch. — 1977. — V. 14, № 3. — P. 277–281.
669. *Maggu P.L., Das G.* Short note on the paper: “Equivalent jobs for job blocks in job sequencing”//Pure Appl. Math. Sci. — 1979. — V. 9, № 1–2. — P. 53–54.
670. *Maggu P.L., Das G.* On idle/waiting time operator $O_{t,w}$ having application to solution of some sequencing/scheduling problem//Pure Appl. Math. Sci. — 1980. — V. 11, № 1–2. — P. 71–78.
671. *Maggu P.L., Das D.* On $2 \times n$ sequencing problem with transportation time of jobs//Pure Appl. Math. Sci. — 1980. — V. 12, № 1–2. — P. 1–6.
672. *Maggu P.L., Das G., Kumar R.* On equivalent-job for job-blocks in $2 \times n$ sequencing problem with transportation-times//J. Oper. Res. Soc. Japan. — 1981. — V. 24, № 2. — P. 136–146.
673. *Maggu P.L., Das G., Singh M., Singhal M.L.* On $3 \times n$ sequencing problem involving equivalent-job for a job-block with transportation-times of jobs//Math. Operationsforsch. Statist. Ser. Optimization. — 1981. — V. 12, № 2. — P. 271–279.
674. *Maggu P.L., Singhal M.L., Mohammad N.* On “ N -job, 2-machine” flow-shop scheduling with minimization of weighted mean flow-time of jobs//Pure Appl. Math. Sci. — 1982. — V. 16, № 1–2. — P. 77–83.
675. *Maggu P.L., Singhal M.L., Mohammad N., Yadav S.K.* On N -job, 2-machine flow-shop scheduling problem with arbitrary time lags and transportation times of jobs//J. Oper. Res. Soc. Japan. — 1982. — V. 25, № 3. — P. 219–227.
676. *Makino T.* On a scheduling problem//J. Oper. Res. Soc. Japan. — 1965. — V. 8, № 1. — P. 32–44.
677. *Malicka-Wasowska J.* Model sekwencyjny z ciągła praca maszyn//Prz. statyst. — 1974. — T. 21, № 2. — S. 211–219.
678. *Manne A.S.* On the job shop scheduling problem//Oper. Res. — 1960. — V. 8. — P. 219–223.
679. *Mason T.A., Moodie C.L.* A branch and bound algorithm for minimizing cost in project scheduling//Manag. Sci. (B). — 1971. — V. 18, № 4. — P. 158–173.
680. *Masuda T., Ishii H., Nishida T.* Some bounds on approximation algorithms for $n|m|1|L_{\max}$ and $n|2|F|L_{\max}$ scheduling problem//J. Oper. Res. Soc. Japan. — 1983. — V. 26, № 3. — P. 212–224.

681. Masuda T., Ishii H., Nishida T. The mixed shop scheduling problem//Discr. Appl. Math. — 1985. — V. 11, № 2. — P. 175–186.
682. Maxwell W., Mehra M. Multiple-factor rules for sequencing with assembly constraints//Nav. Res. Log. Quart. — 1968. — V. 15, № 2. — P. 241–254.
683. McMahon G.B. Optimal production schedules for flow shops//CORS J. — 1969. — V. 7, № 2. — P. 141–151.
684. McMahon G.B. Note on optimal production schedules//CORS J. — 1969. — V. 7, № 2. — P. 154–155, 156.
685. McMahon G.B., Burton P.G. Flow-shop scheduling with the branch-and-bound method//Oper. Res. — 1967. — V. 15, № 3. — P. 473–481.
686. McMahon G.B., Florian M. On scheduling with ready times and due dates to minimize maximum lateness//Oper. Res. — 1975. — V. 23. — P. 475–482.
687. McNaughton R. Scheduling with deadlines and loss functions//Manag. Sci. — 1959. — V. 6, № 1. — P. 1–12.
688. Mensch G. Generalization of Akers "two-dimensional job-shop scheduling model" to n dimensions//SIAM J. Appl. Math. — 1970. — V. 18, № 2. — P. 462–466.
689. Mittal B.S., Bugga P.C. Flow shop scheduling with parallel machines//Cahiers C.E.R.O. — 1979. — V. 21. — P. 87–91.
690. Mitten L.G. Sequencing n -jobs in two machines with arbitrary time lags//Manag. Sci. — 1959. — V. 5, № 3. — P. 293–298.
691. Mitten L.G. A scheduling problem: an analytical solution based on two machine, n jobs arbitrary start and stop lags and common sequence//J. Industr. Eng. — 1959. — V. 10, № 2. — P. 131–135.
692. Mitten L.G. Branch-and-bound methods: general formulation and properties//Oper. Res. — 1970. — V. 18, № 1. — P. 24–34.
693. Miyazaki S. Combined scheduling system for reducing job tardiness in a job shop//Int. J. Prod. Res. — 1981. — V. 19, № 2. — P. 201–211.
694. Miyazaki S., Nishiyama N. Analysis for minimizing weighted mean flow-time in flow-shop scheduling // J. Oper. Res. Soc. Japan. — 1980. — V. 23, № 2. — P. 118–132.
695. Miyazaki S., Nishiyama N., Hashimoto F. An adjacent pairwise approach to the mean flow-time scheduling problem//J. Oper. Res. Soc. Japan. — 1978. — V. 21, № 2. — P. 287–319.
696. Möhring R.H., Radermacher F.J. Scheduling problems with resource duration//Contrib. Prob. Theory, Natur. Resour., Econ. Indices and Relat. Top. — Königstein. — 1984. — P. 423–452.
697. Monma C.L. The two machine maximum flow-time problem with series-parallel precedence constraints: an algorithm and extensions//Oper. Res. — 1979. — V. 27, № 4. — P. 792–797.
698. Monma C.L. Sequencing to minimize the maximum job cost//Oper. Res. — 1980. — V. 28, № 4. — P. 942–951.
699. Monma C.L. Sequencing with general precedence constraints//Discr. Appl. Math. — 1981. — V. 3, № 2. — P. 137–150.
700. Monma C.L., Rinnooy Kan A.H.G. A concise survey of the efficiently solvable special cases of the permutation flow-shop problem//RAIRO Rech. opér. — 1983. — V. 17, № 2. — P. 105–119.
701. Monma C.L., Sidney J.B. Sequencing with series-parallel precedence constraints//Math. Oper. Res. — 1979. — V. 4, № 3. — P. 215–224.
702. Muhlemann A.P., Lockett A.G., Farn C.-K. Job-shop scheduling heuristics and frequency of scheduling//Int. J. Prod. Res. — 1982. — V. 20, № 2. — P. 227–241.
703. Nabeshima I. Sequencing on two machines with start lag and stop lag//J. Oper. Res. Soc. Japan. — 1963. — V. 5, № 3. — P. 97–101.
704. Nabeshima I. On the bound of makespans and its application in M -machine scheduling problem // J. Oper. Res. Soc. Japan. — 1967. — V. 9, № 3–4. — P. 98–136.
705. Nabeshima I. Some extensions of the M -machine scheduling problem//J. Oper. Res. Soc. Japan. — 1967. — V. 10, № 1–2. — P. 1–17.
706. Nabeshima I. General scheduling algorithms with applications to parallel scheduling and multiprogramming scheduling//J. Oper. Res. Soc. Japan. — 1971. — V. 14, № 2. — P. 72–99.

707. *Nabeshima I.* Algorithms for multiprojects scheduling with resource constraints and related parallel scheduling//Дэнки цусин дайгаку гакухо, Repts. Univ. Electro-Comm. — 1972. — V. 23, № 1. — P. 29–50.
708. *Nabeshima I.* A backtrack programming algorithm and reliable heuristics programs for general scheduling problem//Дэнки цусин дайгаку гакухо, Repts. Univ. Electro-Comm. — 1973. — V. 24, № 1. — P. 23–36.
709. *Nabeshima I.* The order of n items processed on m machines. III//J. Oper. Res. Soc. Japan. — 1973. — V. 16, № 3. — P. 163–185.
- 710*. *Nabeshima I.* Notes on analytical results in flow shop scheduling I, II, III, IV//Дэнки цусин дайгаку гакухо, Repts Univ. Electro-Comm. — 1977. — V. 27, № 2. — P. 245–252; 1977. — V. 27, № 2. — P. 253–257; 1977. — V. 28, № 1. — P. 35–44; 1977. — V. 28, № 1. — P. 57–65.
711. *Nabeshima I., Maruyama Sh.* On $n \times m$ mean completion time flow-shop problem: analytical results, extended branch-and-bound algorithm and computational experience//Дэнки цусин дайгаку гакухо, Repts Univ. Electro-Comm. — 1977. — V. 28, № 2. — P. 219–226.
712. *Nawaz M., Ensco E.E., Ham I.* A heuristic algorithm for the m -machine, n -job flow-shop sequencing problem.//OMEGA. — 1983. — V.11, № 1. — P. 91–95.
713. *Neidereichhols J.* Minimale wege in disjunkten graphen//Oper. Res. Verfahren. — 1972. — V. 14. — P. 336–346.
714. *Nemeti L.* Das Reihenfolgeproblem in der Fertigungsprogrammierung und Lenearplanung mit Logischen Bedingungen//Mathematica (RPR). — 1964. — V. 6, № 1. — P. 87–99.
715. *Nghiem P.T.* Les problemes d'ordonnancement avec contraintes disjonctives//Les problemes d'ordonnancement. — Dunod, 1964. — P. 136–151.
716. *Nicholson T.A.J., Pullen R.D.* A permutation procedure for job-shop scheduling//Comput. J. — 1968. — V. 11, № 1. — P. 48–56.
717. *Okamura K., Yamashina H.* Establishment of linear sequences//Mem. Fac. Engng. Kyoto Univ. — 1969. — V. 31, № 3. — P. 307–331.
718. *Page E.S.* An approach to the scheduling of jobs on machines//J. Royal Statist. Soc. Ser. B. — 1961. — V. 23, № 2. — P. 484–492.
719. *Page E.S.* On the scheduling jobs by computer//Comput. J. — 1962. — V. 5, № 3. — P. 214–220.
720. *Page E.S.* On Monte-Carlo methods in congestion problems: I. Searching for optimum discrete situations//Oper. Res. — 1965. — V. 13, № 2. — P. 291–299.
721. *Palmer D.S.* Sequencing jobs through a multi-stage process in the minimum total time. — A quick method of obtaining a near optimum//Oper. Res. Quart. — 1965. — V. 16, № 1. — P. 101–107.
722. *Pandit S.N.N., Subrahmanyam Y.V.* Enumeration of all optimal job sequences//Opsearch. — 1975. — V. 12, № 1–2. — P. 35–39.
723. *Panwalker S.S., Charles O.* Analysis of the left tail for the makespan distribution in flow-shop problems//Opsearch. — 1981. — V. 18. — P. 215–220.
724. *Panwalker S.S., Dudek R.A., Smith M.L.* Sequencing research and the industrial scheduling problem//Lect. Notes Econ. and Math. Syst. — 1973. — V. 86. — P. 29–38.
725. *Panwalker S.S., Iskander W.* A survey of scheduling rules//Oper. Res. — 1977. — V. 25. — P. 45–61.
726. *Panwalker S.S., Khan A.W.* An improved branch and bound procedure for $n \times m$ flow shop problem//Nav. Res. Log. Quart. — 1975. — V. 22, № 4. — P. 787–789.
727. *Panwalker S.S., Khan A.W.* An ordered flow-shop sequencing problem with mean completion time criterion//Int. J. Prod. Res. — 1976. — V. 14, № 5. — P. 631–635.
728. *Panwalker S.S., Khan A.W.* A convex property of a ordered flow shop sequencing problem//Nav. Res. Log. Quart. — 1977. — V. 24, № 1. — P. 159–162.
729. *Panwalker S.S., Khan A.W.* An ordered flow shop sequencing problem with TWK due dates and total tardiness criterion//J. Inst. Math. Appl. — 1979. — V. 23, № 3. — P. 367–372.
730. *Panwalker S.S., Smith M.L., Woollam C.R.* Counterexamples to optimal permutation schedules for certain flow shop problems//Nav. Res. Log. Quart. — 1981. — V. 28, № 2. — P. 339–346.

* Опубликовано более 20 частей этой работы.

731. *Panwalkar S.S., Woollam C.R.* Flow shop scheduling problem with no in-process waiting: a special case//J. Oper. Res. Soc. — 1979. — V. 30, № 7. — P. 661–664.
732. *Panwalkar S.S., Woollam C.R.* Ordered flow shop problem with no in-process waiting: further results//J. Oper. Res. Soc. — 1980. — V. 31, № 11. — P. 1039–1043.
733. *Papadimitriou C.H.* On the complexity of edge traversing//J. Assoc. Comput. Mach. — 1976. — № 23. — P. 544–554.
734. *Papadimitriou C.H., Kanellakis P.C.* Flow-shop scheduling with limited temporary storage//J. Assoc. Comput. Mach. — 1980. — V. 27, № 3. — P. 533–549.
735. *Park Y.B., Pegden C.D., Enscoe E.E.* A survey and evaluation of static flowshop scheduling heuristics//Int. J. Prod. Res. — 1984. — V. 22, № 1. — P. 127–141.
736. *Parker L.G., Rardin R.G.* An overview of complexity theory in discrete optimizations: part I. Concepts; part II. Results and implications//IIE Trans. — 1982. — V. 14, № 1. — P. 3–10.
737. *Peterson C.C.* Solving sequencing problems through reordering operations//AIIE Trans. — 1973. — № 5. — P. 68–73.
738. *Piehlér J.* Ein Beitrag zum Reihenfolgeproblem//Unternehmensforsch. — 1960. — Bd. 4, № 3. — S. 138–142.
739. *Pinedo M.* A note on the two machine job shop with exponential processing times//Nav. Res. Log. Quart. — 1981. — V. 28, № 4.
740. *Pinedo M.* Minimizing the expected makespan in stochastic flow shops//Oper. Res. — 1982. — V. 30. — P. 148–162.
741. *Pinedo M.* A note on the flow time and the number of tardy jobs in stochastic open shops//Eur. J. Oper. Res. — 1984. — V. 18, № 1. — P. 81–85.
742. *Portts C.N.* An adaptive branching rule for the permutation flow-shop problem//Eur. J. Oper. Res. — 1980. — V. 5, № 1. — P. 19–25.
743. *Portts C.N.* Analysis of heuristics for two-machine flow-shop sequencing subject to release dates//Math. Oper. Res. — 1985. — V. 10, № 4. — P. 576–584.
744. *Prasad V.R.* $n \times 2$ flow-shop sequencing problem with random processing times//Opsearch. — 1981. — V. 18. — P. 1–14.
745. *Raimond J.F.* Minimaximal paths in disjunctive graphs by direct search//IBM J. Res. Developm. — 1969. — V. 13. — P. 391–399.
746. *Randolph P.H.* Job shop scheduling — a case study//OMEGA. — 1976. — V. 4, № 4. — P. 463–477.
747. *Randolph P.H., Swinson G., Ellingsen C.* Stopping rules for sequencing problems//Oper. Res. — 1973. — V. 21. — P. 1309–1315.
748. *Reddi S.S.* Sequencing with finite intermediate storage//Manag. Sci. — 1976. — V. 23, № 2. — P. 216–217.
749. *Reddi S.S., Ramamoorthy C.V.* On the flow shop sequencing problems with no wait in process//Oper. Res. Quart. — 1972. — V. 23, № 3. — P. 323–331.
750. *Reddi S.S., Ramamoorthy C.V.* Reply to Dr. Goyal's comments//Oper. Res. Quart. — 1973. — V. 24, № 1. — P. 133–134.
751. *Reddi S.S., Ramamoorthy C.V.* A scheduling problem//Oper. Res. Quart. — 1973. — V. 24. — P. 441–446.
752. *Rendle F.* On the complexity of decomposing matrices arising in satellite communications//Oper. Res. Lett. — 1985. — V. 4, № 6. — P. 5–8.
753. *Richter K.* The robot sequencing problem: polynomial algorithm and complexity//Optimization. — 1985. — V. 16, № 4. — P. 597–605.
754. *Rinnooy Kan A.H.G.* Machine scheduling problems: classification, complexity and computation. — Hague: Martinus Nijhoff, 1976. — 180 p.
755. *Rinnooy Kan A.H.G., Magazine M.J.* Report of the session on scheduling//Ann. Discr. Math. — 1979. — V. 5. — P. 423–426.
756. *Rochette R., Sadowski R.* A statistical comparison of the performance of simple dispatching rules for particular set of job shops//Int. J. Prod. Res. — 1976. — V. 14, № 1. — P. 63–75.
757. *Röck H.* The three-machine no-wait flow-shop problem is NP-complete//J. Assoc. Comput. Mach. — 1984. — V. 31, № 2. — P. 336–345.
758. *Röck H.* Some new results in flow-shop scheduling//Z. Oper. Res. — 1984. — V. 28, № 1. — P. 1–16.
759. *Röck H., Schmidt G.* Machine aggregation heuristics in shop-scheduling//VII Symp. Oper. Res., Nochs. St. Gallen: Proc. Sekt 1–3. — Königstein/Ts., 1983. — P. 303–314.

760. Roy B. Prise en compte des contraintes disjonctives dans la methode du chemin critique// Revue Franc. Inform. Rech. Oper. — 1966. — № 38. — P. 68–84.
761. Roy B. Procedure d'exploration par separation et evaluation// Revue Franc. Inform. Rech. Oper. — 1969. — V. 3, № 1. — P. 61–90.
762. Roy B., Susmann B. Les problemes d'ordonnancement avec contraintes disjonctives// SEMA, Note DS. — Montrouge, 1964. — № 9.
763. Ruiz J.P., Lara R.A. Sobla la implementacion en computadora pequenos de un metodo combinado de secuenciacion// Investig. Operat. — 1977. — № 21–22. — P. 137–145.
764. Sahni S., Cho Y. Complexity of scheduling shops with no wait in process// Math. Oper. Res. — 1979. — V. 4, № 4. — P. 448–457.
765. Sahni S., Horowitz E. Combinatorial problems: reducibility and approximation// Oper. Res. — 1978. — V. 26, № 5. — P. 718–759.
766. Salvador M.S. A solution to a special class of flow shop scheduling problems// Lect. Notes. Econ. Math. Syst. — 1973. — V. 86. — P. 83–91.
767. Schrage L. A bound based on the equivalence of min-max-completion time and min-max-lateness scheduling objectives // Report 7042. — Chicago: University of Chicago, 1970.
768. Schrage L. Solving resource-constrained network problems by implicit enumeration, nonpreemptive case// Oper. Res. — 1970. — V. 18, № 2. — P. 263–278.
769. Schrage L. Solving resource-constrained network problem by implicit enumeration, preemptive case// Oper. Res. — 1972. — V. 20. — P. 668–677.
770. Sekiguchi Y. Optimal scheduling in a GT-type flow-shop under series-parallel precedence constraints// J. Oper. Res. Soc. Japan. — 1983. — V. 26, № 3. — P. 226–251.
771. Shanthikumar J.G., Buzacott J.A. The time spent in a dynamic job shop// Eur. J. Oper. Res. — 1984. — V. 17. — P. 215–226.
772. Shanthikumar J.G., Wu Y.-B. Decomposition approaches in permutation scheduling problems with application to the M-machine flow-shop scheduling problems// Eur. J. Oper. Res. — 1985. — V. 19. — P. 125–141.
773. Shapiro R.D. Scheduling couple tasks// Nav. Res. Log. Quart. — 1980. — V. 27, № 3. — P. 489–498.
774. Sidney J.B. The two-machine maximum flow time problem with series-parallel precedence relation// Oper. Res. — 1979. — V. 27, № 4. — P. 782–791.
775. Sidney J.B. A decomposition algorithm for sequencing with general precedence constraints// Math. Oper. Res. — 1981. — V. 6, № 2. — P. 190–204.
776. Smith M.L., Panwalker S.S., Dudek R.A. Flowshop sequencing problem with ordered processing time matrices// Manag. Sci. — 1975. — V. 21, № 5. — P. 544–549.
777. Smith M.L., Panwalker S.S., Dudek R.A. Flow shop sequencing problem with ordered processing time matrices: a general case// Nav. Res. Log. Quart. — 1976. — V. 23, № 3. — P. 481–486.
778. Smith R.D., Dudek R.A. A general algorithm for solution of the n -job M -machine scheduling problem of the flow shop// Oper. Res. — 1967. — V. 15, № 1. — P. 71–82.
779. Smith R.D., Dudek R.A. Errata// Oper. Res. — 1969. — V. 17, № 4. — P. 756.
780. Smith W.E. Various optimizers for single state production// Nav. Res. Log. Quart. — 1956. — V. 3, № 1–2. — P. 59–66.
781. Smits A.J.M., Baker K.R. An experimental investigation of the occurrence of special cases in the three-machine flowshop problem// Int. J. Prod. Res. — 1981. — V. 19, № 6. — P. 737–741.
782. Spachis A.S., King J.R. Job-shop scheduling heuristics with local neighbourhood search// Int. J. Prod. Res. — 1979. — V. 17, № 6. — P. 507–526.
783. Spencer D.L., Dudek R.A. The dependent shop sequencing algorithm. A general algorithm for solving flow shop, job shop and dependent shop sequencing problems// Int. J. Prod. Res. — 1973. — V. 11, № 4. — P. 325–339.
784. Spinner A.H. Sequencing theory — development to date// Nav. Res. Log. Quart. — 1968. — V. 15, № 2. — P. 319–330.
785. Stanley R.P. Acyclic orientations of graphs// Discr. Math. — 1973. — V. 5, № 2. — P. 171–178.
786. Steinhoff H.W. Two recent developments in scheduling applications// Lect. Notes Econ. Math. Syst. — 1973. — V. 86. — P. 92–108.
787. Stinson J.P., Davis E.W., Khumawala B.M. Multiple resource-constrained scheduling using branch and bound// AIIE Trans. — 1978. — V. 10, № 3. — P. 252–259.

788. *Stinson J.P., Smith A.W.* A heuristic programming procedure for sequencing the static flowshop//Int. J. Prod. Res. – 1982. – V. 20, № 6. – P. 753–764.
789. *Sule D.R.* Sequencing n jobs on two machines with setup, processing and removal times separated//Nav. Res. Log. Quart. – 1982. – V. 29, № 3. – P. 517–519.
790. *Sule D.R., Huang K.Y.* Sequencing on two and three machines with setup processing and removal times separated//Int. J. Prod. Res. – 1983. – V. 21, № 5. – P. 723–732.
791. *Sussmann B.* Scheduling problems with interval disjunctions//Z. Oper. Res. (A) – 1972. – V. 16, № 5. – P. 165–178.
792. *Syslo M.* On some machine sequencing problems (II)//Zastosow matem. – 1974. – T. 14, № 1. – S. 93–97.
793. *Syslo M.* O złożoności obliczeniowej problemów kombinatoryki i teorii grafów//Rocz. Polsk. towarzystwa matem. Ser. 3. Matematyka stosowana. – 1980. – T. 16. – S. 53–98.
794. *Szwarc W.* Solution of the Akers – Friedman scheduling problem//Oper. Res. – 1960. – V. 8, № 6. – P. 782–788.
795. *Szwarc W.* O pewnym zagadnieniu kolejności II//Prz. statyst. – 1963. – T. 10, № 1. – S. 139–154.
796. *Szwarc W.* On some sequencing problems//Nav. Res. Log. Quart. – 1968. – V. 15, № 2. – P. 123–155.
797. *Szwarc W.* Elimination methods in the $m \times n$ sequencing problem//Nav. Res. Log. Quart. – 1971. – V. 18, № 3. – P. 295–305.
798. *Szwarc W.* Optimal elimination methods in the $m \times n$ flow-shop scheduling problem//Oper. Res. – 1973. – V. 21, № 6. – P. 1250–1259.
799. *Szwarc W.* Mathematical aspects of the $3 \times n$ job-shop sequencing problem//Nav. Res. Log. Quart. – 1974. – V. 21, № 1. – P. 145–153.
800. *Szwarc W.* A note on mathematical aspects of the $3 \times n$ job-shop sequencing problem//Nav. Res. Log. Quart. – 1974. – V. 21, № 4. – P. 725–726.
801. *Szwarc W.* Remarks//Oper. Res. – 1975. – V. 23, № 5. – P. 1043.
802. *Szwarc W.* Optimal two-machine ordering in the $3 \times n$ flow-shop problem//Oper. Res. – 1977. – V. 25, № 1. – P. 70–77.
803. *Szwarc W.* Special cases of the flow-shop problem//Nav. Res. Log. Quart. – 1977. – V. 24, № 3. – P. 483–492.
804. *Szwarc W.* Dominance conditions for the three machine flow-shop problem//Oper. Res. – 1978. – V. 26, № 1. – P. 203–206.
805. *Szwarc W.* Permutation flow-shop theory revisited//Nav. Res. Log. Quart. – 1978. – V. 25, № 3. – P. 557–570.
806. *Szwarc W.* Extreme solution of the two-machine flow-shop problem//Nav. Res. Log. Quart. – 1981. – V. 28, № 1. – P. 103–114.
807. *Szwarc W.* A note on the flow-shop problem without interruptions in job processing//Nav. Res. Log. Quart. – 1981. – V. 28, № 4. – P. 665–669.
808. *Szwarc W.* Precedence relations of the flow-shop problem//Oper. Res. – 1981. – V. 29, № 2. – P. 400–411.
809. *Szwarc W.* Flow-shop problems with time lags//Manag. Sci. – 1983. – V. 29, № 4. – P. 477–491.
810. *Szwarc W.* The flow-shop problem with mean completion time criterion//IIE Trans. – 1983. – V. 15, № 2. – P. 172–176.
811. *Szwarc W., Hutchinson G.K.* Johnson's approximate method for the $3 \times n$ job shop problem//Nav. Res. Log. Quart. – 1977. – V. 24, № 1. – P. 153–157.
812. *Tabourier Y.* Schémas d'ordonnement à contraintes purement disjonctives//Revue Franc. Inform. Rech. Oper. – 1969. – V. 5, № 3. – P. 51–66.
813. *Takagi M.* Graphical solution of 2-job m -machine sequencing problem//Дэнки сикэнсэ ихо. Bull. Elektrotechn. Lab. – 1961. – V. 25, № 8. – P. 575–584.
814. *Tang D.T., Wong C.K.* A modified branch-and-bound strategy//Inf. Process. Lett. – 1973. – V. 2, № 3. – P. 65–69.
815. *Tarjan R.E.* Algorithms for maximum network flow//Math. Progr. Study. – 1986. – V. 26. – P. 1–11.
816. The traveling salesman problem. A guided tour of combinatorial optimization. – Chichester: Wiley, 1985. – 465 p.
817. *Thesen A.* Heuristic scheduling of activities under resource and precedence restrictions//Manag. Sci. – 1976. – V. 23, № 4. – P. 412–422.

818. *Towsend W.* Sequencing n jobs on m machines to minimize maximum tardiness: a branch-and-bound solution// *Manag. Sci.* — 1977. — V. 23. — P. 1016–1019.
819. *Towsend W.* A branch-and-bound method for sequencing problem with linear and exponential penalty functions// *Oper. Res. Quart.* — 1977. — V. 28, № 1ii. — P. 191–200.
820. *Towsend W.* Minimizing the maximum penalty in the two-machine flow-shop// *Manag. Sci.* — 1977. — V. 24, № 2. — P. 230–234.
821. *Towsend W.* Minimizing the maximum penalty in the m -machine sequencing problem// *Oper. Res. Quart.* — 1977. — V. 28, № 3 ii. — P. 727–734.
822. *Turksen I.B., Shankar R.* Some extensions of Akers-Friedman production scheduling problem// *Lect. Notes Econ. Math. Syst.* — 1973. — V. 86. — P. 426–431.
823. *Ullrich M.* Konstruktion optimaler Reihenfolgen bei speziellen Reihenfolgeproblemen// *Math. Operationsforsch. Statist.* — 1976. — Bd. 7, № 3. — S. 361–370.
824. *Ullrich M.* Zusammenhang zwischen dem Rundreiseproblem und einem Reihenfolgeproblem der Maschinenbelegungsplanung// *Math. Operationsforsch. Statist.* — 1976. — Bd. 7, № 6. — S. 843–846.
825. *Viviers F.* A decision support system for job shop scheduling// *Eur. J. Oper. Res.* — 1983. — V. 14. — P. 95–103.
826. *Wagner H.M.* An integer linear programming model for machine scheduling// *Nav. Res. Log. Quart.* — 1959. — V. 6. — P. 131.
827. *Wang M.-F., Rosenshine M.* Scheduling for a combination of made-to-stock and made-to-order jobs in a job shop// *Int. J. Prod. Res.* — 1983. — V. 21, № 5. — P. 607–616.
828. *Warshall S.* A theorem on boolean matrices// *J. Assoc. Comput. Mach.* — 1962. — V. 9, № 1. — P. 11–12.
829. *Werra D., de.* On some combinatorial problems arising in scheduling// *CORS J.* — 1970. — V. 8, № 3. — P. 165–175.
830. *Werra D., de.* Balanced scheduling// *INFOR. Canad. J. Oper. Res. Inf. Proc.* — 1971. — V. 9, № 3. — P. 230–237.
831. *Werra D., de.* Scheduling in sports// *Ann. Discr. Math.* — 1981. — V. 11. — P. 381–395.
832. *Werra D., de.* Preemptive scheduling, linear programming and network flows// *SIAM J. Alg. Discr. Meth.* — 1984. — V. 5, № 1. — P. 11–20.
833. *Werra D., de.* A decomposing property of polyhedra// *Math. Progr.* — 1984. — V. 30, № 2. — P. 261–266.
834. *Werra D., de.* Variations on the integral decomposition property// *Math. Progr. Study.* — 1986. — V. 26. — P. 197–198.
835. *Wie S.-H., Pinedo M.* On minimizing the expected makespan and flow time in stochastic flow shops with blocking// *Math. Oper. Res.* — 1986. — V. 11, № 2. — P. 336–342.
836. *Wilbrecht J.I., Prescott W.B.* The influence of setup time on job shop performance// *Manag. Sci.* — 1969. — V. 16, № 4. — P. 274–280.
837. *Wisner D.A.* Solution of the flowshop-sequencing problem with no intermediate queues// *Oper. Res.* — 1972. — V. 20, № 3. — P. 689–697.
838. *Yoshida T., Hitomi K.* Optimal two-stage production scheduling with setup times separated// *AIIE Trans.* — 1979. — V. 11, № 2. — P. 261–263.
839. *Zorychta K.* Optymalizacja dynamiczna na zbiorze permutacji// *Rocz. Polsk. towarzystwa matem. Ser 3. Matematyka stosowana.* — 1977. — T. 9. — S. 25–33.